Charles University in Prague

1st Faculty of Medicine

Study programme: Biomedical Informatics

Studijní program: Biomedicínská informatika

Mgr. Miroslav Nagy

Clinical Content Harmonisation of Electronic Health Record

Harmonizace klinického obsahu elektronického zdravotního záznamu

PhD Thesis
Disertační práce

Supervisor/Školitel: RNDr. Antonín Říha, CSc.
Prague, 2011

Prohlášení

Prohlašuji, že jsem závěrečnou práci zpracoval samostatně a že jsem řádně uvedl a citoval všechny použité prameny a literaturu. Současně prohlašuji, že práce nebyla využita k získání jiného nebo stejného titulu.

Souhlasím s trvalým uložením elektronické verze mé práce v databázi systému meziuniverzitního projektu Theses.cz za účelem soustavné kontroly podobnosti kvalifikačních prací.

V Praze, 29. 3. 2011

Miroslav Nagy

Podpis

Acknowledgement

Author's research presented in this thesis was supported by the following projects: the project of the Academy of Sciences of the Czech Republic No. 1ET200300413, the project of the Institute of Computer Science of Academy of Sciences of the Czech Republic No. AV0Z10300504 and the project of the Ministry of Education, Youth and Sports of the Czech Republic No. 1M06014.

The author would like to very much thank his supervisor RNDr. Antonín Říha, CSc. for his priceless comments and help during the research. The thanks also go to prof. RNDr. Jana Zvárová, DrSc., the head of the Centre of Biomedical Informatics, for institutional and material support. Author's thanks also go the team of researchers at the Department of Medical Informatics at the Institute of Computer Science Academy of Sciences of the Czech Republic and his family for their support during the PhD studies and help with preparation of this thesis.

Identification Record

NAGY, Miroslav. *Clinical Content Harmonisation of Electronic Health Record.* PhD Thesis. Prague, 2011. 119 pages. Charles University in Prague, 1st Faculty of Medicine, Dept. of Medical Informatics ICS AS CR. Supervisor Riha, Antonin.

Identifikační záznam

NAGY, Miroslav. *Harmonizace klinického obsahu elektronického zdravotního záznamu. [Clinical Content Harmonisation of Electronic Health Record].* Praha, 2011. 119 s. Dizertační práce. Univerzita Karlova v Praze, 1. lékarská fakulta, Oddělení medicínské informatiky ÚI AV ČR. Vedoucí práce Říha, Antonín.

Abstract

Title: Clinical Content Harmonisation of Electronic Health Record

Author: Mgr. Miroslav Nagy

Department: Department of Medical Informatics, Institute of Computer Science v.v.i., Academy of Sciences of the Czech Republic

Supervisor: RNDr. Antonín Říha, CSc.

Supervisor's e-mail: riha@euromise.cz

This PhD thesis deals with the possibilities of clinical content harmonisation in electronic health records (EHR). In the first sections, the current state of the art in the field of EHR and other related fields is summarised, as well as that of the tools supported by current software technologies and the results of research in the field of medical informatics. The work is focused on analysis of obstacles in sharing medical data and on proposals of steps for overcoming these obstacles. Next, the utilized tools, modelling approaches, standards and coding systems used during the research are summed up. In the following sections the methods pointing to achieving harmonized clinical models are elaborated on and resulting clinical concepts mappings, supporting tools, developed EHR systems and semantic interoperability platforms are presented. In the conclusion these results are discussed and some future work is proposed.

Keywords: electronic health record, semantic interoperability, clinical content, harmonisation, HL7, archetypes

Abstrakt

Název: Harmonizace klinického obsahu elektronického zdravotního záznamu

Autor: Mgr. Miroslav Nagy

Pracoviště: Oddělení medicínské informatiky, Ústav informatiky v.v.i., Akademie věd České Republiky

Školitel: RNDr. Antonín Říha, CSc.

E-mail školitele: riha@euromise.cz

Tato disertační práce se zabývá možnostmi harmonizace klinického obsahu v elektronickém zdravotním záznamu (EZZ). První část mapuje současný stav výzkumu v oblasti EZZ a příbuzných oborů, nástroje podporované nejnovějšími softwarovými technologiemi a aktuální výsledky výzkumu na poli biomedicínské informatiky. Dále následuje analýza překážek zabraňujících sdílení medicínských dat a návrhy jejich řešení. Práce shrnuje veškeré nástroje, modelovací přístupy, standardy a kódovací systémy použité během výzkumu. Další část práce se zabývá metodami aplikovanými v procesu harmonizace klinických modelů a popisuje výsledky v podobě mapování klinických konceptů, podpůrných nástrojů, vyvinutých systémů EZZ a platforem pro sémantickou interoperabilitu. Závěr práce nabízí celkové shrnutí výsledků výzkumu a jejich diskusi, stejně jako výhled do budoucnosti.

Klíčová slova: elektronický zdravotní záznam, sémantická interoperabilita, klinický obsah, harmonizace, HL7, archetypy

Contents

Chapter 1

Introduction

It is becoming common today that data are recorded in an electronic form. The reason for this is not only an easier manipulation and faster communication of the data but also the fact, that their sender and recipient might be a computer application. Electronic health record (EHR) builds on these premises. It is a crucial part of patient's healthcare documentation, serving as a basis for personalized healthcare based on knowledge and medical guidelinesand thus contributing to a more complex, effective and safer healthcare.

EHR, as described in [1] is ”a longitudinal, electronic repository of information collected during a patient's encounters with healthcare providers and services such as medical history, results of physical examinations, progress notes, laboratory data and imaging reports.”

The difficulty with EHRs is that they are naturally distributed among various healthcare providers who have treated the patient during his or her life. This happens mostly on a regional level, however due to the recent socio-economic changes and the still increasing mobility of individuals for reasons of work and leisure, the number of patient's EHR fragments is rising, especially on international level.

The records at each healthcare provider within both, the patient's home country and abroad, are for various reasons in different formats. Their interoperability is thus limited which prevents the healthcare providers from disposing of a complete record of patient's health status and from the ability to treat him optimally. Overall, the communication and understanding of supplied information may be prevented or impeded by a number of factors such as the text structure, language, set of monitored characteristics and their identification, way of treatment, context, the data type and data structure, definitions of the concepts (we neglect here the temporal or legal limitations, or technical errors in the report including obsolete data).

These obstacles can be reduced by the adoption of a standard or at least a mutual mapping between the standards (terminology definitions, the definition of data structures, the definitions of communication). There exist some well-known standards such as CEN EN 13606 and HL7, openEHR approaches, as well as terminologies such as SNOMED CT, ICD-10, UMLS. Unfortunately, they have not yet been universally accepted and implemented and thus individual countries and even individual vendors have a number of their own local standards.

In order to enable the communication between particular healthcare providers, so-called ”semantic interoperability” is desired, which [2] describes by a following scenario:

”56 year old Padraig recently moved from Ireland to Spain to take up his new job in a multinational IT company. A few weeks after arriving, he falls ill, consults his local (Spanish) GP and is transferred to the next hospital for further tests. (...) In the ideal situation (full semantic interoperability, cooperability) and after thorough authentication, the Spanish hospital information system is able to automatically access, interpret and present all necessary medical information about Padraig to the physician at the point of care. Neither language nor technological differences prevent the system to seamlessly integrate the received information into the local record and provide a complete picture of Padraig's health as if it was collected locally. Further, the anonymised data feeds directly into the tools of public health authorities and researchers.”

Many papers, especially in the last few years, have been dealing with the problem as to how to establish semantic interoperability among various EHR systems [3], [4], [5]. As stated in [6], the semantic interoperability has 4 prerequisites. They are a standardised EHR reference model, standardized service interface models, a standardized set of domain-specific concept models and standardized terminologies. An example of a very concise definition of semantic interoperability provides [7]:”Semantic interoperability is the ability of information systems to exchange information on the basis of shared, pre-established and negotiated meanings of terms and expressions".

The problem of clinical content harmonisation has similar objectives: unambiguous semantics of a common information model connected to international nomenclatures and ontologies. Having EHRs with harmonized clinical content (HCC) and a particular message development process, the semantic interoperability could be achieved. In other words, we can say that apart from the technical realisation of communication between systems, the harmonisation of clinical content of the participating systems is all that is necessary for their semantic interoperability.

The main obstacle to the deployment of EHR systems into the real world is the existence of former well-established information systems in healthcare facilities. Therefore the modularity and openness of the systems used for mutual integration of the inhomogeneous parts of the health record originating from various sources start to play a significant role in life-long patient healthcare.

The exchange of data stored in EHR was described and published in several European and international standards (e.g. CEN ENV 13606 [8], HL7 version 3 [9]). These standards provide formalisms for clinical content description and for structure specification of the transported data. Communication standards form a necessary but not sufficient prerequisite for data sharing among heterogeneous health records. Although the EHR has been studied in many recent projects, only few of them were able to integrate the developed EHR systems into real-life environments.

1.1 Harmonisation of the Clinical Content

In the next paragraphs the importance and difficulties of the Clinical Content Harmonisation process will be explained. In order to clarify the problems connected with the process of the Harmonisation of Clinical Content, it will be likened to the problems dealt with in the field of electric energy transportation in the past.

Electric wires conduct electricity, which can be understood in our case as data. Wires form a power supply network which we can imagine as a data interchange platform (or simply a computer network). Electric sockets in our houses or offices are then the entry

Description	**Electricity**	**Healthcare**
subject of concern	electric energy	data
producer	power plant	some system in healthcare
transport	electric wires	LAN/WAN
consumer	electric appliance	plugin of a HIS
user (service consumer)	power line communication	user application

Table 1.1: The explanation of similarities between electricity transportation and medical data exchange.

points for electric devices to the electric network, similar to the entry points to the interchange platform, and electric appliances are the consumers of the electric energy.

The first power plants were not connected to a national electric transportation network. During the development of commercial electric power systems in the late 19th and early 20th century, many different frequencies (and voltages) were used. Large investments in the equipment at one frequency made the standardisation a slow process. However, as of the turn of the 21st century, places that now use the 50 Hz frequency tend to use 220-240 V, and those that now use 60 Hz tend to use 100-120 V. Both frequencies co-exist today (some countries such as Japan use both) with no technical reason to prefer one over the other and no apparent desire to complete the worldwide standardisation.

The above-mentioned problems are almost exactly the same as those dealt with in the clinical content harmonisation process. Nobody really wants the clinical content to be harmonised because it is not an indispensable feature today and its achievement is very expensive and time consuming. The support of national governments is slow and eHealth is expected to be established in a long period of time. Until somebody needs the harmonised clinical content in eHealth, it will remain an expensive and almost unattainable ”tool”. Alas, the vicious circle is complete.

1.1.1 Relation to Semantic Interoperability

In order to achieve semantic interoperability of EHR information, there are four prerequisites, the first two of witch are also being required for functional interoperability [6]:

- a standardized EHR reference model, i.e. the EHR system architecture, between the sender (or sharer) and receiver of the information,
- standardized service interface models to provide interoperability between the EHR service and other services such as demographics, terminology, access control and security services in a comprehensive clinical information system,
- a standardized set of domain-specific concept models, i.e. archetypes and templates for clinical, demographic, and other domain-specific concepts, and
- standardized terminologies which underpin the archetypes. Note that this does not mean that there needs to be a single standardized terminology for each health domain, but terminologies used should rather be associated with controlled vocabularies.

The last two points are closely related to the clinical content harmonisation provided that an agreement on domain specific concepts and terminologies has been reached.

1.2 The Research Outline

1.2.1 Motivation

Cross border interoperability is highly suggested even by the European Commission to the member states [10]. In the future the need for interoperable electronic healthcare systems will rise significantly. All stakeholders will see more advantages in using documents in an electronic form. Moreover, the clinical data of patients will have annotated their meaning/semantics, which will ease the complex, high-quality and computer-aided healthcare throughout their lives. This implies that one of the cornerstones of the eHealth in the future is the accurate and computer processable semantics annotation of a patient's data harmonised throughout the world.

1.2.2 The Aim of the Thesis

- Study the latest achievements in the field of electronic health record, software technologies and other outcomes of the medical informatics research,
- Identify the obstacles in medical data sharing,
- Describe possible overcoming methods,
- Analyse the clinical content included in healthcare documentation in the field of cardiology or dentistry,
- Annotate the semantics of clinical content in the given field by means of established terminologies,
- Describe the formalised clinical content in given field by openEHR archetypes,
- Test the communication possibilities of the created archetypes via communication standards.

1.3 The Thesis Outline

This thesis is structured in a similar way as scientific papers. It adheres to the IMRAD methodology which proposes following structure: introduction, material and methods, results and discussion. Chapter 1 apprises the reader with the basic problems that are studied in the frame of this thesis. The next chapter summarises the latest achievements in the field of electronic health record, software technologies and other outcomes of the medical informatics research, limited to early stages of the research (years 2005, 2006 and 2007). In Chapter 3 the obstacles in sharing medical data are identified. Solutions to particular obstacles are proposed by the author. Chapters 4, 5 and 6 describe the materials used during the research – in the form of tools, standards, electronic health records, clinical content models from the field of cardiology and dentistry, etc.). The most important part of this thesis is Chapter 7. It describes author's experience in clinical knowledge mapping to established coding systems and standard reference models as well as the resulting mappings. It also presents the results in the field of EHR systems design and development in the field of cardiology and dentistry, in which the author participated mainly as a database architect and programmer. The following sections summarise author's achievements in formal models comparison utilising established solutions. Chapter 7 also includes proposals of two approaches to semantic interoperability platforms, one based on HL7 v3 standard and the other based on openEHR approach. Chapter 8 discusses the results of this thesis research which is finally concluded in Chapter 9.

Software components and applications developed in the frame of the research described in this thesis are gathered in the repository accessible at the following URL address: `http://neo.euromise.cz/nagy/pgs`.

Chapter 2

State of the Art

In this chapter we start with a section that will briefly describe the Czech national healthcare system, especially the data flow in the entire healthcare. Then, two distinct forms of communication will be described and some examples will be introduced. Further, the current efforts in the field of harmonisation of healthcare standards and clinical content will be summarised. Next section will deal with ”materials” that are available to support the research of clinical content harmonisation. It reviews the outcomes of the research in the field of electronic health records (EHRs). Another section will introduce the possibilities of communication in healthcare via standardised protocols. Since the research conducted for the purposes of this thesis took place in the Czech healthcare environment, one section will be dedicated to the Czech efforts in the field of communication standards and making use of them, as well as mentioning some eHealth projects. The final section of this chapter will deal with other research projects, that have more or less similar aim as the one of this thesis.

The state of the art description will make clear the starting position of the research described in this thesis. It is worth noting that this section describes situation which was valid in 2005-07, when this PhD research began, with several updates till the present.

2.1 Flow of Data in Healthcare

From our point of view the most interesting part of the healthcare system description will be the one concerning the flow of data. In the following list some possibilities of data exchange between various institutions/systems are summarised. The term Healthcare facility (HF) is used to denote healthcare institutions from GP practices, through policlinics to large hospitals.

- HF - HF – communication among different levels of HF within its hierarchy (local vs. regional vs. university hospital),
- HF - insurance company – compulsory communication performed in order to earn money,
- HF - medical registry – unidirectional communication where HF sends information about patients with specific diagnoses to specialised registries,
- HF - national statistical office – transportation of statistical summaries to national statistical office,
- HF - laboratory – communication transfering data with the highest degree of structuring, which can flow within a monolithic HIS or from an external laboratory information system.

In healthcare environment two kinds of data exchange (communication) could be recognised – passive and active. **Passive communication** is realized between healthcare institution and registries gathering data of patient with particular diagnosis (e.g. joint replacement, organ transplantation and oncology). **Active communication** is actively initiated by a request or query. Messages in active communication process are typically in a form of application forms, various documents (e.g. medical treatment summary), structured forms (e.g. laboratory results) etc.

2.2 Research Outcomes in the Field of Electronic Health Records

In this section a brief remark will be made on the projects which radically influenced EHR structure and semantics.

2.2.1 openEHR Foundation

The openEHR Foundation is a not-for-profit company, limited by guarantee. Its founding shareholders are University College London, UK and Ocean Informatics Pty. Ltd., Australia. This foundation responded to a need for continuity of research conducted formerly in European and Australian projects (1992-2003): GEHR (1992-1994, supported by the Advanced Informatics in Medicine initiative), Synapses (1995-1998), Synex, Medicate and 6-winit (2001-2002).

The efforts of the openEHR Foundation concentrate on creating well-formulated clinical requirements; rigorous development, implementation and evaluation methodology for systems; common information model for the record and promoting the evolution towards high quality and cost-effective EHR solutions offered. Finally, it concentrates on empirical evaluation of systems performance against clinical requirements.

One of the fundamental contributions of the openEHR Foundation to the research of this thesis are the openEHR clinical content models which are known as archetypes and templates. Archetypes are the foundation building blocks at the clinical concept level; templates aggregate and constrain the archetypes to create context-specific clinical content for use in direct patient care. With the notion of archetypes the 2-level modelling approach is introduced. Archetypes, openEHR templates and the 2-level modelling methodology will be described later.

2.2.2 SNOMED CT

SNOMED CT (Systematized Nomenclature of Medicine – Clinical Terms) is according to [11] a comprehensive clinical terminology, originally created by the College of American Pathologists (CAP) and, as of April 2007, owned, maintained, and distributed by the

International Health Terminology Standards Development Organisation (IHTSDO), a not-for-profit association in Denmark.

On web pages of IHTSDO [12] the aim and benefits of the SNOMED CT can be found. It aims to contribute to the improvement of patient care through underpinning the development of systems to accurately record health care encounters and to deliver decision support to health care providers. Ultimately, patients will benefit from the use of SNOMED CT fro the following reasons: clear description and precise recording of their care process, creation of improved data exchange and interoperability level of EHRs, and creation of health care decision support systems.

The main contribution of SNOMED CT terminology to this thesis is its acceptance by professionals and a huge range of covered domains and institutional mechanisms enabling its translation to various languages (which, however, has not yet been realised in the Czech language). More detailed information about SNOMED CT and the description of its utilisation in this thesis can be found in Section 4.5.2.

2.2.3 MUDR

MUDR EHR system is an implementation of MUltimedia Distributed Record proposed at the EuroMISE Centre. The pilot implementation of MUDR was presented as a part of a master thesis [13] in 2002. The system was designed as a 3-layer application (database, application layer and user interface).

The main contribution of the MUDR EHR was the separation of the clinical knowledge from the patient data. The terms knowledge base, knowledge trees, knowledge nodes were established. The knowledge base consists of knowledge trees and each tree is build of knowledge nodes. Each edge connecting two knowledge nodes specifies its own type, which enables creating a more complex structure similar to ontology. The prevailing type of knowledge edges was the type ”inferior” connecting parent nodes with their children. Another edge-type was the ”equivalence” type, which enabled making a link between two equivalent concepts represented by two different knowledge nodes.

Patient data were stored using data-files organised in tree structure. Each data-file was linked to a knowledge node that determined its meaning and datatype. Supported datatypes could be divided into three groups: basic (BOOLEAN, TEXT, NUMBER), multimedia (PICTURE, AUDIO, VIDEO, LONGTEXT) and reference types (DATA_REF, KN_REF).

The MUDR exists in further two implementations called MUDRorb [14] using CORBA in 2004 and a Java implementation named MUDRj [15] published in 2008.

This PhD thesis mostly employes the data modelling technique proposed by the MUDR EHR, which was applied in the field of dentistry and a resulting model is described on page 60. The MUDR EHR itself will be described in Section 5.3.1.

2.2.4 MUDRLite

MUDRLite EHR was developed at EuroMISE Centre by J.Spidlen as part of his PhD thesis [16] in 2005. The motivation of creating such an EHR was filling the gap in deployment possibilities of the MUDR EHR. The MUDRLite EHR was designed to be less complicated, easier to manage and suitable for use in small environments like outpatient departments or small clinics. This system comprises only two layers – database and user interface combining the application logic.

The aim of the MUDRLite EHR system was to provide just a basic functionality and to be extensible according to special needs in a particular environment. The extensibility was mainly achieved by introducing an XML-based language (MLL - MUDRLite Language) for describing the user interface, which completes similar task as the HTML language, defining the look of a web-page.

The most important features of the MUDRLite EHR exploited in this thesis were its data models defined directly as relational database schemas and the special element of the MLL language enabling external components to extend the functionality of the MUDRLite EHR. More detailed description of this EHR system is on page 54 of this thesis or in [17].

2.3 Communication Research Outcomes

This section summarises most important results in the field of clinical data exchange. The projects mentioned are crucial for the research described within this thesis.

2.3.1 HL7

The HL7 is a non-for-profit, ANSI-accredited standards developing organisation (SDO) dedicated to providing a comprehensive framework and related standards for the exchange, integration, sharing, and retrieval of electronic health information that supports clinical practice and the management, delivery and evaluation of health services [9].

The HL7 is also a name for some of the standards developed by HL7 organisation. Namely it refers to HL7 version 2.x messaging standard and HL7 version 3 messaging standard.

The abbreviation HL7 stands for Health Level Seven, which refers to the seventh level, i.e. application layer, of the ISO/OSI reference model [18].

The HL7 v2.x will be described in Section 4.2 on page 35 and the HL7 v3 will be mentioned in much more detail in Section 4.3 on page 39.

2.3.2 EN 13606

The European Standard EN 13606 (EN – Europäische Norm) is a multipart standard named ”Health Informatics – Electronic health record communication” and consists of the following parts:

1. Reference model – comprehensive, generic model for communicating part or all of an EHR between heterogeneous systems

2. Archetypes interchange specification – constraint-based approach for defining clinical ”business objects” that are built from the Reference Model

3. Reference archetypes and term lists – an initial set of inter-reference model conversion archetypes, mapping to openEHR and to the HL7 version 3 RIM Act classes, vocabularies for the Part 1 model

4. Security – measures and models to share the access control, consent and auditability of EHR communications

5. Messages for exchange – message and service interfaces to enable EHR and archetype communication

This standard was released in 2007 by Technical Comittee CEN/TC251 ”Health informatics” and supersedes the former version ENv 13606 [8]. Its scope is to specify the communication of part or all of the EHR of a single identified subject of care between EHR systems, or between EHR systems and a centralised EHR data repository. Another possibility of using this standard is to establish communication between EHR-S/repository and clinical applications or middleware components (e.g. decision support systems) that need to access or provide EHR data. However, this standard is not intended to specify the internal architecture or database design of EHR-S [19].

2.4 State of Harmonisation Efforts

In the scientific literature many references to the harmonisation process can be found. In must be stated clear that not all harmonisation activities are connected with the aim of this thesis. Two different approaches can be distinguished: harmonisation of standards in healthcare and harmonisation of clinical content in an electronic health record. The harmonisation of clinical content is discussed in this thesis.

2.4.1 Harmonisation of Standards

The harmonisation of standards typically starts with consolidation and mapping of datatypes. Further specific features of standards are much more complicated to synchronise or decide which processes are equivalent.

Some efforts in mapping standards one to each other can be seen in the standard EN 13606 part 3 published in 2007. Appendix A2 of this document shows mapping of classes (`Entry`, `Action`, `Instruction`, `Observation`, `Evaluation`) defined within openEHR Reference Model to respective classes (`RECORD_COMPONENT` class and its inheritors) of the EN 13606 standard. Some classes from the HL7 v3 Clinical Statement model are mapped to EN 13606 classes in Appendix A3 of the 3rd part of the EN 13606 standard. Following HL7 classes can be found in this mapping: `Observation`, `Procedure`, `SubstanceAdministration`, `Supply`, `Act` and `Encounter`. Finally, a clinical example of the mapping between HL7 v3 and EN 13606 can be found in the annex B. This example contains HL7 v3 representation of the Barthel Index and a table of correspondence between HL7 v3 classes and classes defined within the EN 13606 standard.

Mapping of archetypes based on EN 13606 model and openEHR RM can be found in [20] by Catalina Martinez-Costa et al. Its authors address the semantic interoperability of two EHR standards: openEHR and EN 13606; both of which follow the dual model approach (see p. 43). This approach distinguishes information and knowledge, where the latter one is being represented through archetypes. The proposed solution is capable of transforming OpenEHR archetypes into ISO EN 13606 and vice versa by combining semantic web and model-driven engineering technologies. This work, however, was published in 2010 and could not be included in this PhD thesis as one of its starting conditions.

The HARP project is another initiative dealing with attempts of harmonisation.

Despite many positive results in synchronising communication standards in healthcare there exists some criticism of the standards themselves. Barry Smith published a critical article about the HL7 RIM at the MIE 2006 conference [21]. His main objection was that even after 10 years of development work, the RIM is still subject to a variety of logical and ontological flaws.

2.4.2 Harmonisation of Clinical Content

Currently there are a lot of non-harmonized data dictionaries and data models at regional level, scientific community level, national level and as well in the other healthcare related organisations. This needs to be harmonized if we want to not only exchange but also effectively share comparable data. There has been no success so far, nevertheless the UMLS [22] is a good starting point of this long journey.

EHR systems with harmonized clinical content are the most appropriate ones to achieve semantic interoperability. Their clinical content that refers to the same domain is ready for exchange with minimum transformations and mappings during its delivery. However, the actual implementation of this approach is lacking commonly acceptable repository of domain specific models associated to an all-embracing coding system or an ontology.

The HL7's vision of the term ”harmonisation” is the following consensus process [23]. No changes are introduced to the RIM without first providing an opportunity for the entire HL7 working group to review and comment on the proposed change. Following a defined comment period, representatives from each Work Group review the proposed changes, along with any comments received from the working group. Finally, a consensus process is used to determine which of the proposed changes are actually applied to the RIM. This process, referred to as ”harmonisation”, ensures that the RIM is a shared view of the entire working group.

2.5 National Efforts

Sharing and reusing the data among different institutions in the Czech healthcare environment is at relatively low level. The majority of healthcare information systems in the Czech Republic communicate with each other using a national communication standard called DASTA [24], which is based on the national nomenclature called National code-list of laboratory items (NCLP) [24]. These standards are developed and administered by the developers of healthcare information systems that are either specialized companies, university IT centers or research institutions in the Czech Republic. The development of the standard is supported by the Czech Ministry of Health. DASTA is specialized mainly in transfer of requests and results of laboratory analyses. The current version of DASTA is XML based and provides also the functionality for sending statistical reports to the Institute of Health Information and Statistics of the Czech Republic [25] and limited functionality of free text clinical information exchange. Unfortunately, the DASTA has almost no relation to international communication standards such as HL7 [9] or European standards like EN13606 [26].

2.5.1 National Healthcare System

In order to understand problems with sharing data in the Czech healthcare a brief summary of relevant institutions and information systems is provided. The Czech healthcare system is governed and administered by following institutions:

Ministry of the Interior of the Czech Republic – and its division for informatics manages public state information systems, provides their standardisation and interconnection. It also establishes and operates internet portals of public service.

Ministry of Health of the Czech Republic – mainly through its Department for Informatics manages and checks all activities in the field of healthcare informatics. It constitutes, manages and processes data from various healthcare information systems, as well as registries for hygiene.

Institute of Health Information and Statistics of the Czech Republic (IHIS CR) – Its main task and object of activity is management and co-ordination of the National Health Information System (NHIS), which is crucial for national healthcare management. This institute collects data and conducts statistical analyses in order to manage the whole healthcare system.

State Institute for Drug Control – manages information systems used in state drug politics.

The National Institute of Public Health – manages information systems for hygienic service.

Transplantations Coordinating Center (KST) – is an organisational unit of the state, established by the Czech Ministry of Health. Its tasks are management of specialized registries, coordination of procurement and transplantations and gathering, evaluation and processing of data on performed procurements and transplantations as well as on their results.

General Health Insurance Company (VZP) – is the biggest Health Insurance Company and plays a key role in the field of financial reporting of healthcare facilities.

Other health insurance companies adopt protocols and formats for financial reporting developed by VZP.

eHealth activities in the Czech healthcare gain importance very slowly. Contemporary data exchange is limited to obligatory tasks defined by law (i.e. financial reporting, contribute by statistical reports to the NHIS or sending data to various registries). Reusing or sharing medical records has either a local character or it is not implemented at all.

2.5.2 Making Use of International Standards

The use of international standards such as HL7 v2, HL7 v3, EN13606 or DICOM [27] is induced mainly by the local requirements within healthcare institutions to communicate with modern instruments and modalities. Here, the major role is played by the HL7 version 2 and DICOM standards; however, it represents only a minor part of the overall communication within the Czech healthcare system.

2.5.3 IZIP

One of the well-known eHealth activities in the Czech Republic is the IZIP project [28]. This project is involved in the epSOS project (Smart Open Services for European Patients) [29].

The main idea behind IZIP is that the citizen should have access to a centralized copy of his healthcare documents (so-called ”Internet Health Book”) and he/she should be able to allow access to this information repository to his/her GP or other caring physicians. The Internet Health Book contains information about patient intended for emergency use, so-called ”emergency set”, summary of patient’s medication, allergies and a retrospective financial overview of medical examinations supplied to the patient.

Although there are still some issues regarding security, origin, completeness and opportunities of publishing medical reports, the system received several awards at the national and international level [30].

One of the main contributions of this project is the establishment of some sort of semantic interoperability realized by humans. The main obstacle in achieving computable semantic interoperability [31], where the exchange of data is unambiguously established among machines, was the orientation of the IZIP mainly on the data in form of text documents.

2.5.4 Information Technologies for the Development of Continuous Shared Healthcare

A national research project of the ”Information Society” programme of the Academy of Sciences of the Czech Republic called ”Information Technologies for the Development of Continuous Shared Healthcare” (ITDCSH) [32] (2004-2008) had among its main goals the creation of an interoperability platform for structured healthcare data exchange, serving as a basis for lifelong healthcare support, which would be based on international communication standards. For this purpose the HL7 standard v3 was chosen from the set of HL7, DICOM [27], openEHR [33] and ENV 13606 [8].

This unique project (in the context of the Czech healthcare environment) served primarily as a demonstration of possibilities, tasks and issues. It was not possible to cover the whole area of medicine as an interoperability domain. The EuroMISE Center at the Institute of Computer Science (ICS) AS CR has a long history of interdisciplinary research oriented on the field of cardiology. Since the ICS AS CR was the main contractor of the project the cardiology was chosen as the medical domain for the pilot realisation of the semantic interoperability platform.

The proposal of the Semantic Interoperability Platform based on HL7 v3 messaging standard will be described in results chapter in Section 7.5.

Chapter 3

Obstacles in Sharing Medical Data

Obstacles can occur on various levels of communication. Since the data in healthcare are stored naturally in a distributed manner whole healthcare system can be described by using open distributed processing – RM-ODP [34].

3.1 Obstacles

- *Different sets of collected values.*

  Description: The main reason of this issue is the big amount of HIS providers, who do not cooperate on the technological level and are thus unable to agree on common set of collected values. Another reason of this obstacle are hospital-specific conventions and habits in healthcare provision.

  Solution: The harmonisation of processes in hospitals might help as well as minimizing the interchangeability of data describing the healthcare provision with actual clinical data related to patient's health status.

- *Lack of detailed semantic description of collected values.*

  Description: Collected values describe basic anthropometric information about patient, medication list, allergies, results of laboratory exams but nothing more. The rest of patient's records is usually a free text because no suitable set of values has yet been agreed on.

Solution: Agreement on a higher level (ministry, university hospitals) about which data to collect, consequently to be enriched by semantics description.

- *Using isolated coding systems to describe semantics.*

  Description: Commonly used standard for healthcare data interchange is DASTA. In this standard the National Coding List of Laboratory Items is used.

  Solution: Despite the existence of international coding systems like SNOMED CT, the main obstacle is the non-existing translation into Czech. Even after translation of international classifications into Czech, their full adoption will require their acceptance by software vendors and groups of experts.

- *Natural language is being used in the healthcare documentation.*

  Description: More convenient for a user is to express the state of patient's health in free text form. The main motivation for such a habit is freedom in data entry.

  Solution: Repetitive discussions about the structure of collected data with physicians might improve adoption of structured forms instead of free text. Using vocabularies to describe individual terms may be another way of describing semantics of collected data that can be done directly by the user (physician).

- *Free text reports, i.e. lack of structured data.*

  Description: This obstacle is closely connected with the previous one and is mainly caused by adherence to historical practice.

  Solution: Gradual formalisation of given field based on agreement on usage of stable structures.

- *Restriction to reporting to insurance companies and creating statistical reports.*

  Description: Users of HISes naturally do only those activities that are mandatory. In order to get paid for their work they must create summaries of performed medical acts for insurance companies. The summaries for national statistical office are obliged by legislature. HISes do not support more sophisticated communication scenarios, e.g. sharing patient's data among regional healthcare facilities.

Solution: The greater expansion of sharing clinical data depends on health insurance companies. They should financially motivate GPs to share data prior to performing redundant examinations.

- *The lack of real motivation to exchange clinical data in structured form.*

  Description: Lack of motivation is caused by the whole healthcare system. eHealth solutions are still not deployed in real life, insurance companies do not motivate general practitioners (GPs) to reuse data, GPs do not trust foreign data sources, etc.

  Solution: The government should accomplish the eHealth activities to become a routine, concentrate on usage of ICT in healthcare with the intention of cutting costs and assign certification authorities aimed on medical equipment, clinics and their information systems.

- *Monolithic architecture of healthcare information systems.*

  Description: HISs available on the Czech market usually comprise modules communicating via messages in proprietary formats, or share common database, which makes such solution hardly extensible by an external module.

  Solution: The monolithic architecture of HISs can be eliminated by opening the Czech healthcare information systems market to foreign HIS vendors by government's encouragement of using international standards.

- *Political reasons.*

  Description: Political culture in the Czech Republic is still on a low level, therefore interest of the society is replaced by self-interest. This can be seen in creation of artificial obstacles in deploying new technologies, approaches or methods.

  Solution: The author is not aware of possibilities of rising political culture other than general elections. It will take a considerable amount of time until the Czech society will take public affairs seriously, thus voting for operational and reformative government.

3.2 Suggestions on Solving Problems in Clinical Data Exchange

Various obstacles could be solved only by adhering standards, others would need a change in legislature, but most of the obstacles could be fetched through by changes in the whole healthcare system.

The market still does not offer positions for IT experts with knowledge of biomedical domain. However, this is only a matter of time to change, therefore the education of such experts is inevitable.

Two possible approaches to communication in healthcare can be distinguished:

- communication via a mediator, i.e. transforming data to a common format (HL7 v3 messages, sets of openEHR archetypes etc.). This approach is suitable for a large number of communicating systems.

- peer to peer communication schema, i.e. each communication site transforms data into negotiated format processable by receiving system. This approach is suitable for a small amount of systems.

The research behind this thesis is aiming on robust solutions, therefore only the first approach is considered.

Chapter 4

Standards in Healthcare

Usage of standards in all fields of human activity brings valuable assets in form of development and production costs reduction, upgrade of the quality and reliability of products and services, higher level of compatibility, minimizing the versions of products and thus lowering the price.

The healthcare does not form any exception to the rules described above. Unless the standards are used in healthcare applications, we would end up with a Tower of Babel.

Communication among EHRs can be understood as data exchange in form of messages/documents with well-defined syntax that is supported by all participants. This ensures so-called syntactical interoperability where the structure and provenance of information or knowledge is understood by a clinical system – data are in machine readable form.

As stated on page 19 two ways of data exchange in healthcare can be identified – passive and active. From the standards' point of view the passive communication can be covered by the EN13606 with EHR extracts and HL7 v3 CDA with annotated documents. The active communication can be realized by using HL7 v2.x or v3 messages.

Despite the long term effort in the field of communication standardisation there still does not exist one universally accepted communication standard. There are two commonly used standards: HL7 v3 (international) and EN 13606 (European). The HL7 standard served as a basis for the solution described in Section 7.5 and EN 13606 has much in common with the proposal in Section 7.6 since it defines archetypes and EHR extracts for data exchange originating from openEHR Foundation.

4.1 EN 13606

This European standard deployed by CEN organisation was already introduced in Section 2.3.2 on page 23. According to the aim of this thesis, its Part 1 describing the refrence model is the relevant one.

The CEN Reference Model for EHR Communication is a generic model capable of representing the structure and context of part or all of the electronic health record of one subject of care, to support interoperable communications between systems and services that might request or provide EHR data. The information in health record is inherently hierarchical. The CEN Reference Model reflects this hierarchical structure and the main hierarchy components of this model are summarised in Table 4.1.

The EN 13606 Standard also defines UML Class diagrams for all classes of the reference model divided into packages. Figure 4.1 shows the content of the package called Extract. Here classes can be found representing the main hierarchy components from Table 4.1 like EHR extract, Folder, Composition, Section, Entry, Cluster and Element. Another packages are defined as well, but there are either too technical (e.g. Support or Primitive Data types) or out of scope of the clinical content (e.g. demographics)

The biggest positive quality of this standard is its existence. There is a eminent need of an accredited archetype based standard. The openEHR approach is only an open specification and many of its components could not be used in standards oriented solution. Therefore, the overlap of the openEHR specification to this standard is considered to be very useful. Negatively can be perceived its vagueness in the implementation field, thus it is very difficult to develop a functioning system conforming this standard. Lack of tooling support is illustrating this shortcoming.

4.2 HL7 Version 2.x

HL7's v2.x messaging standard is the workhorse of electronic data exchange in the clinical domain and arguably the most widely implemented standard for healthcare in

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.

Table 4.1: Main hierarchy components of the EN 13606 EHR Extract Reference Model. (Taken from [19].)

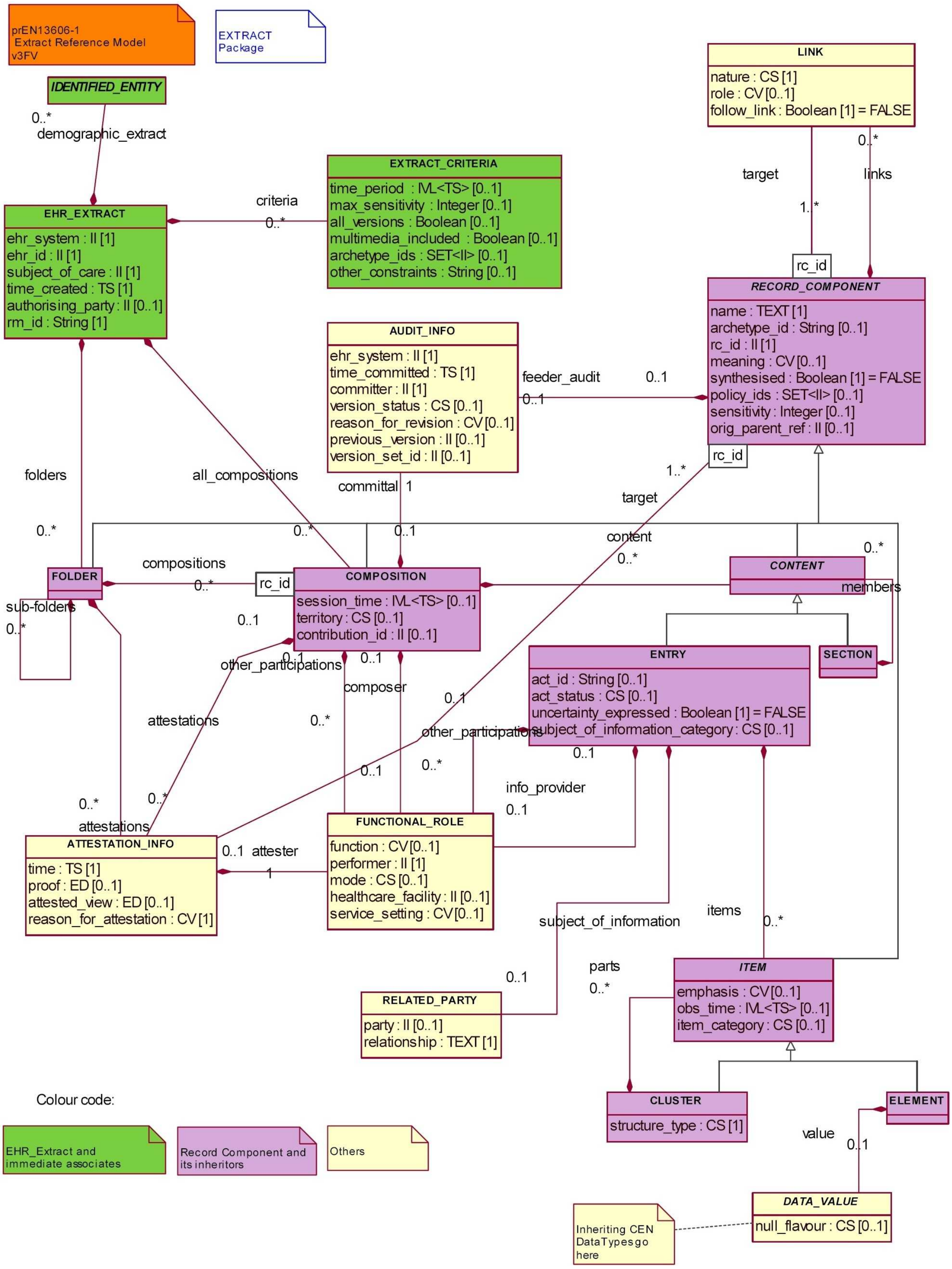


Figure 4.1: The class diagram of the Extract package of the EN 13606 Reference Model. (Taken from [19] CEN EN 13606 norm part 1)

the world. Thanks to its massive implementation in various medical equipment and modalities it is still under development, despite the newer version 3 exists. There have been already seven releases of the v2.x standard to date.

The HL7 v2.x messages were formerly defined as a "pipe"-separated values forming a stream of data. Since the version 2.3.1 it is possible to transfer the data in the XML format. This standard covers messages that exchange information in the general areas of:

- Patient Demographics
- Patient Charges and Accounting
- Patient Insurance and Guarantor
- Clinical Observations
- Encounters including Registration, Admission, Discharge and Transfer
- Orders for Clinical Service (Tests, Procedures, Pharmacy, Dietary and Supplies)
- Observation Reporting including Test Results
- The synchronisation of Master Files between systems
- Medical Records Document Management
- Scheduling of Patient Appointments and Resources
- Patient Referrals-Specifically messages for primary care referral
- Patient Care and problem-oriented records.

Thank to simplicity of this standard and its primary orientation on simple messages utilised mainly in medical devices it is widely used and supported. Various tools and frameworks exist that support HL7 v2.x messages creation. The main problem of the v2.x is the nonexistence of strict development methodology and all-embracing reference model. This ambiguity resulted in many various interpretations of this standard by various vendors. Achieving the semantic interoperability is impossible without adhering to a common reference model and follow a strict development methodology, which minimizes ambiguities and variations among different implementations.

4.3 HL7 Version 3

HL7 version 3 (HL7 v3) is an international communication standard developed and managed by ANSI-accredited Standards Developing Organisation (SDO) named HL7 (see p. pagerefhl7.soa). As opposed to the previous version v2.x the HL7 v3 standard defines a complete message development process [35] and means to defining the content of an EHR. One of them is a reference information model.

The Reference Information Model (RIM) [36] is used to express the information content for the collective work of the HL7 Working Group. It is the information model that encompasses the HL7 domain of interest as a whole. The RIM is a coherent, shared information model that is the source for the data content of all HL7 messages. As such, it provides consistent data and concept reuse across multiple information structures, including messages.

This information model consists of the following components: classes, their attributes, and relationships between the classes; data types for all attributes and vocabulary domains for coded attributes; state transition models for some classes. All HL7 information models are based upon the Unified modelling Language (UML), and may be adequately represented in a graphical form. The UML Class diagram of the RIM is depicted in Figure 4.2.

Because the RIM is too complex for a HL7 non-expert it is usual to explain the meaning of RIM classes on a simpler subset of RIM classes called ”backbone classes”.

Let us consider a situation when a physician is examining a patient. This would be modelled by means of RIM as follows: Two people (modelled as `Person` class, which is a descendant of `Entity`) one of which is playing the role of a patient (`Patient` class) and the other the role of a physician (`Employee` class). Both roles are participating (`Participation` class) on an act called medical examination (modelled as `Observation` class, which is specialisation of `Act` class). Roles can be connected via `Role Link` class,

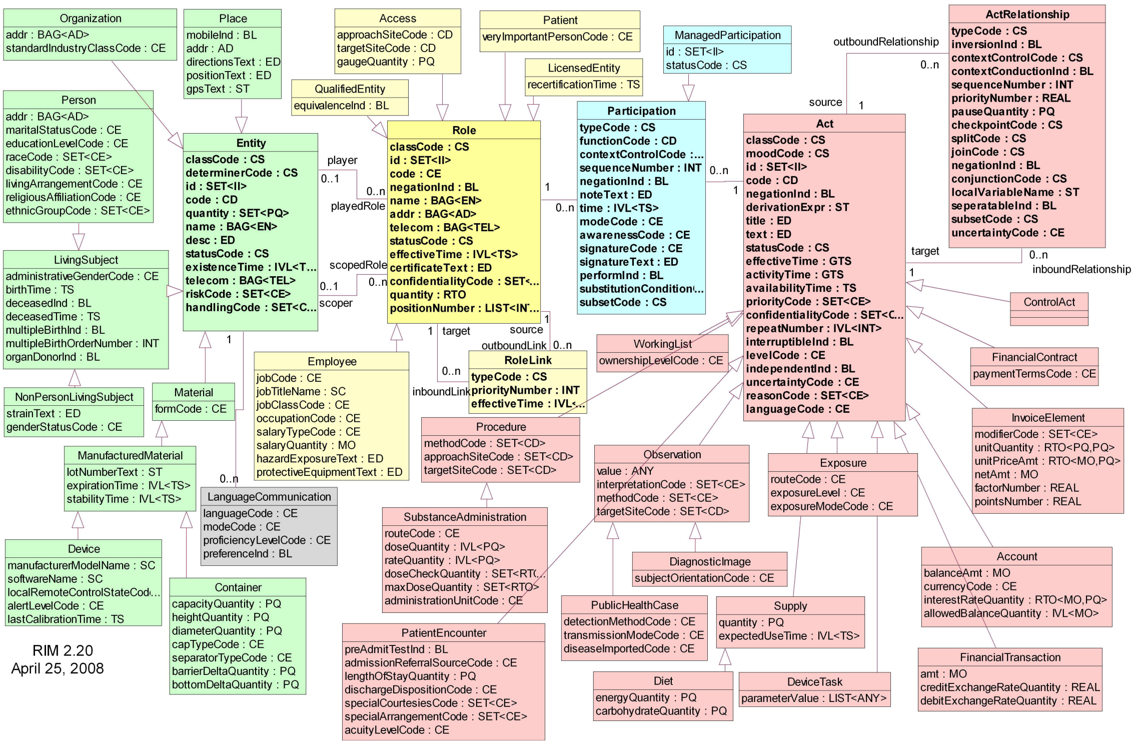


Figure 4.2: Reference Information Model defined in the HL7 v.3 standard.

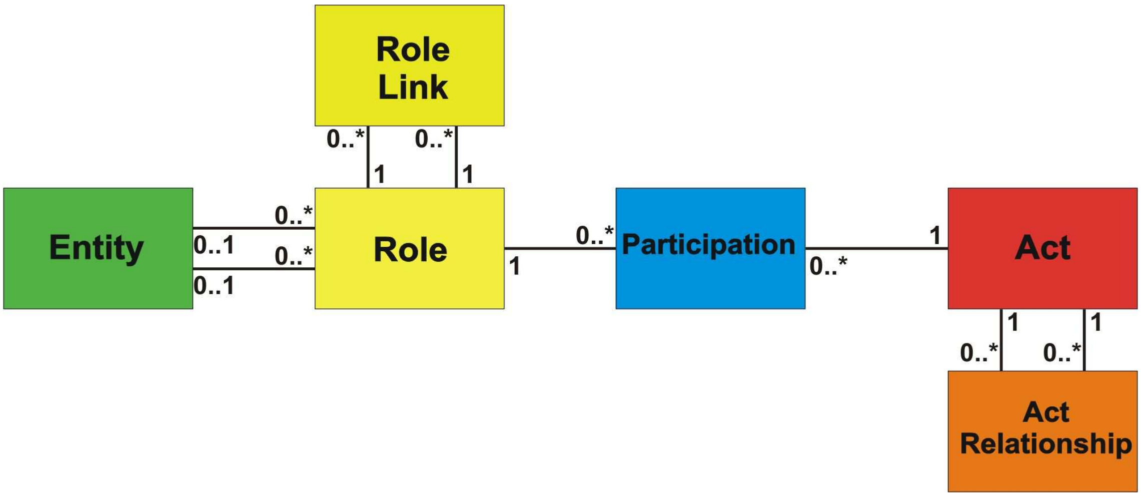


Figure 4.3: HL7 v3 Reference Information Model backbone classes.

similarly `Act`-derived classes can have interconnections modeled as instances of `Act Relationship` class.

The HL7 information modelling process recognizes three interrelated types of information models. The RIM was already mentioned above and the other two are D-MIM and R-MIM. A D-MIM (Domain Message Information Model) is a refined subset of the RIM that includes a set of class clones, attributes and relationships that can be used to create messages for a particular domain (a particular area of interest in healthcare). The D-MIM is used as a common base upon which all R-MIMs within a domain are built. The R-MIM (Refined Message Information Model) is a subset of a D-MIM that is used to express the information content for a message or set of messages with annotations and refinements that are message specific. The content of an R-MIM is drawn from the D-MIM for the specific domain in which the R-MIM is used. The R-MIM may include clones of selected classes with alias names specific to the perspective of the message(s) to be derived. The R-MIM represents the information content for one or more abstract message structures (HL7 v3 messages), also called Hierarchical Message Definitions (HMDs).

V3 messaging is an interoperability specification for transactions that are derived from the HL7 v3 models and vocabulary and define communications produced and received by computer systems. V3 messages include the concepts of message wrappers, sequential interactions, and model-based message payloads. These specifications are published as a collection of topics that describe the transaction interactions by domain [9]. A complete list of covered domains is part of balloted release of the HL7 v3 standard. Finally, a summarised description of the HL7 v3 message can be stated as follows.

HL7 v3 message consists of instances of R-MIM classes which are composed in a hierarchy defined by means of Hierarchical Message Definition(HMD).

4.4 openEHR Approach

Under the auspices of the openEHR Foundation (see p. 20) the openEHR Specification Project is being administered. This project (see [37] for more details) is responsible for developing the specifications on which the openEHR Health Computing Platform is based. The project deliverables include requirements, abstract specifications, implementation technology specifications (ITSs), computable expressions and conformance criteria.

The requirements the openEHR architecture is built on were captured over many years as part of numerous reasearch projects from around the world. Because the architecture is highly generic, and particularly due to being archetype-driven, it satisfies many requirements outside the original concept of the "clinical EHR".

The abstract specifications consist of the Reference Model (RM), the Service Model (SM) and Archetype Model (AM). The first two correspond to the ISO RM/ODP information and computational viewpoints respectively. The latter formalises the bridge between information models and knowledge resources.

The openEHR approach to modelling information, services and domain knowledge is based on a number of design principles, like ontological separation, separation of responsibilities and separation of viewpoints. More detailed information on these design principles can be found in [37].

At this point fundamental concepts of the openEHR approach will be defined to clarify their meaning in the following text.

Archetype according to [6] (the technical definition) is "a computable expression of a domain-level concept in the form of structured constraint statements, based on some reference information model". In [38] an endorsement to this definition can be found: "From a clinical point of view, archetypes serve an intuitive means to define, discuss and present clinical content."

openEHR template according to [39] it is a directly, locally usable definition which composes archetypes into a larger structure logically corresponding to a screen form. Templates may add further local constraints on the archetypes it mentions, including removing or mandating optional sections, and may define default values.

The term template is not reserved only for openEHR templates. In the HL7 v3 standard the templates are understood in the following way. **HL7 templates** are used to apply constraints on R-MIMs (refined message information models) generated from generic reference information model (HL7 v3 RIM) [40].

4.4.1 2-level Modelling

openEHR Foundation defined an alternative approach of application modelling called **two level methodology** [41]. The single-level approach is a commonly used information system development methodology (e.g. the object-oriented methodology). The resulting model encodes only the requirements found during the current development, along with best guesses about future ones. The implementation of future requirements, demanded by target domain (medicine is a rich in changes field), becomes a complicated task. This was overcame by the two level methodology that separated the semantics of information and knowledge into two levels of model.

Figure 4.4 illustrates an overview of the class structure of the EHR Information Model, along with the main concepts on which it relies, namely Data Types, Data Structures, Archetypes, and Identification. The reference models forms a 1st level of the 2-level modelling approach. The fundamental concepts of the 2nd level are archetypes and templates.

Archetypes play a key role in development of future proof EHR systems [41]. Archetypes are structured constraint statements based on some reference model.

4.4.2 Archetype Definition Language

Archetype Definition Language (ADL) is a formal language for expressing archetypes. ADL uses further two syntaxes, cADL (constraint form of ADL) and dADL (data definition form of ADL) to describe constraints on data that are instances of the information model [42].

The archetype binding to a reference model is realized in archetype formal definition that is formalized in ADL particularly in the part called **definition** (see Figure 4.5). This binding is represented in formal language – cADL. This language enables constraints on data defined by object-oriented information models to be expressed in archetypes or other knowledge definition formalisms. It is most useful for defining the specific allowable constructions of data whose instances conform to very general object models.

The dADL syntax provides a formal means of expressing instance data based on an

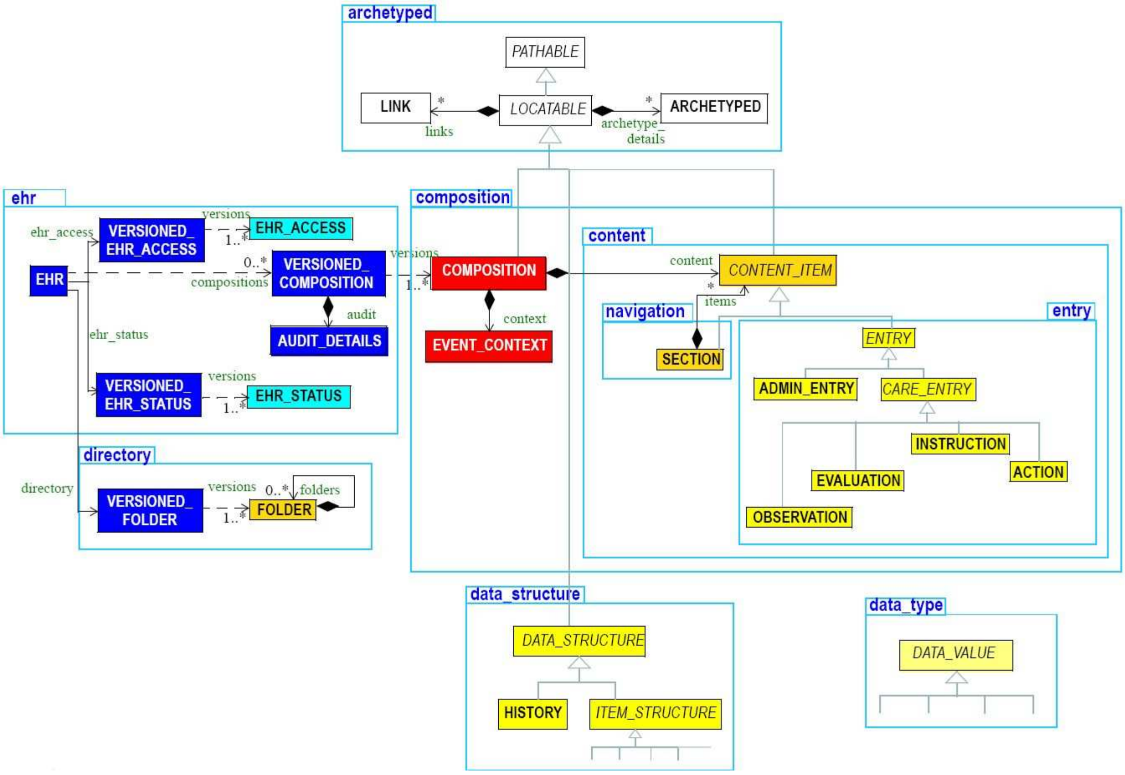


Figure 4.4: openEHR EHR Information Model Overview. (Taken from [43] EHR Information Model

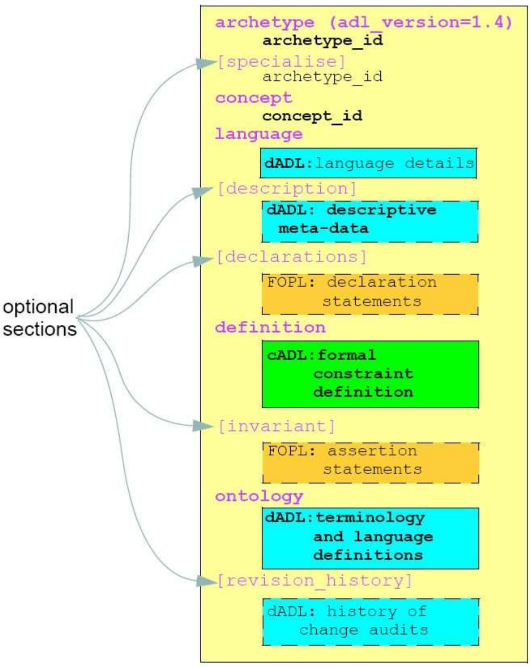

optional sections
archetype (adl_version=1.4)
archetype_id
[specialise]
archetype_id
concept
concept_id
language
dADL:language details
[description]
dADL: descriptive meta-data
[declarations]
FOPL: declaration statements
definition
cADL:formal constraint definition
[invariant]
FOPL: assertion statements
ontology
dADL:terminology and language definitions
[revision_history]
dADL: history of change audits

Figure 4.5: ADL Archetype Structure [42]

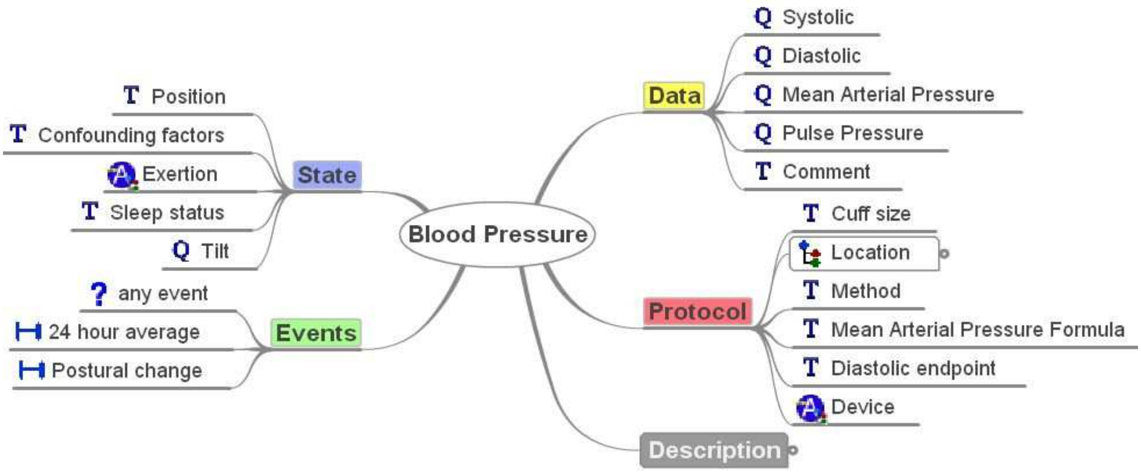


Figure 4.6: Blood pressure archetype example.

underlying information model, which is readable both by humans and machines. This language has many similarities with the XML, but orients on improving the readability by humans. The XML should by also readable by humans, however, realistic examples of XML are too complex for people to read. More details can be found in [42].

The openEHR Foundation presents an application called Clinical Knowledge Manager (CKM) accessible from web site [44]. Its purpose is to concentrate archetypes in one place in order to be reviewed by the community and create a repository of archetypes that could serve as a basis for development of EHRs with Harmonised Clinical Content (HCC).

In Figure 4.6, the structure of archetype representing blood pressure concept is depicted. The part **data** contains values of the actual pressure, i.e. systolic, diastolic, mean arterial pressure, pulse pressure and textual comment on blood pressure reading. **State** is a list of information describing conditions of the measurement, e.g. the position of the patient at the time of measuring blood pressure. **History** covers separate measurements and adds temporal data in the implicit form, i.e. base measurement in the history, another reading after 5 minutes of rest, 10 minutes etc. Finally, the **protocol** holds technical data such as size of a sphygmomanometer's cuff if it is used or a specification of an instrument used to measure the blood pressure.

Archetypes behave neutrally to the used ref. model, therefore usage of different ref. model is not difficult and transformation from one to another has to be considered [20].

4.5 Healthcare Related Terminologies and Coding Systems

Coding systems and terminologies are very useful especially in a such rich on details field as healthcare. Enumeration of concepts' semantics isolates in a certain way the knowledge from the implemented system. Without coding systems all items listed in it would be modeled directly as a concept of the application model. This would certainly cause the growth of the model to overwhelming dimensions.

Therefore, imagine a situation where a part of an EHR should be modeled describing possible cardio-vascular risk diagnoses of a patient. One way of modelling this situation would consist of creating n concepts of type boolean, each describing the presence of certain diagnose. The other way using coding system would involve creating one list of variables of type codelist's values covering the possible cardio-vascular risk diagnoses. Such a list does not have a restricted count of items nor does it have specified an fixed count of them. Moreover, if a new diagnose is added to the coding system, the second solution does not involve the model change, whereas the first solution requires adding a new boolean diagnoses concept to the model. A solution based on a list of string values was omitted intentionally as the worst one, since it enables user to enter any string without any posibility (for the system) of checking its consistency and relevance.

4.5.1 LOINC

Logical Observation Identifiers Names and Codes (LOINC) is one of the most used code-base systems in medical practise today. Its purpose is to facilitate the exchange and pooling of clinical results for clinical care, outcomes management, and research by providing a set of universal codes and names to identify laboratory and other clinical observations.

Each LOINC record corresponds to a single test result or panel. The record includes fields for specifying:

1. Component (analyte) - e.g., potassium, hemoglobin, hepatitis C antigen.

2. Property measured - e.g., a mass concentration, enzyme activity (catalytic rate).
3. Timing - i.e., whether the measurement is an observation at a moment of time, or an observation integrated over an extended duration of time - e.g., 24-hour urine.
4. The type of sample - e.g., urine, blood.
5. The type of scale - e.g., whether the measurement is quantitative (a true measurement) ordinal (a ranked set of options), nominal (e.g., E. coli; Staphylococcus aureus), or narrative (e.g., dictation results from x-rays).
6. Where relevant, the method used to produce the result or other observation.

The main data file contains among fields mentioned above a column named `LOINC_NUM` which was used to encode MDMC concepts in LOINC. The reason of using this coding system as the preferred one was, that it contained a significant subset of concepts modeled by MDMC and also was used in LIM models (see Section 7.1.2) for its preference by the HL7 v3 standard.

The LOINC is distributed in the form of a table stored in a text file (tab-separated values) available from web pages [45].

4.5.2 SNOMED CT

The Systematised Nomenclature of Medicine – Clinical Terms (SNOMED CT) aims to be a comprehensive clinical terminology that provides clinical content and expressivity for clinical documentation and reporting. Another effects of SNOMED CT on healthcare are it becomes essential for EHRs, its content is scientifically validated, it is able to cross-map to other international standards and is already used in more than 50 countries [12].

This nomenclature is one of the largest in the field of medicine – approx. 311000 of active concepts (Jan. 2008) and over 1 million connections. SNOMED CT concepts are organized in hierarchies with multiple levels of granularity. The broad coverage of topics included in SNOMED CT is illustrated by the following list: Clinical finding/disorder, Procedure/intervention, Observable entity, Body structure, Organism, Substance, Pharmaceutical/biologic product, Specimen, Special concept, Physical object, Physical force, Event, Environment or geographical location, Social context, Staging and scales.

The main data in each SNOMED release are stored in so-called Core Tables. These consist of three main tables: **Concepts**, **Descriptions** and **Relationships**. According to their names, these tables consist of unique concepts, a list of descriptive names for each concept and a range of relationships with other concepts in the terminology. Each release also contains so-called Cross Mappings files that contain mappings of the SNOMED codes with other coding schemes like ICD-10 or ICD-9CM.

In this thesis mainly the Concepts table was utilised in the mapping of MDMC concepts with SNOMED CT concepts. The Concepts table comprises following columns: `CONCEPTID`, `CONCEPTSTATUS`, `FULLYSPECIFIEDNAME`, `CTV3ID`, `SNOMEDID` and `ISPRIMITIVE`. The column ConceptID was used as a key identifier of a SNOMED concept in the mapping process. Experience and results of this mapping can be found in Section 7.1.1 and tables with mapped concepts in appendix B on page 134.

Chapter 5

Electronic Health Record

In this chapter EHR will be defined in more precise way and some facts about EHRs will be summarised. Afterwards, some EHR-Ss will be described, all of which were part of the research.

Electronic Health Record (EHR) is defined in [6] as "a repository of information regarding the health of a subject of care in computer processable form, stored and transmitted securely, and accessible by multiple authorised users. It has a commonly agreed logical information model which is independent of EHR systems. Its primary purpose is the support of continuing, efficient and quality integrated health care and it contains information which is retrospective, concurrent and prospective."

EHR system (EHR-S) is defined as "a system for recording, retrieving, and manipulating information in electronic health records" [26].

5.1 Possible Content Stored in an EHR

The following text describes an example of what concept groups may appear in an EHR as shown in [46].

1. A collection of concepts that together form fixed attributes of a higher level concept that is not recorded as its component parts alone - e.g.:

- a blood pressure measurement with its two pressure measurements, patient position, cuff size etc.,

- a body weight with details about the baby's state of undress and the device used for measurement,

2. A generic concept (with other fixed attributes) that is a value or a collection of values which form a subset of a larger (or very large) known set - e.g.:

- a diagnosis - the value - with fixed attributes such as the date of onset, the stage of the disease etc.,

- a laboratory battery result which includes an arbitrary set of values - the collection - with fixed attributes such as the time of sampling, or a challenge applied to the patient at the time the sample was taken (e.g. fasting),

3. A collection of these higher level concepts that are usually measured together and might be considered themselves concepts - e.g.:

- vital signs – with temperature, blood pressure, pulse and respiratory rate,

- physical examination – with for example observation, palpation and auscultation (and other findings),

4. A collection of these aggregations which might form a record composition or a document - e.g.:

- a clinic progress note containing symptoms, physical examination, an assessment and a plan,

- a laboratory report that contains the results as well as interpretation and details about any notifications and referrals that have been made,

- an operation report detailing the participants and their roles, a description of the operation, any complications and follow-up monitoring and care required.

Content of the EHR directly influences the structure of the stored data and thus inevitably the process of data workflows.

5.2 Heterogeneous EHR Systems

The problem in sharing medical data is the possibility of different definitions of concepts despite using the same modelling methodology (the term modelling methodology comprises all steps necessary to create the resulting model).

Heterogeneity in models occurs when there is a disagreement about the meaning, interpretation or intended use of the same or related data. Usually two separated individuals (experts, developers etc.) model the same domain in more or less different way, even when using the same methodology.

The similarity or heterogeneity of the models can be considered on two levels. The first one is the functional (implementational) level where information systems use different network protocols (e.g. IP - Internet Protocol), transport binding (e.g. HTTP, FTP) or message format (e.g. XML, ASCII text) for communication. The second level is the semantical one where systems have to understand each other's formal definitions of domain concepts. The latter level is the one of the concerns of this thesis.

5.3 Real EHR Systems

In this section some EHR systems (EHR-s) important for the author's research will be mentioned. The MUDR EHR served as a modelling and tooling basis for the dental knowledge base (see p. 60). It was also utilised as a data source for the testing of the HL7 v3 based semantic interoperability platform (SIP) proposed in Section 7.5. The MUDRLite EHR was the environment for the developed EHR for dentistry. The WinMedicalc 2000 was another data source for SIP testing, representing the commercial EHR solution.

5.3.1 MUDR

The development of the electronic health record at the EuroMISE Centre, Institute of Computer Science AS CR started in the year 2000 based on inspirations and experiences from existing CEN/TC251 standards and several European projects, mostly the I4C and TripleC projects [47]. The main requirement for the proposed system was the

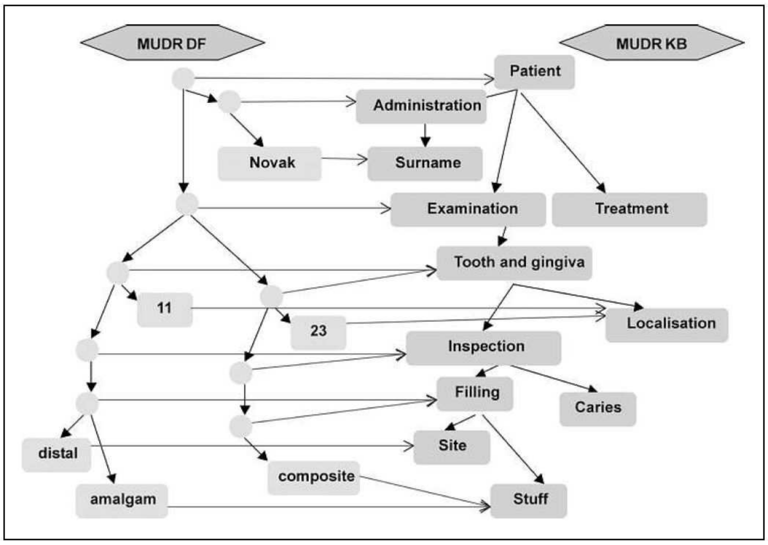


Figure 5.1: Example of MUDR EHR data and knowledge graph structures in dental medicine.

structured way of data storage combined with free text with possibility of dynamic extension and modification of the set of collected attributes without any change of the database structure. The main goal of the research in this field was to suggest common general principles to increase the quality of EHR systems, to simplify data sharing and data migration among various EHR systems and to help to overcome the classical freetext-based information stored in medical records. The suggested solutions were implemented in a pilot application named "MUltimedia Distributed Record" (MUDR) [48], [49].

The design of MUDR EHR [50] was based on a 3-layer architecture – the database layer, the application layer and the user interface layer. Because of the requirement of a dynamically extensible and modifiable set of collected attributes, it was complicated to use a classical relational database structure with columns corresponding to the gathered features as the basis of information storage. The solution is based on two main structures described by tools of graph theory. The example of data and knowledge structures in dental medicine is depicted in Figure 5.1.

The set of collected attributes and relations among them are stored in a directed graph structure called a knowledge base. The vertices of the graph describe the collected attributes by their unique id, internal name, physical data type and other auxiliary information; the graph edges describe the relations among attributes. The dominant edge of type "inferior" exists in the graph. This edge defines the hierarchical tree structure of the knowledge base, so that the knowledge base can be described by directed forest with a few trees. These trees are also called "knowledge base domains". Another hierarchical graph structure named "data-files" is used to store the patient data itself. Each tree in the graph describes the data of one patient. Each vertex in the tree describes one instance of the medical concept from the knowledge base by the identification of the concept (internal name of the vertex), its value (with the possibility to specify the range of values), date and time of examination and other administrative data. The values are physically stored in separate tables according to the physical data type.

The application layer consists of four basic components – the HTTP server used for communication with client applications, the EHR-AppL service implementing the main application logic, the CGI-script (potentially more of them) serving as an interface between HTTP server and the EHRAppL service and possibly medical guidelines modules. The application layer realizes the set of functions provided to the client application and realizes the functionality of the EHR system on a more abstract level by isolating clients from database implementation details. This solution enables the change of the used database in the future without influencing client applications and helps to achieve a higher level of safety and security. The communication between clients and the application layer is done by a secure HTTPS protocol (Hypertext Transfer Protocol). The HTTP server is therefore a necessary part of the application layer. This approach opens the possibility to create special CGI-scripts for communication with lightweight clients like web or wap browsers, enabling the use of mobile devices to connect to the EHR. An important attribute of the EHR, that is improving the effectiveness and quality of physician's work during data entry, is a decision support capability. Application layer of MUDR supports the decision support functionality by modules implementing algorithms, for example algorithms described in medical guidelines for the management of hypertension [51].

When designing a system, two types of its usage should be taken into account – consultation and data entry. Consultation requires minimal search time, an inspectional way of information presentation, problem-oriented grouping of findings and patient visits. Data entry requires the maximal ease and speed of process of entering the data into the system. The development of the user interface of MUDR EHR focuses on both parts – data entry and consultation. The client application implements the functionality of structured data entry combined with free text, including the tools for the formalisation of free text. The structured data can be entered either directly by selecting the appropriate items from the tree structure of the knowledge base or by using dynamically created forms. The printable reports and user entry forms are created dynamically following the definitions in XML documents. The consistence of the entered data can be checked by the mechanism, controlled by the set of consistency rules. The application utilizes the guideline module part of the application layer of the MUDR EHR and shows either the text of selected guideline or starts the process of consultation of the selected patient's data with the guideline module.

Practical experiences from the evaluation of the MUDR system showed that the core of the system is well prepared to serve as a dedicated application server allowing multiple clients to connect and manipulate stored data. However, the implementation of this system in a resource-limited office of a typical general practitioner is very demanding and difficult. Therefore, a new system with simplified data storage and an enhanced user interface was developed. Because the new system was derived from the EHR MUDR, it was named the MUDRLite.

5.3.2 MUDRLite

The MUDRLite architecture is based on two tiers. The first one is a relational database (e.g. MS SQL) and the second one is a MUDRLite User Interface layer (MUDRLite UI). The database schema corresponds to particular needs and therefore varies in different environments, as opposed to the fixed database schema in the MUDR data layer. Thus, a basic way to share the healthcare data is to use this client-server architecture, install more user interfaces and access the data through a common database.

The core of MUDRLite – MUDRLite Interpreter – is able to handle various database schemas. This feature often simplifies the way of importing old data stored in other databases or files. The visual aspects as well as the behavior of the MUDRLite UI are completely described by an XML configuration file. The end-user can see a set of forms with various controls placed on them by appropriate XML elements. MUDRLite operates as a kind of commands' interpreter; it processes the instructions encoded in MLL language as described in [17] and manipulates the database layer as well as the visual aspects of the MUDRLite UI.

As the set of predefined controls is limited, MUDRLite provides interfaces to include user-defined controls and components. These interfaces can be used to offer graphically and functionally advanced components as well as new features, e.g. an advanced security policy, integration with other existing information systems, or sharing of electronic health data based on standardized EHR communications. If such a user-defined component is trusted, it gets the access to the structured EHR. However, this access can be prohibited or limited by the MUDRLite Interpreter for security reasons in accordance to a defined security policy. Anyway, the MUDRLite Interpreter may be able to monitor the access and create a record concerning all the read/write actions. By virtue of the interfaces, a trusted component can access the data in the EHR and thus it serves as a kind of "intelligent proxy" implementing a standardized EHR communication.

5.3.3 WinMedicalc 2000

The WinMedicalc 2000 (WMC2000) is a full featured HIS and for the purpose of our project we limited our interest on its EHR part. The WMC2000 stores its data in a relational database and thus uses E-R model [52] to represent its information model.

The model of WMC2000 system consists of basic administrative information, cardiological examinations (e.g. ECG examination, Holter monitor, stress test ECG etc.), lab examination, physical examination and family history. Each of these data (except admin. information) are connected with a clinical event, that binds together the object and subject of the event, i.e. the patient and the physician. Clinical event also contains information on the place of the event (e.g. ward, emergency room). Moreover, WMC2000 system covers a broader scope than just clinical data (e.g. catering, bed management), but these are out of our concern so we leave them out.

Chapter 6

Clinical Content Modelling

The successful acceptance of healthcare information systems (HcIS) by its users depends mainly on the ability of these systems to provide high-quality patient's clinical data at the point of care. The properly modelled clinical content implemented in HcIS supports not only the decisions made during the care provision, but also guidelines, which help professionals to cope with rare situations, make a huge profit of data with precisely defined semantics. Unambiguously and accurately defined clinical content is one of cornerstones of high-quality and save healthcare, therefore research of this area is of enormous importance.

In this chapter the term clinical content (CC) will be explained and some clinical models describing the CC in cardiology and dentistry will be introduced. Since there are various approaches to data modelling, some tools utilised in this thesis's research are presented at the end of this chapter.

6.1 Clinical Content

In the scientific literature many definitions and explanations of the term clinical content can be found. To illustrate this term we mention some of these explanations.

According to the study [1] prepared by Deloitte , "the clinical content refers to the substance of data and information utilized by an advanced clinical information systems".

For the purposes of this thesis, the term clinical content is defined in this way:

Clinical content is a part of set of concepts that underline the EHR and refer to medical domain concepts such as physical examination, laboratory, medication, rather than demographic information, billing or bed management.

Rong Chen is referring to clinical content in EHR system in the article [53] as follows: "Currently deployed EHR systems with large user bases have their own proprietary way of representing clinical content using various models. Semantic interoperability between different EHR systems across technical platforms and organisational boundaries requires a consistent way of sharing the syntactic and semantic definitions of clinical content. Clinical content models represented in a common format can speed up the installation and configuration of EHR systems and facilitate the progression of nationally, or even internationally, standardised representation of clinical content."

According to [38] the clinical content can be formally defined by clinicians via openEHR archetypes.

Clinical content harmonisation is either a specification of the clinical content using one particular formalism, or a mapping consolidating clinical content already described using two or more different formalisms.

The decision which formalism should be the representation of CC based on is crucial. The most likely scenario of the CC harmonisation will include the mapping creation process since some parts of CC already exist in formalised form. If the CC will be modelled by means of archetypes, following ideas found in cited works might be exploited.

According to SFI Archetype Track [54] "the structured clinical content material is a good point of departure for modelling archetypes. However, this will take time and require considerable work since the level of detail in the clinical content material is insufficient for modelling archetypes. The clinical content material was created for very specific purposes and thus differs from that which has to be expressed in an archetype. Clinical content can be seen as minimum data sets (proper subsets), while archetypes are maximum data sets (complete sets). According to openEHR archetype principles, clinical content should be expressed at template level. It was relatively easy to enter clinical content just as it appears in the Capital Region's forms into the archetype tool, thereby obtaining 'flat archetypes' that are directly usable electronic expressions of the clinical content from which they were modelled. Clinical content entered into the archetype editor in this way is a kind of template rather than an archetype. It represents minimum data sets for specific purposes and can therefore not be reused in other contexts. The flat archetypes represent a more structured specification basis for IT systems than MS Word documents and are presumably easier to maintain. Considerable consistency can be achieved across a library and systems. The archetype editor is thus suitable for creating 'portable clinical content', with the important proviso, however, that it is not possible to include user interface information and business rules that may need to be documented with the clinical content. Flat archetypes have little or no reuse value across an archetype library."

The article [55] by Heather Leslie contributes to the clinical content discussion as follows: "It is essential that clinicians are able to contribute to the development of clinical content for electronic health records. Clinicians can and do engage in the clinical content authoring, reviewing and maintenance activities – participating in the development of standardized clinical content models, known as archetypes, that are key contributions to communication and hence quality patient care. As their purpose is to be computable, unambiguous and consistently implementable, they are not by nature human-friendly. The openEHR clinical content models are known as archetypes and templates. Archetypes are the foundation building blocks at the clinical concept level; templates aggregate and constrain the archetypes to create context-specific clinical content for use in direct patient care. Is it really achievable to develop archetypes for all clinical content? It is estimated that the number of archetypes required as the foundation for an EHR would be in the order of 2,000 archetypes, with as few as 10 archetypes being enough to create a simple shared emergency summary."

6.2 Minimal Data Model for Cardiology

A set of important medical concepts in the field of cardiology named Minimal Data Model for Cardiology (MDMC) [56] was prepared by the representatives of the Czech Society of Cardiology and statisticians specialized in medical data processing. This set of concepts served as a basis for information models of EHR systems used in this thesis.

The main purpose of this model was to support data collection in the outpatient department of preventive cardiology. It consists of following parts:

- Demographic data – name, surname, address, telephone number, e-mail, sex, date of birth, code of patient's health insurance company.

- Family anamnesis – information about patient's mother, father, brothers and sisters.

- Social anamnesis and toximania – marital status, education level, level of stress, physical load in job, leisure physical activity, tobacco smoking behavior, alcohol intake.

- Allergies – allergy manifestation, drug allergy.

- Personal anamnesis – list of patient's diseases like diabetes, essential hypertension, hyperlipoproteinemia, ischemic heart disease, cerebrovascular disease, ischemic peripheral artery disease and aneurysm of aorta; other relevant diseases, menopause, date and cause of death.

- Actual difficulties probably with cardio-vascular background – dyspnoea, dull chest pain, palpitation, swelling of edema, syncope, cough, haemoptysis, claudication.

- Treatment – dietary treatment, medicamentous treatment (dosage, active substance content, etc.).

- Physical examination – weight, height, body temperature, hips size, waist size, BMI, WHR, systolic and diastolic blood pressure (left and right arm separately), pulse rate, respiratory rate, pathological findings (carotids, peripheral arteries, on heart when physically examined, lungs when physically examined), other pathological finding.

- Laboratory examination – blood sugar, uric acid, total cholesterol, High density lipoprotein (HDL) cholesterol, Low density lipoprotein (LDL) cholesterol, triacygglycerols.

- ECG – characteristics of the ECG curve (rhythm, frequency, PQ interval, QRS complex feature), ECG findings, overall ECG description.

First implementation of the MDMC was in the application ADAMEK [57]. Data collection started in 2002 in two outpatient departments – one in Prague located at the Institute of Computer Science AS CR and the other one at the Municipal Hospital in Caslav.

6.3 Dental Knowledge Base in the MUDR EHR

MUDR (MUltimedia Distributed Record) is a pilot application of EHR developed at the EuroMISE center. In this system the data and the concepts that describe them are stored separately. MUDR EHR was described in more detail on page 51.

In the years 2004 and 2005 the dental knowledge base was created. Dentists and computer scientists from the joint workplace of the EuroMISE Center produced a model in the form of a knowledge tree as a part of the knowledge base of the MUDR EHR. The knowledge tree comprised basic information about a patient, family history, social history, personal history, information about medication and exhaustive information about patient's oral cavity status from the dentist's point of view.

The dentistry section of this knowledge base could be divided into two main parts: examination and treatment of teeth, soft and hard tissues in oral cavity.

Under examination section the following details are recognised:

- inspection of the tooth – sound tooth, caries, filling, endodontic examination, mechanical attrition, erosion of the tooth from various reasons (e.g. due to overproduction of gastric acid), odontogenic resorption, anomalies of the tooth (e.g the size of the tooth) and prostodoncy information.

- examination of vitality – observing the reaction of the tooth on iritation by coldness, heat and electricity, meassuring the tooth pain threshold values by vitalitester.

- examination by percussion of the tooth – recording the behaviour of the tooth on vertical and horizontal pat.
- parodontal examination – PBI index, CPI index, tooth mobility, meassurement of gingiva recession and depth periodontal pockets in milimeters.
- surgery – information on dentoalveolar surgery procedures (osteitis in tooth area, dentitio difficilis and others).
- trauma – injury of hard oral cavity tissues, dental pulp, periodontium, dental alveolus and gingiva.

The next part dealing with treatment comprises following concepts:

- filling – size, position and matierial of the filling.
- endodontic treatment – information on location and count of root canal(s) afflicted with infection, method used to clean and liquid used to rinse the root canal, materials used for root canal filling.
- prostodoncy – description of fixed (crown, pontic or implant), removable and special types of dental prostheses, impression material.
- primary dentition treatment – apexification and protective crown for immature tooth.
- periodontology – removal of supragingival tartar, topical fluoridisation, chlorhexidine application.
- preventive dentistry – methods of preventive dentistry to prevent caries.
- surgery – dentoalveolar surgery procedures applied on one tooth (e.g. extraction, resection or excision).
- trauma – treatment of injured hard oral cavity tissues, dental pulp, periodontium, dental alveolus or gingiva.

- local anestezia – details of local anaesthesia used during tooth treatment (e.g. way of application anaesthetic, type of used anaesthetic – supracain, marcain, etc.)

Approximately 1000 dental concepts were structured in the dental knowledge base. The knowledge tree served as a modelling basis for development of an EHR system for dentistry described in the results section on page 75. In the frame of the project Center of Biomedical Informatics [58] this knowledge base was extended by concepts describing temporomandibular joint disorders. This extension is one of the results described in Chapter 7 (see p. 78).

6.4 Modelling Tools

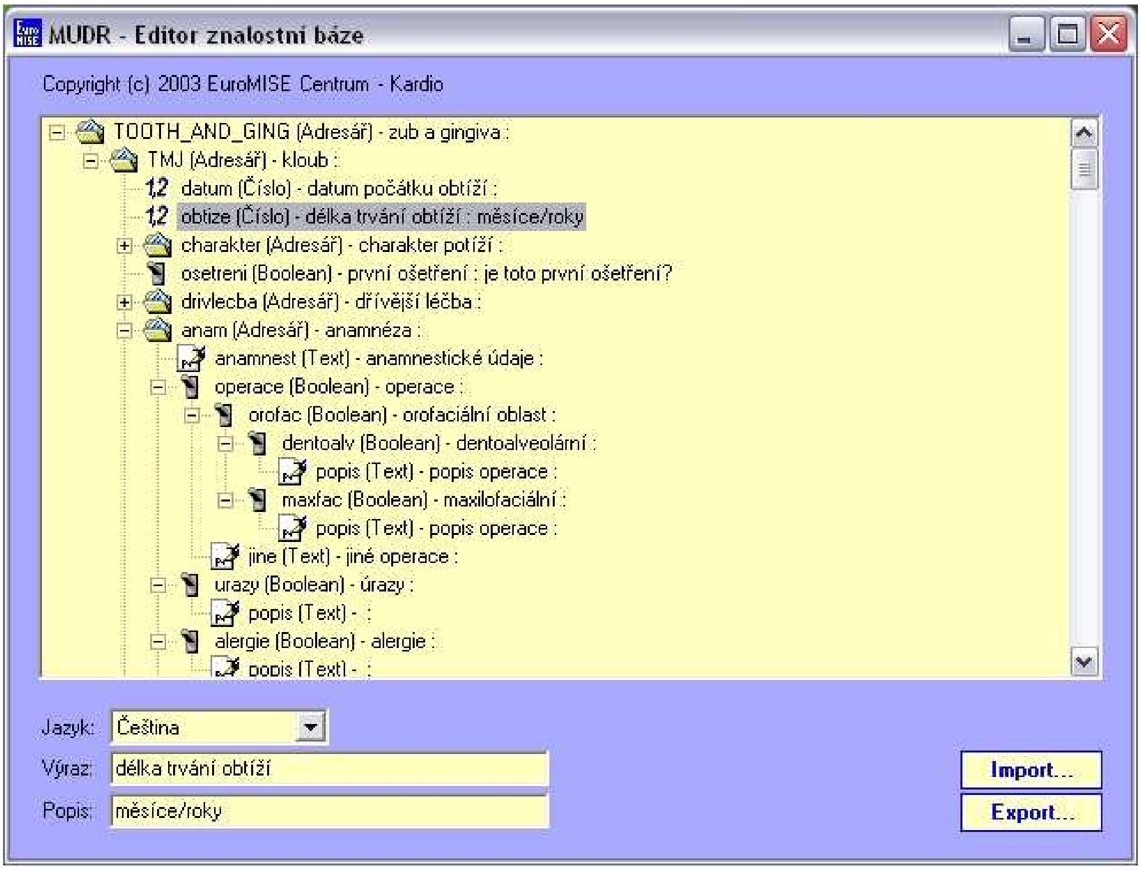
In this section some modelling tools that were used in the research of this thesis will be described.

6.4.1 MUDR Knowledge Base Editor

A supporting tool for knowledge base trees creation is called MUDR Knowledge Base (MUDR KB) Editor. It was developed by Josef Spidlen as part of his PhD thesis [16]. This editor is a .NET application capable of creating a knowledge tree from scratch, importing an existing tree in XML format and also exporting a tree in XML file.

The MUDR KB supports all features of knowledge nodes defined in Section 2.2.3 and 5.3.1. It covers the manipulation with nodes in a tree (creating a new root node, new descendant, remove a node), it supports the same data types of knowledge nodes as MUDR EHR does (BOOLEAN, TEXT, NUMBER, PICTURE, AUDIO, VIDEO, LONGTEXT, DATA_REF, KN_REF) and finaly the import and export of knowledge tree or subtree is possible. Multilinguality of knowledge node terms is supported too. Advanced manipulation with knowledge tree comprises copying or moving whole subtrees via clipboard or merging two subtrees to create one.

The Figure 6.1 shows a Temporomandibular Joint (TMJ) knowledge tree viewed in the MUDR KB Editor. More details about the TMJ knowledge tree will be in the results chapter.



MUDR - Editor znalostní báze

Copyright (c) 2003 EuroMISE Centrum - Kardio

```
TOOTH_AND_GING (Adresář) - zub a gingiva :
   TMJ (Adresář) - kloub :
      1,2 datum (Číslo) - datum počátku obtíží :
      1,2 obtize (Číslo) - délka trvání obtíží : měsíce/roky
      charakter (Adresář) - charakter potíží :
      osetreni (Boolean) - první ošetření : je toto první ošetření?
      drivlecba (Adresář) - dřívější léčba :
      anam (Adresář) - anamnéza :
         anamnest (Text) - anamnestické údaje :
         operace (Boolean) - operace :
            orofac (Boolean) - orofaciální oblast :
               dentoalv (Boolean) - dentoalveolární :
                  popis (Text) - popis operace :
               maxfac (Boolean) - maxilofaciální :
                  popis (Text) - popis operace :
            jine (Text) - jiné operace :
         urazy (Boolean) - úrazy :
            popis (Text) - :
         alergie (Boolean) - alergie :
            popis (Text) - :
```

Jazyk: Čeština

Výraz: délka trvání obtíží

Popis: měsíce/roky

Figure 6.1: Temporomandibular joint knowledge base in MUDR KB editor.

6.4.2 E-R Model Editor

Entity-relationship models (E-R models) are in software engineering an abstract and conceptual representation of data. E-R modelling is a database modelling method, used to produce a type of conceptual schema or semantic data model of a system, often a relational database, and its requirements in a top-down fashion.

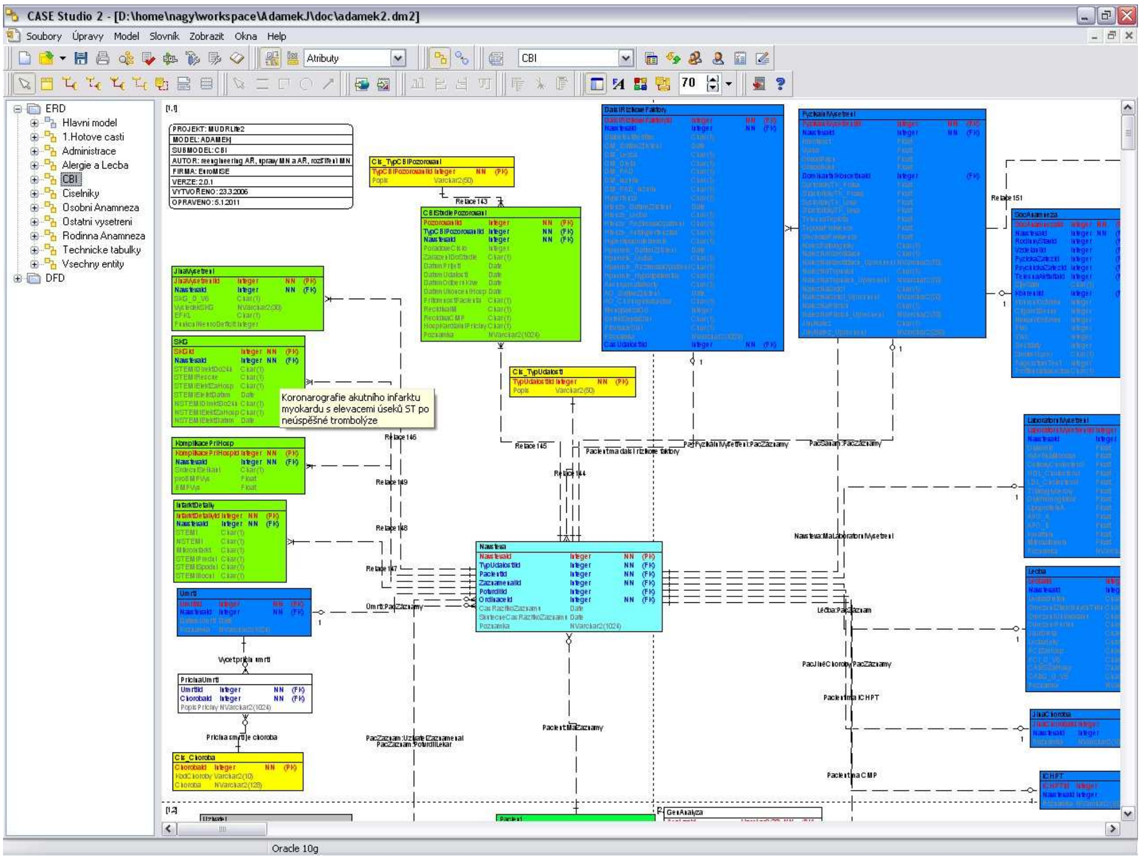
Fundamental parts of E-R models are entities, relationships and attributes, which can have graphical representation thus forming an E-R diagram. These diagrams have many variations in the graphical representation one of which is so-called Crow's Foot Notation. Crow's Foot diagrams represent entities as boxes, and relationships as lines between the boxes. The ends of these lines are shaped to represent the cardinality of the relationship.

Diagramming tools that support E-R diagrams are often called CASE (Computer-aided software engineering) tools. In this thesis a CaseStudio v.2.25 was used to model relational dabase schemas used in MUDRLite based EHR-Ss. This tool enables to create an E-R diagram, extend it with textual database objects like triggers or stored procedures and finally it can export a SQL script to initialize the database. Various database engines are supported (we used MS SQL and Oracle 10g). The Figure 6.2 shows a part of the E-R diagram of ADAMEKj EHR, which is described in the Section 7.3.2. The diagram itself is depicted in Figure D.1 in Appendix D.

6.4.3 openEHR Clinical Knowledge Manager

The notion of archetypes has been known for several years. As a result there already exist some archetypes describing most common concepts and their sharing started to have sense. For this purpose a Clinical Knowledge Manager (CKM) is available at http://www.openehr.org/knowledge/.

It is a repository designed to share archetypes (serves as a library of archetypes and templates), supports the full life cycle management of archetypes (from its proposal, through draft and to published state) and provides governance of the knowledge artifacts. The Figure 6.3 depicts the web user interface of the CKM showing result (`archetype openEHR-EHR-OBSERVATION.blood_pressure.v1`) of a search for ”blood pressure”.

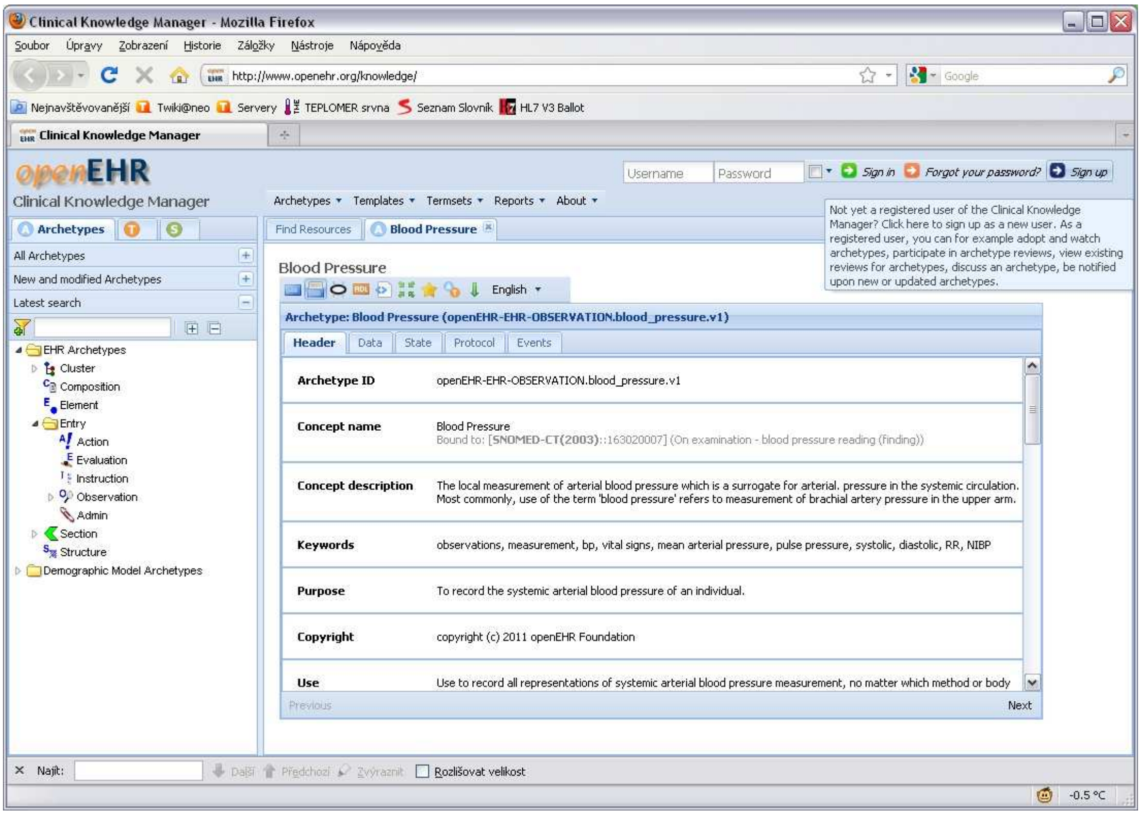


CASE Studio 2 - [D:\home\nagy\workspace\AdamekJ\doc\adamek2.dm2]

PROJEKT: MUDRLite2
MODEL: ADAMEK
SUBMODEL: CBI
AUTOR: reengineering AR, úpravy MN a AR, rozšíření MN
FIRMA: EuroMISE
VERZE: 2.0.1
VYTVOŘENO: 23.3.2006
OPRAVENO: 5.1.2011

Koronarografie akutního infarktu myokardu s elevacemi úseků ST po neúspěšné trombolýze

Figure 6.2: Case tool called CaseStudio by Quest Software Inc.



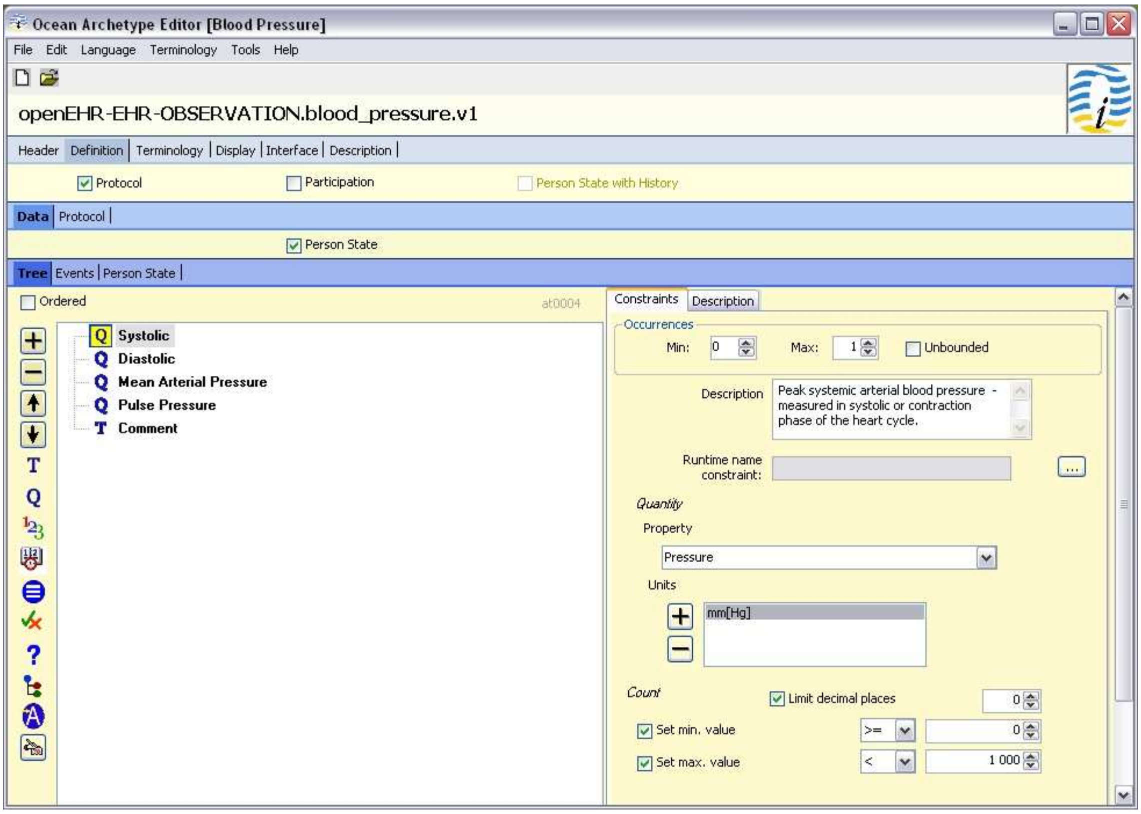
Not yet a registered user of the Clinical Knowledge Manager? Click here to sign up as a new user. As a registered user, you can for example adopt and watch archetypes, participate in archetype reviews, view existing reviews for archetypes, discuss an archetype, be notified upon new or updated archetypes.

Blood Pressure

Archetype: Blood Pressure (openEHR-EHR-OBSERVATION.blood_pressure.v1)

Archetype ID	openEHR-EHR-OBSERVATION.blood_pressure.v1
Concept name	Blood Pressure Bound to: [SNOMED-CT(2003)::163020007] (On examination - blood pressure reading (finding))
Concept description	The local measurement of arterial blood pressure which is a surrogate for arterial. pressure in the systemic circulation. Most commonly, use of the term 'blood pressure' refers to measurement of brachial artery pressure in the upper arm.
Keywords	observations, measurement, bp, vital signs, mean arterial pressure, pulse pressure, systolic, diastolic, RR, NIBP
Purpose	To record the systemic arterial blood pressure of an individual.
Copyright	copyright (c) 2011 openEHR Foundation
Use	Use to record all representations of systemic arterial blood pressure measurement, no matter which method or body

Figure 6.3: Blood pressure archetype in Clinical Knowledge Manager web page.


Ocean Archetype Editor [Blood Pressure]
File Edit Language Terminology Tools Help
Header Definition Terminology Display Interface Description
Data Protocol
Tree Events Person State
Constraints Description
at0004

openEHR-EHR-OBSERVATION.blood_pressure.v1

☑ Protocol □ Participation □ Person State with History

☑ Person State

□ Ordered

- Q Systolic
- Q Diastolic
- Q Mean Arterial Pressure
- Q Pulse Pressure
- T Comment

Occurrences
Min: 0 Max: 1 □ Unbounded

Description: Peak systemic arterial blood pressure - measured in systolic or contraction phase of the heart cycle.

Runtime name constraint:

Quantity
Property: Pressure
Units: mm[Hg]

Count
☑ Limit decimal places 0
☑ Set min. value >= 0
☑ Set max. value < 1 000

Figure 6.4: Ocean Informatics archetype editor editing the blood pressure archetype.

For the purpose of this thesis the CKM was utilised as an archetypes repository. During the mapping process of MDMC to openEHR archetypes (see Section 7.1.3) the CKM was used to search for existing archetypes, which represent concepts from the MDMC.

In case that the CKM repository does not contain certain archetypes, there is a tool for designing new archetypes described in the next section. Methods how to create new archetypes are discussed in Section 7.7.

6.4.4 Ocean Informatics Archetype Editor

The Ocean Archetype Editor is a tool developed by Ocean Informatics and supports creating archetypes and binding terms they contain to coding systems. After creating an archetype it can be exported in an abstract syntax, i.e. the ADL format [42].

Archetypes approach is independent to the underlying reference model. However, Ocean Archetype Editor only supports creation of archetypes based on the openEHR reference model (described on page 4.4.1). Apart from this limitation this tool is capable of managing all features of archetypes (header section with data concerning purpose, use and misuse of the archetype, definition section, terminology section, multilinguality of archetypes, etc.). The modelled archetypes can be exported in addition to ADL in RTF, HTML, OWL and XML formats.

6.4.5 Tool for Formalised Models' Synchronisation

Similar to text comparing tools there is a need to compare various formal models emerging mainly in the software development process. Under the term "formal" or "formalised models" we understand UML models (e.g. class diagrams), E-R models, data dictionary representation in a database, etc. In this thesis openEHR archetypes describing CC or MUDR knowledge base trees representing structure of collected variables can be considered as formal models as well.

For relational database schemas (RDS) synchronisation there exist several tools most of which are commercial however a trial version is available for free. These tools fulfil the basic requirements (database schema comparison and synchronisation) but differ in the range of supported database engines, possibilities of extensibility and other features. To name only some: MySQL table patcher [59], SQLDBCompare [60], Omega Sync Pro [61] and Adept SQL Diff [62]. For the purpose of this thesis a tool named Schemagic was chosen, which was developed as a part of author's master thesis [63] in 2005.

The Schemagic tool was developed to compare and synchronise RDS. For operating system independency reasons the application was based on Java and implemented following requirements: universality, extensibility, off-line usage (synchronising schemas is possible also without a direct connection to database), automation support as a command-line-oriented tool, national languages support and open-source. The tool is hosted at the SF web pages [64]. Its source codes are based on design patterns [65] in order to be easily modifiable and extendable by independent developers in the future. Thanks to its universality and extensibility, Schemagic tool could be extended to compare instances of MUDR Knowledge Trees and openEHR archetypes. The description of this extension is desrcibed in the Results chapter, Section 7.4.3 and 7.4.4.

Chapter 7

Results

This chapter presents the results of the clinical content harmonisation research conducted within the frame of this PhD thesis. First, modelling of clinical content using MDMC (see p. 58) is presented. In order to fix the semantics of its concepts, these were mapped to coding systems, which is described in Section 7.1.1. Than the MDMC was transformed (see Section 7.1.2) into HL7 v3 RIM derived classes, thus producing Local Information Models (LIM), enabling the implementation of HL7 v3 based communication. MDMC concepts were also paired with existing archetypes (see Section 7.1.3) to test the exchange of clinical information via openEHR solutions.

In the sections 7.2 and 7.3 developed EHR systems from the field of dentistry and cardiology are described. They fill the gap in the Czech healthcare environment lacking structured health records.

Next, formal models comparison and synchronisation was implemented (see Section 7.4). The ability to compare and possibly synchronise given formal models is inevitable during the clinical content harmonisation process. A universal tool for this purpose was developed and named Schemagic (see p. 68). User interface of this tool was implemented in the frame of this thesis (Section 7.4.1). Furthermore the system was extended to support MUDR KB trees and openEHR archetypes. The former one is described in Section 7.4.3 and the latter in Section 7.4.4.

Various approaches to semantic interoperability achievement are shown in order to validate the possibilities of utilisation of the CC modelled in MDMC and mapped by means of conding systems, HL7 RIM and openEHR archetypes. One solution is based on HL7 v3 messaging and is described in Section 7.5. The other utilises openEHR architecture and is summarised in Section 7.6. Finally, an attempt to define an archetype modelling methodology is discussed in Section 7.7.

7.1 Modelling the Clinical Content

Modelling of the clinical content was based mainly on the MDMC introduced on page 58. In order to test the clinical data exchange between two different EHR-Ss based on MDMC using HL7 v3 standard and openEHR archetypes, it was necessary to create an equivalent representation of the clinical content described by the MDMC using methodologies defined by each of these approaches.

7.1.1 Mapping MDMC Concepts to Established Coding Systems

The concepts of the MDMC were mapped to the coding systems mainly in order to ease the development of equivalent representations of the MDMC utilising the `code` attribute of HL7 v3 RIM classes or openEHR Archetypes' term bindings.

In Table 7.1 a list of MDMC concepts and equivalent terms from SNOMED CT and LOINC are summarised. Due to the size of the MDMC (approx. 200 concepts) only concepts related to ”Physical examination” are summarised as an illustration.

The first column represents the name of the concept in MDMC translated from Czech into English. The second column contains SNOMED CT descriptions as they were found in concepts core file. These descriptions are the terms or names assigned to a SNOMED CT concept. The next column is a concept ID assigned to given concept by SNOMED CT. The fourth column represents the LOINC Component, particularly its principal name part. The `LOINC_NUM` column contains a unique permanent code – the LOINC code, which should be used by systems to identify LOINC components. The scale of the measure is specified by the abbreviation of type of scale, which is stored in the `LOINC Scale Type` column. In the presented example mapping only following scale

Name of encoded	**SNOMED CT**	**SNOMED CT**	**LOINC**	**LOINC**	**LOINC**	**LOINC**
MDMC concept	**description**	CONCEPT ID	Component	LOINC-NUM	Scale Type	Class
physical examination	-	-	Physical findings	29544-4	Nom	H&P.PX
weight	Weight finding (finding)	107647005	Body weight	3141-9	Qn	BDYWGT.ATOM
	Height and weight (observable entity)	162879003				
height	Body height measure (observable entity)	50373000	Body height	3137-7	Qn	BDYHGT.ATOM
body temperature	Body temperature finding	105723007	Body temperature	8328-7	Qn	BDYTMP.MOLEC
	Body temperature (observable entity)	276535009				
waist size	Abdominal girth measurement (procedure)	48094003	Circumference	9844-2	Qn	BDYCRC.ATOM
hips size	-	-				
body mass index	BMI	162859006	Body mass index measurement	39156-5	Qn	BDYWGT.ATOM
waist hip ratio	WHR (waist/hip ratio)	248367009	-	-	-	-
BP - upper right extremity	-	-	-	-	-	-
systolic pressure	On examination - Systolic BP reading (finding)	163030003	Intravascular systolic	8479-8	Qn	BP.ATOM
diastolic pressure	On examination - Diastolic blood pressure reading (finding)	163031004	Intravascular diastolic	8462-4	Qn	BP.ATOM
BP - upper left extremity	-	-	-	-	-	-
systolic pressure	On examination - Systolic BP reading (finding)	163030003	Intravascular systolic	8479-8	Qn	BP.ATOM
diastolic pressure	On examination - Diastolic blood pressure reading (finding)	163031004	Intravascular diastolic	8462-4	Qn	BP.ATOM
pulse rate	Pulse rate finding (finding)	301147003	Heart beat	8893-0	Qn	HRTRATE.MOLEC
	Pulse rate (observable entity)	78564009				
respiratory rate	Respiratory rate (observable entity)	86290005	Respiration rate	18686-6	Set	ATTACH.ED
somatic finding is physiological	-	-	-	-	-	-
pathological finding on carotids	-	-	-	-	-	-
pathological finding on peripheral arteries	-	-	-	-	-	-
pathological finding on heart when physically examined	-	-	-	-	-	-
pathological finding on lungs when physically examined	-	-	-	-	-	-
other pathological finding	-	-	-	-	-	-

Table 7.1: Mapping Physical examination concepts of MDMC to SNOMED CT and LOINC.

types are used: Nom – nominal, Qn – Quantitaive, Ord – Ordinal. The column `LOINC Class` represents further refinement of the LOINC structure. A complete list of LOINC classes can be found in [66]. In Table 7.1 only following classes appeared:

- `ATTACH.ED` – Emergency department attachment.
- `BDYWGT.ATOM` – Body weight atomic.
- `BDYHGT.ATOM` – Body height atomic.
- `BDYTMP.MOLEC` – Body temperature atomic.
- `BDYCRC.ATOM` – Body circumference atomic.
- `BP.ATOM` – Blood pressure atomic.
- `H&P.PX` – Physical.
- `HRTRATE.MOLEC` – Heart rate molecular.

The mapping process was performed manually by author because of the complexity of finding matching concept code to given concept from the MDMC. These concepts originally named in Czech had to be translated into English before the mapping process started. After that SNOMED CT Core Concepts file and LOINC DB file were searched for equivalent concepts names. This search was performed by the author in cooperation with medical doctors and the process was finished with following results: Approx. 85% of MDMC concepts appear to have an equivalent concept in at least one coding system [67]. During the mapping some problematic concepts were found, which can be divided into four groups. Partially problematic concepts have several mapping possibilities to various synonyms, which differ slightly in their meanings and usually also in their classification codes. Concepts may also be too general or too narrow, so that classification systems contain only concepts of a narrower or wider meaning. There were also concepts that could not be mapped to any of the classification systems.

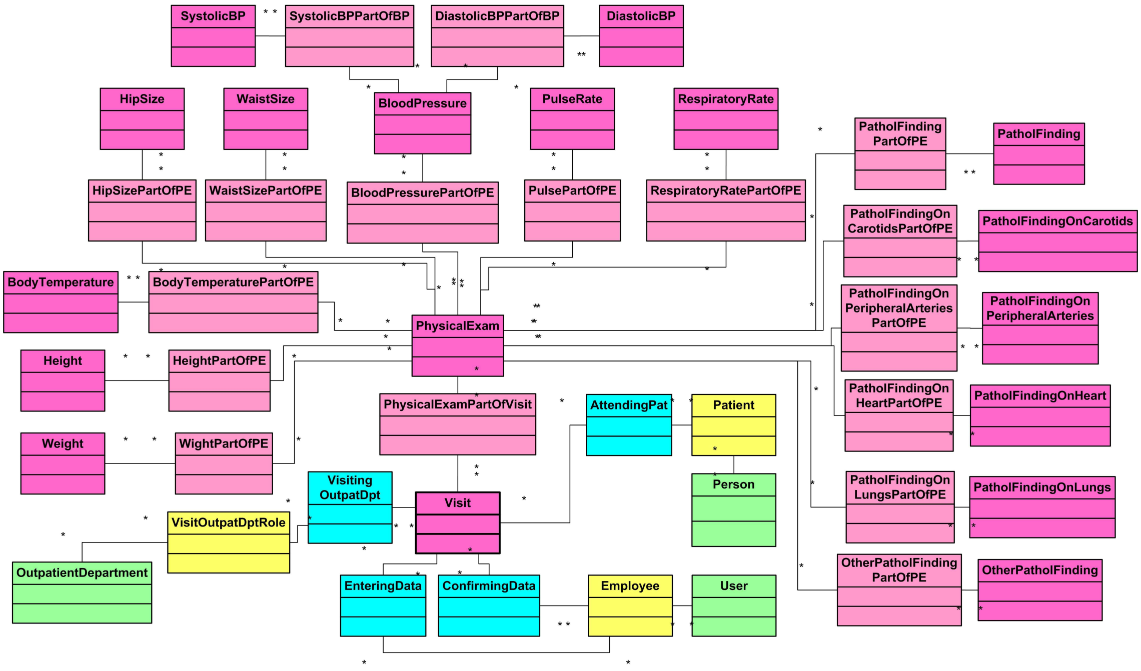


Figure 7.1: LIM for Physical Examination of the Patient.

7.1.2 Modelling MDMC Concepts by Means of HL7 v3 RIM Classes

After the process of mapping clinical concepts of the MDMC on established coding systems, it was necessary to create refined models based on HL7 v3 RIM in order to examine the possibilities of the HL7 v3 standard in structured medical data exchange. These refined models were named Local Information Models (LIM) and its main purpose was to work as a first step for developers of the particular EHR into the HL7 v3 world. A tool for this purpose was developed, called Modeller [68]. It contained documentation on RIM classes and their attributes to simplify the refinement process for EHR developers. All this work was performed in the frame of the ITDCSH project. The author's task was to create LIM classes representing clinical content defined by ADAMEKj application, which will be introduced in Section 7.3.2.

As an example of a LIM model a class diagram of Physical examination concepts stored in ADAMEKj EHR is shown in Figure 7.1. A complete set of LIM models in form of UML Class Diagrams is presented in Appendix A on page 129.

The Figure 7.1 shows a predominance of pink-colored classes in the model, which

MDMC concept name	**Archetype ID**
Height	openEHR-EHR-OBSERVATION.height.v1
Weight	openEHR-EHR-OBSERVATION.body_weight.v1
Body temperature	openEHR-EHR-OBSERVATION. body_temperature.v1
Blood pressure	openEHR-EHR-OBSERVATION.blood_pressure.v1
Heart rate	openEHR-EHR-OBSERVATION. heart_rate-pulse.v1
Breath frequency	openEHR-EHR-OBSERVATION. respiration.v1
Waist circumference	openEHR-EHR-OBSERVATION.waist_hip.v1
Hip circumference	openEHR-EHR-OBSERVATION.waist_hip.v1

Table 7.2: Selected archetypes matching the concepts of the MDMC.

refers to the `Act` class of the RIM. Most of clinical concepts are modelled either as subclasses of the `Act` or `Observation` class (descendant of `Act`). Classes located in gray rectangle are repeated in all class diagrams in Appendix A. They represent the object, the place and the subject of all Acts modeled in LIMs, i.e. patient, outpatient office and GP.

After the preparation of LIM models the next step was to integrate them into the solution described in Section 7.5.

7.1.3 Matching MDMC Concepts to Existing Archetypes

After mapping MDMC concepts to established coding systems like SNOMED CT, or LOINC, they were modelled using HL7 v3 RIM classes to create LIM models. Each RIM derived class contains a `code` attribute, which could be filled with a value [coding system name, code value] to create a semantic link to a coding system. At this point the mapping done in Section 7.1.1 was sucessfully utilised. To show that the HL7-solution is not the only one, the openEHR archetype repository (CKM) was searched for archetypes covering the same field as MDMC concepts. Archetypes found to match only MDMC Physical examination concepts are shown in Table 7.2 for the lack of space reasons.

Other archetypes representing equivalent concepts to those from MDMC are summarised in Appendix C. The search process was performed manually, since there is no stable mapping (stored in a repository) of archetypes on any coding systems. Searching archetypes using e.g. SNOMED CT concept codes would ease this process and already encoded MDMC concepts could be utilised too. Instead, English equivalents (or their fractions) of clinical concept names were entered into the resource search section in the CKM.

7.2 Electronic Health Record in Dentistry

7.2.1 MUDRLite EHR with the DentCross Component

The first practical implementation of the MUDRLite system (see p. 54) was realised in the area of dentistry. A specific requirement for the implemented EHR instance was the advanced form of a user interface for data entry and presentation. The main part of the user interface is represented by a so-called dental cross – a graphical schema of dental arcs, divided by quadrants and showing the results of all examinations and treatments for each tooth. This kind of functionality could not be realized by the standard set of visual controls and components of MUDRLite, therefore a special component named DentCross (Dental Cross) was developed. The DentCross component is implemented as a stand-alone library DentCross.dll that was completely developed for the NET Framework platform using the Microsoft Visual Studio.NET 2003 development tool. For the end-user the DentCross component looks similar to a dental orthopantomogram. This component is fully interactive and enables recording fully structured dental information that can be inserted in a user-friendly way either by mouse or keyboard (Figure 7.2).

A dentist can choose among about 60 different actions, treatment procedures or tooth parameters that are displayed graphically and lucidly. The features of the component include the support of various forms and shapes of teeth, the exact position of a tooth, impacted teeth, agenesis of a tooth, primary and secondary caries, filling of a tooth, pulp and periodontal pathology, root canal treatment, inlay, onlay and overlay, post and core, crowns, partial veneer crowns, bridge(s), dentures – complete denture, over denture, as well as removable partial dentures, implants, dent alveolar surgery, calculus, PBI, movement of a tooth, periodontal pocket, bone resorption and temporomandibular joint. The DentCross component includes a treatment plan combined with a calendar

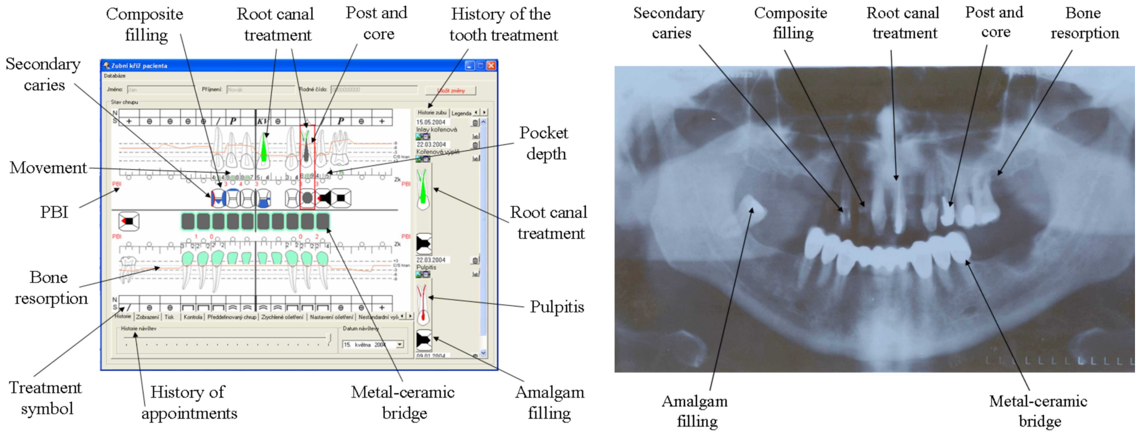


Figure 7.2: DentCross Component and its comparison to orthopantomogram.

that enables scheduling the patients'visits and treatments.

Data structuring in the Czech healthcare is on a relatively low level and in dentistry the data capture is done via filling in dentition characteristics into pre-printed paper forms, which include more or less standardized symbols (e.g. "/ " for caries, "-" for pulpitis, or "x" for a tooth to be extracted). Symbols are placed in the section corresponding to a particular tooth. These symbols were preserved also in DentCross component for dentists' convenience and for reasons of user acceptance of the new tool.

This advanced form of health documentation will lead to easier and more complex treatment evaluation based on the bigger amount of relevant information, which is stored transparently using the Interactive Dental Cross Component.

The database layer of this tool is implemented using MS SQL server. The stored clinical content is based on dental knowledge base described in Section 6.3 on page 60. The Dental KB tree was transformed by the author into relational database schema consisting of almost 50 tables. The E-R Diagram of this schema is in Appendix E.

This tool helps to create the transparent part of the EHR on the whole dentition and individually accomplished examinations. The information recorded in a graphical form accelerates dentist's decision-making and it enables a more complex view in suggesting an evaluation. Another very important use of structured dental information is in the field of forensic dentistry. The usage of the information collected by the interactive DentCross component to support the identification in forensic dentistry is demonstrated in [69].

7.2.2 Voice-controlled Data Entry in the Dental Electronic Health Record

Interconnection of ASR (Automated Speech Recognition) module developed at the Department of Cybernetics, University of West Bohemia in Pilsen [70] and the DentCross component of the MUDRLite EHR resulted in an application called DentVoice. The junction of voice control and graphical representation of the dental arch makes hand-busy activities in dental practice easier and more comfortable thanks to minimisation of the contact with a computer. Dentists were involved in the whole development process; therefore manipulation with the DentCross component was designed to be as easy as possible.

The prototype application consists of the DentCross component with the integrated TCP/IP client of the ASR server and the voice commands (discrete speech) definition file. The ASR client uses a DentCrossHandler class that implements all functionality of the DentCross component. The ASR server remains independent on the target domain – dentistry, and makes its usage in other domains possible (e.g. cardiology).

The speech recognition is activated immediately after the DentCross component start-up. The recognition process can be paused or stopped by a special voice command or using the user interface. Voice commands can be divided in two groups: global manipulating commands and context dependent commands. Global commands are always available and designed to manipulate the recognition process (pause, resume and stop) and to close message boxes opened by the application to warn the user. Context dependent commands rely on the current state of the DentCross component and can be further divided into 33 command groups because it can fall into one of 33 states (e.g. tooth treatment, caries placement, caries type, root canal treatment material).

The DentCross component was successfully tested by students of the 1st Faculty of Medicine of Charles University in Prague. During these tests students checked their own dental status as well as some volunteer patients. Experts from the Department of Biological Analysis, Institute of Criminalistics in Prague used the DentCross component for dead corpses' identification. The DentVoice application was deployed in November 2007 for testing at the Department of Paediatric Stomatology at the Motol University

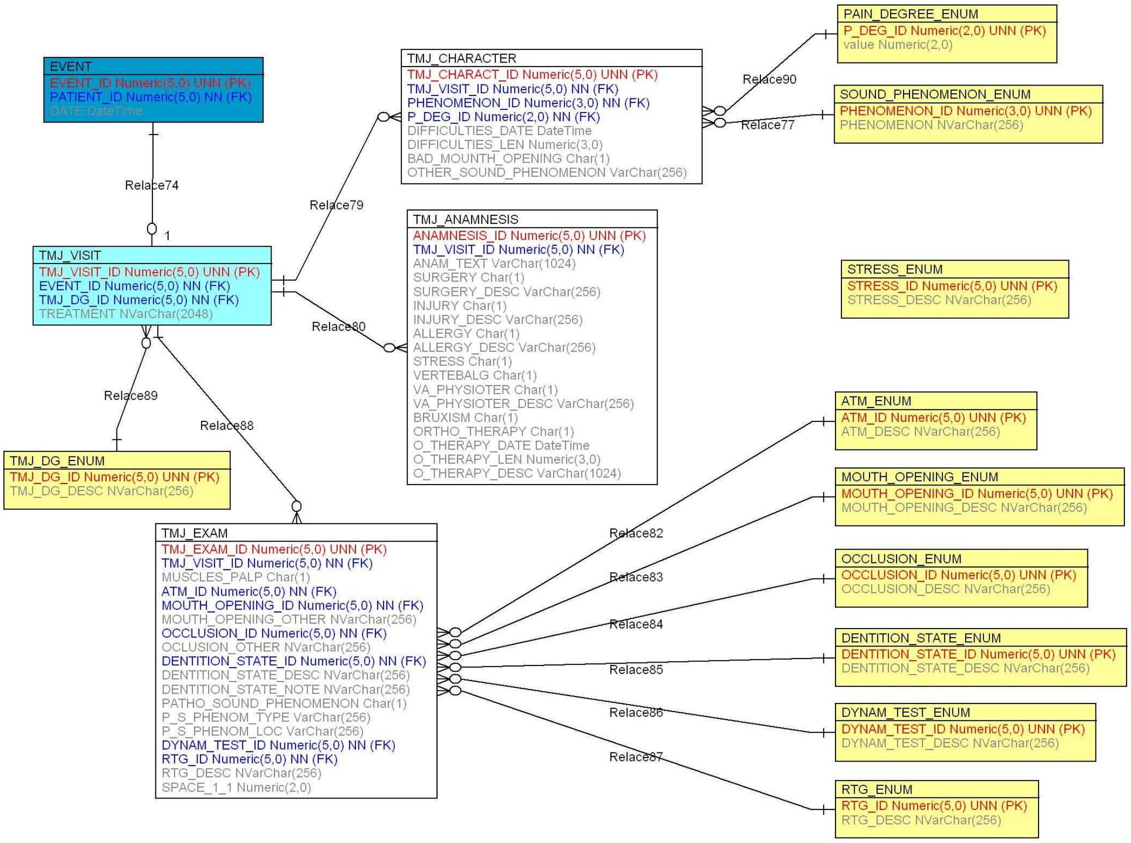

EVENT
EVENT_ID Numeric(5,0) UNN (PK)
PATIENT_ID Numeric(5,0) NN (FK)
DATE DateTime
Relace74
1
TMJ_VISIT
TMJ_VISIT_ID Numeric(5,0) UNN (PK)
EVENT_ID Numeric(5,0) NN (FK)
TMJ_DG_ID Numeric(5,0) NN (FK)
TREATMENT NVarChar(2048)
Relace89
Relace88
TMJ_DG_ENUM
TMJ_DG_ID Numeric(5,0) UNN (PK)
TMJ_DG_DESC NVarChar(256)
Relace79
Relace80
TMJ_CHARACTER
TMJ_CHARACT_ID Numeric(5,0) UNN (PK)
TMJ_VISIT_ID Numeric(5,0) NN (FK)
PHENOMENON_ID Numeric(3,0) NN (FK)
P_DEG_ID Numeric(2,0) NN (FK)
DIFFICULTIES_DATE DateTime
DIFFICULTIES_LEN Numeric(3,0)
BAD_MOUNTH_OPENING Char(1)
OTHER_SOUND_PHENOMENON VarChar(256)
Relace90
Relace77
PAIN_DEGREE_ENUM
P_DEG_ID Numeric(2,0) UNN (PK)
value Numeric(2,0)
SOUND_PHENOMENON_ENUM
PHENOMENON_ID Numeric(3,0) UNN (PK)
PHENOMENON NVarChar(256)
TMJ_ANAMNESIS
ANAMNESIS_ID Numeric(5,0) UNN (PK)
TMJ_VISIT_ID Numeric(5,0) NN (FK)
ANAM_TEXT VarChar(1024)
SURGERY Char(1)
SURGERY_DESC VarChar(256)
INJURY Char(1)
INJURY_DESC VarChar(256)
ALLERGY Char(1)
ALLERGY_DESC VarChar(256)
STRESS Char(1)
VERTEBALG Char(1)
VA_PHYSIOTER Char(1)
VA_PHYSIOTER_DESC VarChar(256)
BRUXISM Char(1)
ORTHO_THERAPY Char(1)
O_THERAPY_DATE DateTime
O_THERAPY_LEN Numeric(3,0)
O_THERAPY_DESC VarChar(1024)
STRESS_ENUM
STRESS_ID Numeric(5,0) UNN (PK)
STRESS_DESC NVarChar(256)
TMJ_EXAM
TMJ_EXAM_ID Numeric(5,0) UNN (PK)
TMJ_VISIT_ID Numeric(5,0) NN (FK)
MUSCLES_PALP Char(1)
ATM_ID Numeric(5,0) NN (FK)
MOUTH_OPENING_ID Numeric(5,0) NN (FK)
MOUTH_OPENING_OTHER NVarChar(256)
OCCLUSION_ID Numeric(5,0) NN (FK)
OCLUSION_OTHER NVarChar(256)
DENTITION_STATE_ID Numeric(5,0) NN (FK)
DENTITION_STATE_DESC NVarChar(256)
DENTITION_STATE_NOTE NVarChar(256)
PATHO_SOUND_PHENOMENON Char(1)
P_S_PHENOM_TYPE VarChar(256)
P_S_PHENOM_LOC VarChar(256)
DYNAM_TEST_ID Numeric(5,0) NN (FK)
RTG_ID Numeric(5,0) NN (FK)
RTG_DESC NVarChar(256)
SPACE_1_1 Numeric(2,0)
Relace82
Relace83
Relace84
Relace85
Relace86
Relace87
ATM_ENUM
ATM_ID Numeric(5,0) UNN (PK)
ATM_DESC NVarChar(256)
MOUTH_OPENING_ENUM
MOUTH_OPENING_ID Numeric(5,0) UNN (PK)
MOUTH_OPENING_DESC NVarChar(256)
OCCLUSION_ENUM
OCCLUSION_ID Numeric(5,0) UNN (PK)
OCCLUSION_DESC NVarChar(256)
DENTITION_STATE_ENUM
DENTITION_STATE_ID Numeric(5,0) UNN (PK)
DENTITION_STATE_DESC NVarChar(256)
DYNAM_TEST_ENUM
DYNAM_TEST_ID Numeric(5,0) UNN (PK)
DYNAM_TEST_DESC NVarChar(256)
RTG_ENUM
RTG_ID Numeric(5,0) UNN (PK)
RTG_DESC NVarChar(256)

Figure 7.3: Data model for TMD EHR.

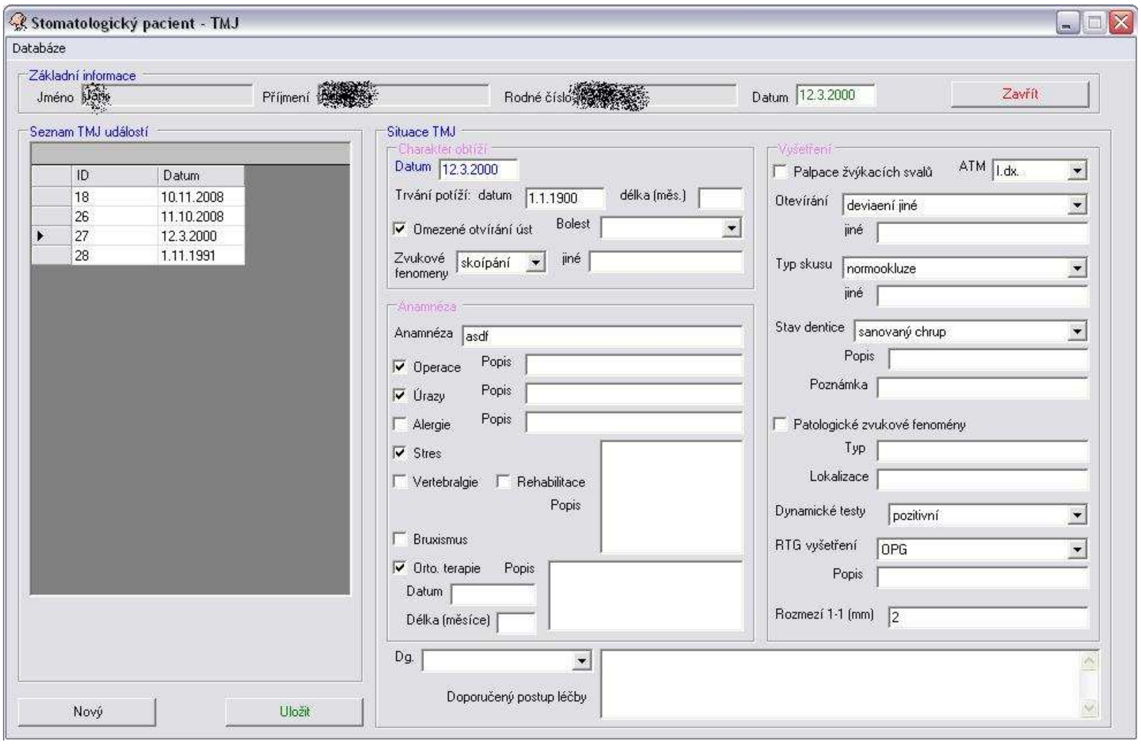
Hospital.

7.2.3 Temporomandibular Joint Disorders Recorded in MUDRLite EHR

A knowledge modelling tool from the MUDR EHR application suite was used to create a specialized TMD (TemporoMandibular joint Disorder) Knowledge Base. This knowledge base contains more than 80 concepts and extends the former Dental Knowledge Base, described in Section 6.3 on page 60.

In order to create MUDRLite forms, concepts from the TMD Knowledge Base had to be transformed in a relational database model (see Figure 7.3). The user interface and application for TMD was encoded into MLL in MUDRLite EHR, thus creating user forms interconnected with database layer.

The basic part of the interactive EHR in dentistry is composed by DentCross com-



Základní informace

Jméno [redacted] Příjmení [redacted] Rodné číslo [redacted] Datum 12.3.2000

Seznam TMJ událostí

	ID	Datum
	18	10.11.2008
	26	11.10.2008
▸	27	12.3.2000
	28	1.11.1991

Situace TMJ

Charakter obtíží

Datum 12.3.2000

Trvání potíží: datum 1.1.1900 délka (měs.)

☑ Omezené otvírání úst Bolest

Zvukové fenomeny skoípání jiné

Anamnéza

Anamnéza asdf

☑ Operace Popis

☑ Úrazy Popis

☐ Alergie Popis

☑ Stres

☐ Vertebralgie ☐ Rehabilitace Popis

☐ Bruxismus

☑ Orto. terapie Popis

Datum

Délka (měsíce)

Vyšetření

☐ Palpace žvýkacích svalů ATM l.dx.

Otevírání deviaení jiné

jiné

Typ skusu normookluze

jiné

Stav dentice sanovaný chrup

Popis

Poznámka

☐ Patologické zvukové fenomény

Typ

Lokalizace

Dynamické testy pozitivní

RTG vyšetření OPG

Popis

Rozmezí 1-1 (mm) 2

Dg.

Doporučený postup léčby

Figure 7.4: MUDRLite form to form TMD EHR.

ponent with the ASR and TTS system, which is built on the MUDRLite EHR. The specialized TMD EHR contains concepts defined in TMD Knowledge Base and is integrated in the user interface together with the DentCross component. It is initiated from the main window of the Integrated Dental-TMJ EHR (DentCross Component, ASR, TTS and TMD part) and has its own user interface (see Figure 7.4).

This application consists of 3 main parts. In the first part basic patient facts (name, id number, date of birth etc.) can be found, which helps in orientation. The second part consists of a list of events. They are sorted by the dates and by clicking on these dates records connected with one particular patient visit can by accessed. Third part consists of subjective patient problems, anamnestic datas, results of investigation, diagnosis statement and therapy recommendations. This part is the most important and represents all TMD data about the particular patient.

The circumstances that lead patient to visit the specialist and patient's personal feelings are very significant piece of information. A clinician can recognize many important facts from the patient's description. The type and character of difficulties, their duration and closer description can be recorded. In the anamnestic section basic personal anamnestic data and different conditions relevant to the TMD (parafunctions, stress, orthodontic treatment) are collected. Probably the most important is the section with investigation modalities. Investigation is in the TMD form classified into aspection (deformities, status of dentition, mouth opening etc.), palpation (masticatory muscles, TMJ) and auscultation (TMJ sounds). The last part in the TMD investigation playing the crucial role is the X-ray imaging (ortopantomographic image, computer tomography) and magnetic resonance. In the diagnosis classification the abovementioned RDC/TMD form was used. Also the therapy recommendation can be recorded and additionally can be changed at the end of each visit.

This application serves as a complex information source for the TMD treatment and is part of the whole system supporting decision making in the therapy of TMD, which leads to the final treatment plan:

1. performing dental examination and collecting anamnestic data and storing all into the interactive DentCross component,
2. training of opening of the mouth in the axis, without deviation,
3. using stabilisation splint,
4. performing computer-aided face bow analysis,
5. realising prosthetic treatment,
6. finalizing the treatment and applying soft splint,
7. recalling and evaluating the treatment by the patient.

7.3 Electronic Health Record in Cardiology

7.3.1 MUDRLite 2

The MUDRLite EHR suffered from limitations of user interface capabilities and the gradually rising complexity of the MLL Language. Thus, modern programming technologies were searched to replace MUDRLite's weaknesses and preserve its benefits.

MUDRLite2 was proposed as succesor technology to MUDRLite. Fundamental parts of MUDRLite2 are Hibernate [71], Spring Framework [72] and Spring Rich Client Project (Spring RCP) [73].

The first part is a high performance object/relational persistence and query service. Its task, among others, is to map application's domain objects to tables in the object/relational database. Objects' persistency is configured by XML mapping files (HBM – Hibernate Mapping). They map attributes of POJOs (Plain Old Java Objects) to columns of tables and define constraints on relations among tables or objects.

The second part, the Spring Framework, is a layered Java/J2EE [74] application framework. It provides automated configuration and wiring of application objects. Spring is well integrated with Hibernate, which simplifies management of domain objects' persistency.

The latter, the Spring RCP, is used to build a user interface. The Spring RCP is based on Java Swing; its main advantage is providing an elegant way to build highly-configurable, GUI standards following rich-client applications.

Applications using MUDRLite2 are developed in three steps. At the beginning, the developer creates a model of domain concepts in cooperation with a domain expert (GP, specialist etc.). Then the domain model is rewritten into the HBM file which is used to generate the SQL script with the definition of data layer and source files of Java classes with POJOs. Finally, the user interface is developed using the Spring RCP, whose constructs are wired with POJOs by the Spring Framework.

Enterprise programming technologies offer well-balanced ratio of Java programming (Spring RCP) vs. creating XML configuration files (Hibernate, Spring Framework). This resultes in still simple configurability by XML files while complex parts written in Java are developed using standard application development tools, like Eclipse or netBeans IDE.

7.3.2 ADAMEKj

ADAMEKj was created as a tool for data collection with an emphasis on collecting genetic data and clinical data in cardiology simultaneously. This extensibility was achieved by using MUDRLite2.

The ADAMEKj application is a pilot implementation of MUDRLite2 and it evaluates the benefits of used technologies. It is a two-tiered application consisting of the data layer and the user interface. The domain model of the application is based on MDMC. The user interface was inspired by the former ADAMEK application [57]. Since ADAMEKj is going to replace ADAMEK, a big focus was put on the acceptance of it by its users, i.e. physicians that were closely involved in the whole development process for smoother migration.

Data model of the ADAMEKj application is based on concepts from MDMC. It consists of 67 tables and is depicted in form of E-R Diagram in Appendix D on page 137.

An integral part of each EHR is its communication with other systems in health care environment. Some systems are just limited to import and export data in proprietary format, but ADAMEKj is capable of communicating utilizing HL7 v3 [9] messaging standard. Information models describing EHR system were created and necessary mapping of collected variables using international classifications and nomenclatures was prepared. To initiate communication one HL7 storyboard was implemented and the communication in the form of HL7 messages between two heterogeneous EHRs based on MDMC was thus achieved. The process of data interchange implemented using HL7 v3 standard is described in more detail in Section 7.5.

7.4 Formal Models Comparison and Synchronisation

Under the term formal or formalised models we understand UML models (e.g. class diagrams), E-R models, data dictionary representation in a database, etc. Here, openEHR archetypes describing clinical content and MUDR knowledge base trees representing structure of collected variables will be considered as such formal models.

The Schemagic tool was already introduced in Section 6.4.5 on page 68 and here its structure and extensions will be described in more detail (see also [63]). It was designed to compare and synchronise formal models. To achieve its universality these models had to be described using structures defined by Schemagic. The names of these structures are inspired by the original aiming of the tool to relational databases (db objects, db machine, db connection and schema handle).

Each formal model is considered to have a source/repository where it is stored. Relational database schemas are stored in a data dictionary of a RDBMS (Relational Database Management System), MUDR Knowledge trees are stored in MUDR EHR and openEHR archetypes are stored in ADL files. For such repository the Schemagic tool defines a structure called database machine (or db machine). Its definition contains the following information:

- *identifier of the machine* – unique identifier of the db machine among all machines defined in the tool. It is used for referencing from other configuration files.

  Example: `<dbmachine-def id="oracle">`

- *connections* – list of supported database connections by that machine. Their definitions are in separate files. As types of connections can be shared among db machines, each connection is referenced through its unique identifier.

  Example: `<connection-ref ref="jdbc"/>`

- *model definition* – the hierarchy forest definition is arranging db objects into specific hierarchy trees. Db objects are defined in separate file. The model definition is surrounded by a `<model-def>` element.

  Example:

```
<model-def>
 <hierarchy-forest>
  <hierarchy-tree id="schema_tree">
   <object-ref type="ora_schema">
    <object-ref type="ora_table">
     <object-ref type="ora_table_column"/>
    </object-ref>
   </object-ref>
  </hierarchy-tree>
 </hierarchy-forest>
</model-def>
```

- *list of loaders* – each object type bound in the hierarchy forest must have its corresponding loader in this list. Each loader definition is represented by a `<object-load>` element.

  Example:

  ```
  <object-load object_type="ora_schema" class_name="org.schemagic.
     plugin.database.load.impl.JdbcSqlDbObjectLoader">
   <parameters>
    <parameter name="SQL_COMMAND" value="..."/>
   </parameters>
  </object-load>
  ```

The next formal definitions are modelling actual formal model components as database objects (for short db objects) and connections to the repository where the formal model is stored. Database objects are represented as lists of ”property-value” pairs. The following example is a definition (therefore there are only properties’ names, not values) of a ”table column” object:

```
<object-def type="table_column">
 <properties-def>
  <property-def name="SCHEMA_NAME" />
  <property-def name="TABLE_NAME" />
  <property-def name="COLUMN_NAME" key="true" />
  <property-def name="TYPE_NAME" />
  <property-def name="COLUMN_SIZE" />
  <property-def name="IS_NULLABLE" />
  <property-def name="DEFAULT_VALUE" />
 </properties-def>
</object-def>
```

The definition of a ”connection” object is designed in a similar manner. It comprises unique identifier, list of connection properties names and finally a class name for a so-called factory class, which creates instances of given connection type. The following example shows a ”jdbc connection” definition:

```
<?xml version="1.0"?>
<connection-def id="jdbc" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
   instance" xsi:noNamespaceSchemaLocation="../xsd/connection_def.xsd">
 <properties-def>
  <property-def name="URL"/>
  <property-def name="USER"/>
  <property-def name="PASSWORD"/>
  <property-def name="DRIVER"/>
 </properties-def>
 <factory class_name="org.schemagic.plugin.database.connection.jdbc.
    JdbcConnectionFactory"/>
</connection-def>
```

Hitherto only definition structures were summarised. In order to instantiate database objects, forming a hierarchy defined by a db machine using a connection, there must be a way to define which formal model from the repository should be loaded. Schema handle is such an entry point to an instance of a formal model.

```
<?xml version="1.0"?>
<schema-handle id="scott_schema" xmlns:xsi="http://www.w3.org/2001/
    XMLSchema-instance"
xsi:schemaLocation="../def/xsd/schema_handle.xsd">
 <use-dbmachine id="oracle"/>
 <use-connection id="jdbc"/>
 <model-load class_name="org.schemagic.plugin.database.load.impl.
     DbModelLoader"/>
 <parameters>
  <connection id="jdbc">
   <properties>
    <property name="DRIVER" value="oracle.jdbc.driver.OracleDriver"/>
    <property name="URL" value="jdbc:oracle:thin:@neo.euromise.
        cz:1521:neodb"/>
    <property name="USER" value="HR"/>
    <property name="PASSWORD" value="treatment"/>
   </properties>
  </connection>
  <object id="global">
   <properties>
    <property name="LOADED_SCHEMA" value="HR"/>
   </properties>
  </object>
  <object id="ora_schema">
   <properties>
    <property name="SCHEMA_NAME" value="HR"/>
   </properties>
  </object>
 </parameters>
</schema-handle>
```

Once the Schemagic tool contains all definitions mentioned above and implementations of model-specific classes, it is ready to load, compare and, if possible, synchronise instances of given formal model.

The synchronisation process in a simplified form is described in Figure 7.5. During the synchronisation source schemas' meta-data are obtained either from a database or from an XML file describing the formal model. The next step of the synchronisation process is the execution of a specially designed diff algorithm that compares both the target and the source schemas and finds differences. The diff algorithm produces a diff model that holds all divergences. A rigorous description of the diff algorithm is described in [63] where all formal structures utilised in the diff algorithm are defined.

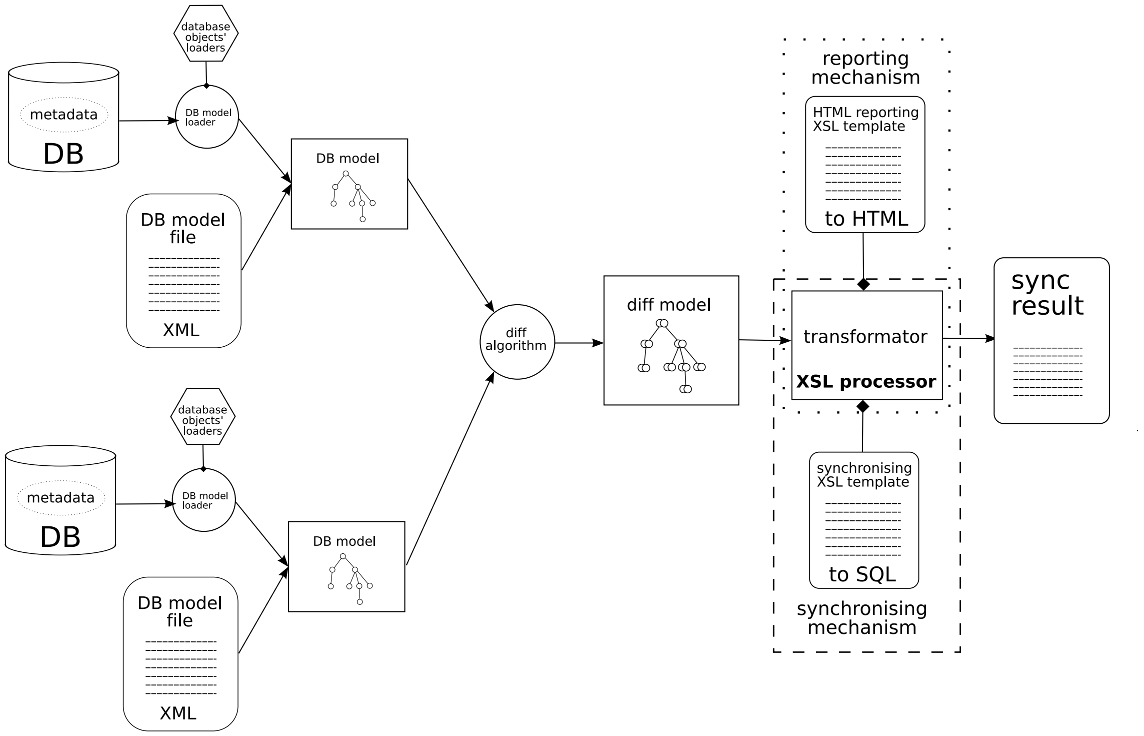


Figure 7.5: Schemagic's functionality model.

The final step of the process consists of creation of the synchronisation SQL script via XSL transformation [75] mechanism. The script can be run anytime to perform the synchronisation itself. The Schemagic tool together with its documentation can be found at WWW pages [64].

7.4.1 SchemagicCenter Application

Schemagic was a command line oriented application and lacked the user interface. This was improved by the development of the application called SchemagicCenter, which makes usage of the whole system more convinient. It covers following features of the Schemagic tool:

- creating, viewing and editing definition files (dbobjects, machines, connections, schema handles and plugins),
- executing all steps of the schema loading, comparison and synchronisation process in a wizard style or each step separately,

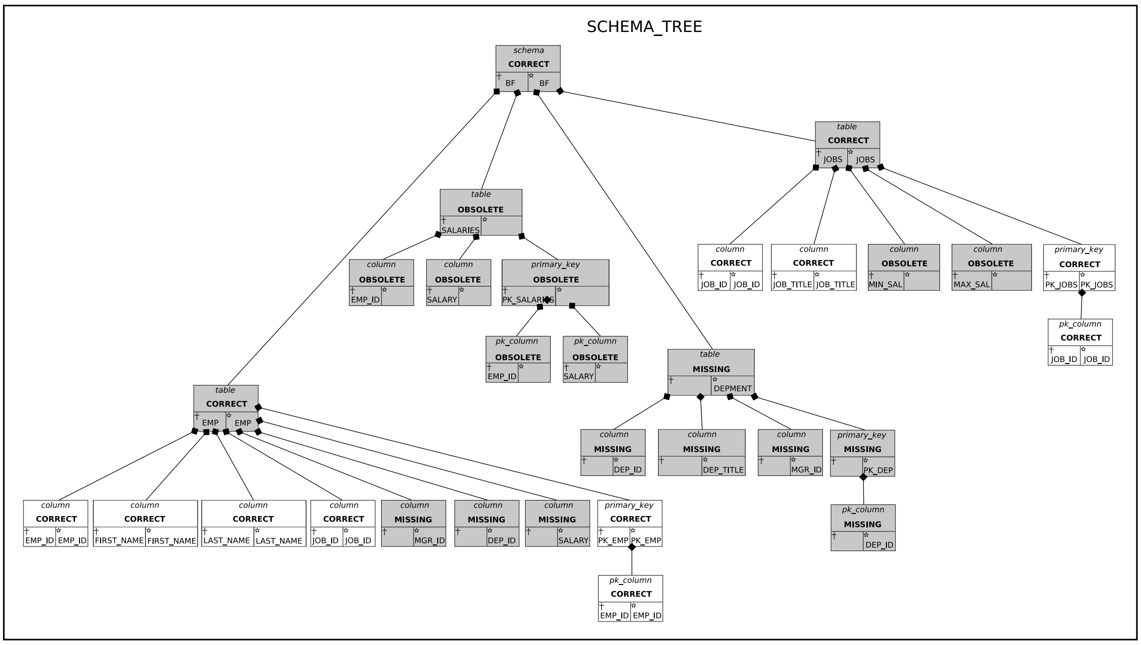
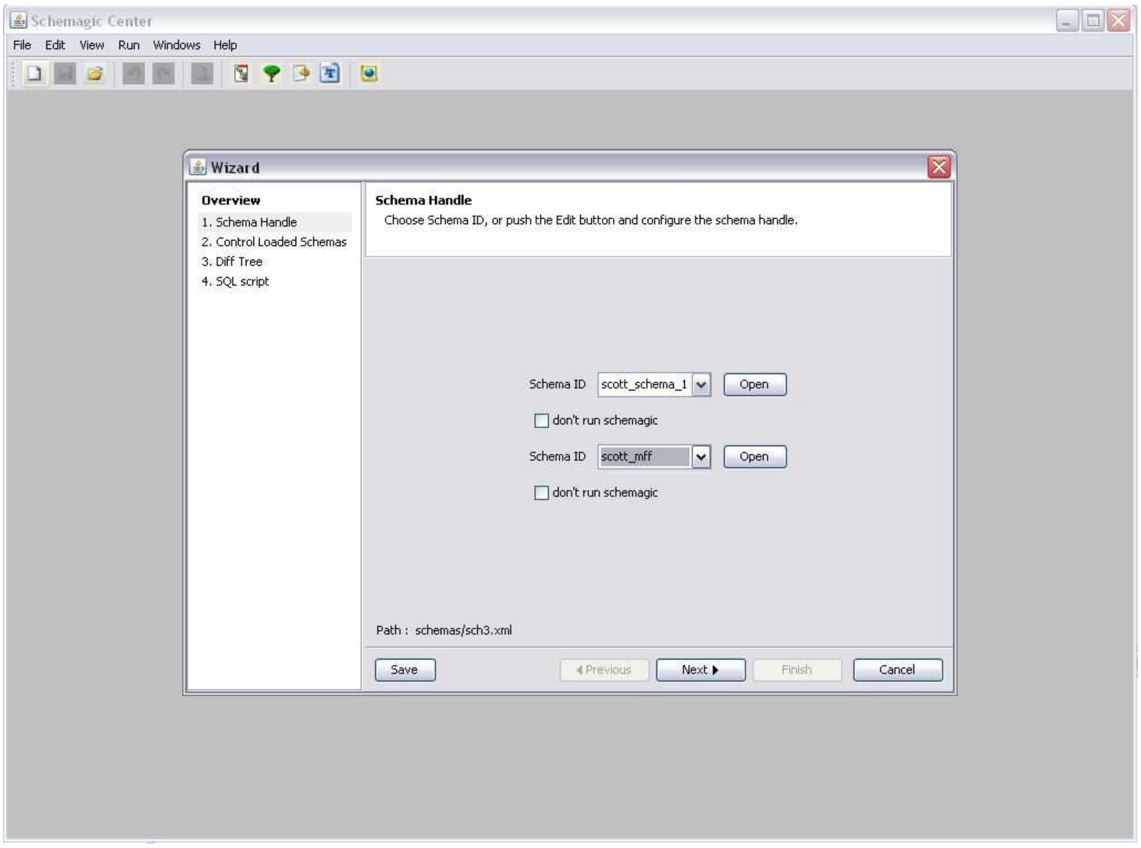


Figure 7.6: Diff model – the result of the diff algorithm.

- browsing the tree structure of the loaded formal model instance,
- viewing so-called diff tree, which is a result of the comparison step,
- syntax-highlighted viewer of the synchronisation result of relational schemas – the SQL script.

The SchemagicCenter tool is based on Java environment and is independent on operating system in the same way as the Schemagic tool. The first step of the wizard is depicted in Figure 7.7.

Following extensions of the Schemagic tool were succesfully accomplished using the editing features of the SchemagicCenter. However, configuration files contain among others class names for classes like connection factories, database object loaders, database object comparators etc. In order to extend the Schemagic tool to support MUDR knowledge trees and openEHR archetypes, these classes had to be programmed using a Java IDE tool (e.g. Eclipse).



Overview

1. Schema Handle
2. Control Loaded Schemas
3. Diff Tree
4. SQL script

Schema Handle

Choose Schema ID, or push the Edit button and configure the schema handle.

Schema ID: scott_schema_1

□ don't run schemagic

Schema ID: scott_mff

□ don't run schemagic

Path : schemas/sch3.xml

Figure 7.7: Schemagic's user interfare named SchemagicCenter.

7.4.2 Relational Database Schemas Synchronisation

In this section the synchronisation process will be explained. After accomplishing configuration of Schemagic to support new formal model, the comparison process is independent on formal models being compared. An example of relational database schemas comparison and synchronisation will be described in the following text.

Since both MUDRLite and ADAMEKj are based on relational databases and their data are stored directly in database tables, the first and most straightforward idea of synchronisation of EHR content was comparing relational schemas and creating synchronising script for solving differences. These differences occur during the development process as well as after deployment of the application when new versions of database schemas (better satisfying new requirements) are created.

As an example of comparison and synchronisation of relational schemas the following situation will be considered: During the development of a new version of an EHR system a need for albumin storage in the laboratory results section emerged, which was not supported on the database layer in the old version, which is already beeing used by GP. This situation requires modification of the existing database (old version) in order to work correctly with the new application that expects the new variable "albumin" in the database.

The first step the Schemagic tool is used to is loading metadata of both schemas, the old and the new one. This can be loaded either from the database system directly or an XML file already containing the metadata in specified format.

Next, the `diff` module of the Schemagic tool is invoked that takes two parameters – the old and the new schema. These schemas are compared by the diff algorithm implemented in the module and the diff-model is the result of the comparison process. Each diff-node from the diff-model contains a couple of db object instances, one from the old schema and the other is from the new schema. Most of diff-nodes will contain both db objects, which means that these db objects refer to the same piece of the relational schema. Since there is only one change the diff-model contains one diff-node that contains only one db object (the new albumin variable is the only change made to the database model in this example) holding the `table_column` named albumin.

Finally, the `sync` module of the Schemagic tool processes the diff model and applies predefined XSL templates on it. During this process no action is performed on diff-nodes containing both parts (old and new db object) only the diff-node with one new db objects results to the following synchronising SQL script:

Listing 7.1: synchronisation SQL script altering schema `ehr_schema` to add an `albumin` column to table `lab_exam`.

```
ALTER TABLE ehr_schema.lab_exam ADD (albumin NUMBER);
```

7.4.3 Synchronisation of Knowledge Base Trees of MUDR EHR

Definitions

In the process of Schemagic configuration files creation defining specific feartures of the MUDR EHR, the SchemagicCenter was utilised. Database objects representing MUDR Knowledge Base components were defined as follows: `domain`, `node`, `lang_term` and `edge`.

The Listings 7.2 represents a db machine formalising the MUDR KB storage. It contains a `mudr_xml` connection, hierarchy tree named `knowledge_tree` and a list of db object loaders.

Listing 7.2: MUDR machine definition in Schemagic tool.

```
<?xml version="1.0"?>
<dbmachine-def id="mudr" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="../xsd/dbmachine_def.xsd">
 <connection-ref ref="mudr_xml"/>
 <model-def>
  <hierarchy-forest>
   <hierarchy-tree id="knowledge_tree">
    <object-ref type="domain">
     <object-ref type="node">
      <object-ref type="lang_term"/>
      <object-ref type="edge"/>
     </object-ref>
    </object-ref>
   </hierarchy-tree>
  </hierarchy-forest>
 </model-def>

 <object-load object_type="domain" class_name="org.schemagic.plugin.database.load.impl.MudrObjectLoader">
  <parameters>
```

```
   <parameter name="XSL_TEMPLATE" value="def/machine/mudr.domain.load.xsl"/>
   <parameter name="INITIAL_MUDR_COMMAND" value="&lt;command␣id=&quot;schm1&quot;&gt;&lt;get_knowledge_node&gt;&lt;node_name&gt;:global[@LOADED_DOMAIN]:&lt;/node_name&gt;&lt;/get_knowledge_node&gt;&lt;/command&gt;␣
&lt;command id=&quot;schm2&quot;&gt;&lt;get_knowledge_domains/&gt;&lt;/command&gt;"/>
   <parameter name="TERMINAL_CONDITION" value="KNOWLEDGE_DOMAIN_OK"/>
   <parameter name="SAX_PARSER_CLASS_NAME" value="org.schemagic.plugin.database.load.impl.MudrNodeSaxParser"/>
  </parameters>
  <pre-processing class_name="org.schemagic.plugin.database.load.impl.MudrPreprocessor"/>
  <post-processing class_name="org.schemagic.plugin.database.load.impl.MudrDomainPostProcessor">
   <parameters>
    <parameter name="NODE_NAME" value="DOMAIN_NAME"/>
    <parameter name="CREATE_EDGE_FULL_NAMES" value="" />
   </parameters>
  </post-processing>
 </object-load>
 <object-load object_type="node" class_name="org.schemagic.plugin.database.load.impl.MudrObjectLoader">
  <parameters>
   <parameter name="XSL_TEMPLATE" value="def/machine/mudr.node.load.xsl"/>
   <parameter name="INITIAL_MUDR_COMMAND" value="&lt;command␣id=&quot;c1&quot;&gt;&lt;get_knowledge_node&gt;&lt;node_id&gt;:domain[@NODE_ID]:&lt;/node_id&gt;&lt;/get_knowledge_node&gt;&lt;/command&gt;"/>
   <parameter name="TERMINAL_CONDITION" value="&lt;get_knowledge_node/&gt;"/>
   <parameter name="SAX_PARSER_CLASS_NAME" value="org.schemagic.plugin.database.load.impl.MudrNodeSaxParser"/>
   <parameter name="FULL_POST_PROCESSOR" value="org.schemagic.plugin.database.load.impl.MudrFullNameCreator"/>
  </parameters>
  <pre-processing class_name="org.schemagic.plugin.database.load.impl.MudrPreprocessor"/>
 </object-load>
 <!-- object-load elements for "lang_term" and "edge" were omitted due to lack of space -->
</dbmachine-def>
```

Implementation of Schemagic Interfaces

The mudr_xml connection definition contains a factory class org.schemagic.plugin.database.connection.http.MudrXmlConnectionFactory which implements the interface org.schemagic.plugin.database.connection.IDbConnectionFactory. This factory creates a HTTP connection wrapper called org.schemagic.plugin.database.connection.http.MudrXmlConnection. This class provides following methods:

- `public void Send (String commands)` – sends commands to MUDR EHR compliant with MUDR XML communication API (see [13] p. 78 for the list of functions).

- `public String Receive ()` – since the communication with MUDR EHR is stateless, this function is necessary to get results from the MUDR corresponding to requests sent by the `Send()` method.

The loading process of components of MUDR KB utilised all features that the Schemagic tool offers in object loading definition. First, a loader class had to be specified and implemented (`org.schemagic.plugin.database.load.impl.MudrObjectLoader`). Single loader class was implemented for all types of db objects representing MUDR KB tree components. As a result, following parameters of the loader class were developed:

- `INITIAL_MUDR_COMMAND` – is a XML command from the MUDR XML communication API, which initiates the loading process of the given db object from the MUDR EHR.

- `XSL_TEMPLATE` – its value is a path to an XSL template file which is used to process responses from the MUDR EHR and eventually creating new requests for loading other nodes. As a result of this, the MUDR KB tree is loaded in layers. In order to stop this process a ”no-data” mark from the server has to be recognised. Such a stop condition is defined in the next parameter.

- `TERMINAL_CONDITION` – when the response from the MUDR EHR contains this string, it means that the communication has reached the end.

- `SAX_PARSER_CLASS_NAME` – is a class for an XML SAX parser specific to each MUDR db object, which parses the response of the MUDR EHR during the loading process and stores properties' values of given MUDR db object.

- `FULL_POST_PROCESSOR` – is a class that computes a full name of a `node` db object. The full name is a dot-separated list of nodes' names laying on the path from the root to the given node. This class is invoked after the given `node` has been loaded, which implies all predecessor nodes have been already loaded, and it is possible to compute the full name.

It was also necessary to add a pre-processor class (`org.schemagic.plugin.database.load.impl.MudrPreprocessor`), which is invoked directly before the loader's method `public IDbObject[] load(Map allParameters, IDbConnection conn, IProgressListener pl)` is called. The pre-processor is used to modify input parameters for the loader class. In this case the `global[@LOADED_DOMAIN]` variable in Listing 7.2 on line 20 was replaced by the actual name of a domain to load from MUDR EHR. This value was specified in the schema handle file (see Listing 7.3, line 18).

Listing 7.3: A schema handle for `PATIENT` MUDR KB tree.

```
 1 <?xml version="1.0"?>
 2 <schema-handle id="mudr1_sch" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
       instance"
 3  xsi:noNamespaceSchemaLocation="../def/xsd/schema_handle.xsd">
 4   <use-dbmachine id="mudr"/>
 5   <use-connection id="mudr_xml"/>
 6   <model-load class_name="org.schemagic.plugin.database.load.impl.
        DbModelLoader"/>
 7   <parameters>
 8     <connection id="mudr_xml">
 9       <properties>
10         <property name="URL" value="http://10.0.0.6:6003/cgi-bin/ehrcgi2.0xp.
               exe?MUDR2.0"/>
11         <property name="FIRST_NAME" value="a"/>
12         <property name="FAMILY_NAME" value="a"/>
13         <property name="PASSWORD" value="a"/>
14       </properties>
15     </connection>
16     <object id="global">
17       <properties>
18         <property name="LOADED_DOMAIN" value="PATIENT"/>
19       </properties>
20     </object>
21     <object id="domain">
22       <properties>
23         <property name="DOMAIN_NAME" value="PATIENT"/>
24       </properties>
25     </object>
26   </parameters>
27 </schema-handle>
```

Loading a db object `domain` required a definition of a post-processor class `org.schemagic.plugin.database.load.impl.MudrDomainPostProcessor`. A post-processor is intended to be used after the loading process of given db object finished. It is possible to modify loaded data of given db object or its inferior nodes (according to the structure defined in the db machine – see Listing 7.2 line 6). In this case it was used to rename the property `NODE_NAME` to `DOMAIN_NAME` for a `domain` db object which is technically an ordinary knowledge tree node. Another task which was implemented by this post-process was adding a value to a key property `TGT_NODE_FULL_NAME` of the `edge` db object. Since the MUDR EHR does not provide full names of the target node the edge is connected to. This had to be computed after having the whole MUDR KB tree loaded.

7.4.4 Structural Comparisons of openEHR Archetypes

Definitions

The editors of various configuration files implemented in the SchemagicCenter were utilised to create XML definition files for the Schemagic tool in the same way as it was done during the implementation of MUDR KB extension.

Since an ADL file was considered as the source of an archetype, following db objects were defined with respect to Archetype Object Model (AOM): `archetype`, `description`, `original_author`, `details`, `element`, `definition`, `complex_object`, `single_attribute`, `multiple_attribute`, `archetype_slot`, `archetype_internal_ref`, `primitive_object`, `dv_quantity`, `dv_quantity_item`, `assumed_value`, `code_pharse`, `unrecognized_object`, `ontology`, `term_definitions`, `language`, `item`, `term_bindings`, `ontology_binding` and `binding_item`.

Next, a db machine named "archetype machine" was defined (see Listing 7.4). This machine is "connected" to connection defined as "archetype" (`connection-ref` element on line 3). Db objects describing archetype's components are ordered in a hierarchy represented by a hierarchy tree named `archetype_structure` (`hierarchy-tree` element on line 6). This hierarchy tree corresponds to AOM pats of which are shown in Figure 7.9 and Figure 7.9.

Listing 7.4: Archetype machine definition in Schemagic tool.

```
1 <?xml version="1.0"?>
2 <dbmachine-def id="archetype" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
     instance" xsi:noNamespaceSchemaLocation="../xsd/dbmachine_def.xsd">
3  <connection-ref ref="archetype"/>
4  <model-def>
```

```
 <hierarchy-forest>
  <hierarchy-tree id="archetype_structure">
   <object-ref type="archetype">
    <object-ref type="description">
              <object-ref type="original_author"/>
              <object-ref type="details">
                 <object-ref type="element"/>
     </object-ref>
    </object-ref>
    <object-ref type="definition">
     <object-ref type="complex_object"/>
     <object-ref type="single_attribute"/>
     <object-ref type="multiple_attribute"/>
     <object-ref type="archetype_slot"/>
     <object-ref type="archetype_internal_ref"/>
     <object-ref type="primitive_object"/>
     <object-ref type="dv_quantity">
      <object-ref type="dv_quantity_item"/>
      <object-ref type="assumed_value"/>
     </object-ref>
     <object-ref type="code_pharse"/>
     <object-ref type="unrecognized_object"/>
    </object-ref>
    <object-ref type="ontology">
     <object-ref type="term_definitions">
      <object-ref type="language">
       <object-ref type="item"/>
      </object-ref>
     </object-ref>
     <object-ref type="term_bindings">
      <object-ref type="ontology_binding">
       <object-ref type="binding_item"/>
      </object-ref>
     </object-ref>
    </object-ref>
   </object-ref>
  </hierarchy-tree>
 </hierarchy-forest>
</model-def>
<object-load object_type="archetype_structure" class_name="org.schemagic.plugin.database.load.impl.ArchetypeObjectLoader">
 <parameters>
  <parameter name="NAME" value="archetype_structure"/>
 </parameters>
</object-load>
<object-load object_type="archetype" class_name="org.schemagic.plugin.database.load.impl.ArchetypeObjectLoader">
 <parameters>
  <parameter name="NAME" value="archetype"/>
 </parameters>
</object-load>
<object-load object_type="description" class_name="org.schemagic.plugin.database.load.impl.ArchetypeObjectLoader">
 <parameters>
  <parameter name="NAME" value="description"/>
 </parameters>
</object-load>
```

```
 <object-load object_type="definition" class_name="org.schemagic.plugin.database.load.impl.ArchetypeObjectLoader">
  <parameters>
   <parameter name="NAME" value="definition"/>
  </parameters>
 </object-load>
 <object-load object_type="multiple_attribute" class_name="org.schemagic.plugin.database.load.impl.ArchetypeObjectLoader">
  <parameters>
   <parameter name="NAME" value="multiple_attribute"/>
  </parameters>
 </object-load>
 <object-load object_type="complex_object" class_name="org.schemagic.plugin.database.load.impl.ArchetypeObjectLoader">
  <parameters>
   <parameter name="NAME" value="complex_object"/>
  </parameters>
 </object-load>
   <object-load object_type="single_attribute" class_name="org.schemagic.plugin.database.load.impl.ArchetypeObjectLoader">
  <parameters>
   <parameter name="NAME" value="single_attribute"/>
  </parameters>
 </object-load>
    <!-- other object-load elements were left out for clarity reasons -->
</dbmachine-def>
```

At the end of the db machine definition file a list of object loaders is present. Each db object must have its corresponding object loader definition, which comprises a name of the given db obejct, a class name of the loader itself and a list of parameters used by the loader class. In this solution only one class named `ArchetypeObjectLoader` was implemented, which is able to load all kind of archetype related db objects. It takes only one parameter called `NAME` to distinguish which type of db object to load.

Implementation of Schemagic Interfaces

Archetype comparison support in the Schemagic tool involved implementation of some interfaces defined by Schemagic.

Following classes were developed:

- `ArchetypeXmlConnection` implementing interface `org.schemagic.plugin.database.connection.IDbConnection` – in this class the archetype instance from the ADL file is loaded. This archetype is loaded during the construction of this class using `se.acode.openehr.parser.ADLParser`, which was implemented in the frame of openEHR Java Reference Implementation Project [76].

- `ArchetypeXmlConnectionFactory` implementing interface `org.schemagic.plugin.database.connection.IDbConnectionFactory` – one mandatory method of this interface was implemented, namely `IDbConnection createDbConnection( Properties props )`, which returns an instance of an `ArchetypeXmlConnection` class configured according to the properties `props`. This factory is implemented according to Singleton design pattern, i.e. there exist only one instance of this factory.

- `ArchetypeObjectLoader` implementing interface `org.schemagic.plugin.database.load.IDbObjectLoader` – the following method required by the interface was implemeneted: `public IDbObject[] load(Map param, IDbConnection conn, IProgressListener listener)`. The loader class loads given archetype component whose name is passed as a parameter in `param` argument. The archetype components' names are in conformity with the Archetype Object Model, parts of which are depicted in Figure 7.8 and Figure 7.9.

Comparison of Two Archetype ADL Instances

The comparison process is performed on two instances of archetypes represented in form of database objects defined on page 94 and sorted in structure defined in archetype machine (see `hierarchy-tree` element on line 6 in Listing 7.4). Each db object must contain one property that is marked as a key property. A key property of an object is used in the comparison process. The Schemagic tool allows to customise the comparison process (implemented by the `org.schemagic.plugin.diff` plugin) by defining suitable comparators for given db objects. In this solution a universal comparator class was used (`KeyComparator`), which compares the key properties as strings.

The result of the comparison is a diff-tree composed of diff-nodes (for an example see Figure 7.6 on page 87). Each diff-node contains a couple of db object instances, one from the old compared formal model and the other from the new formal model (here it is loaded archetype). If the new db object in a diff-node is not present, i.e. its value is

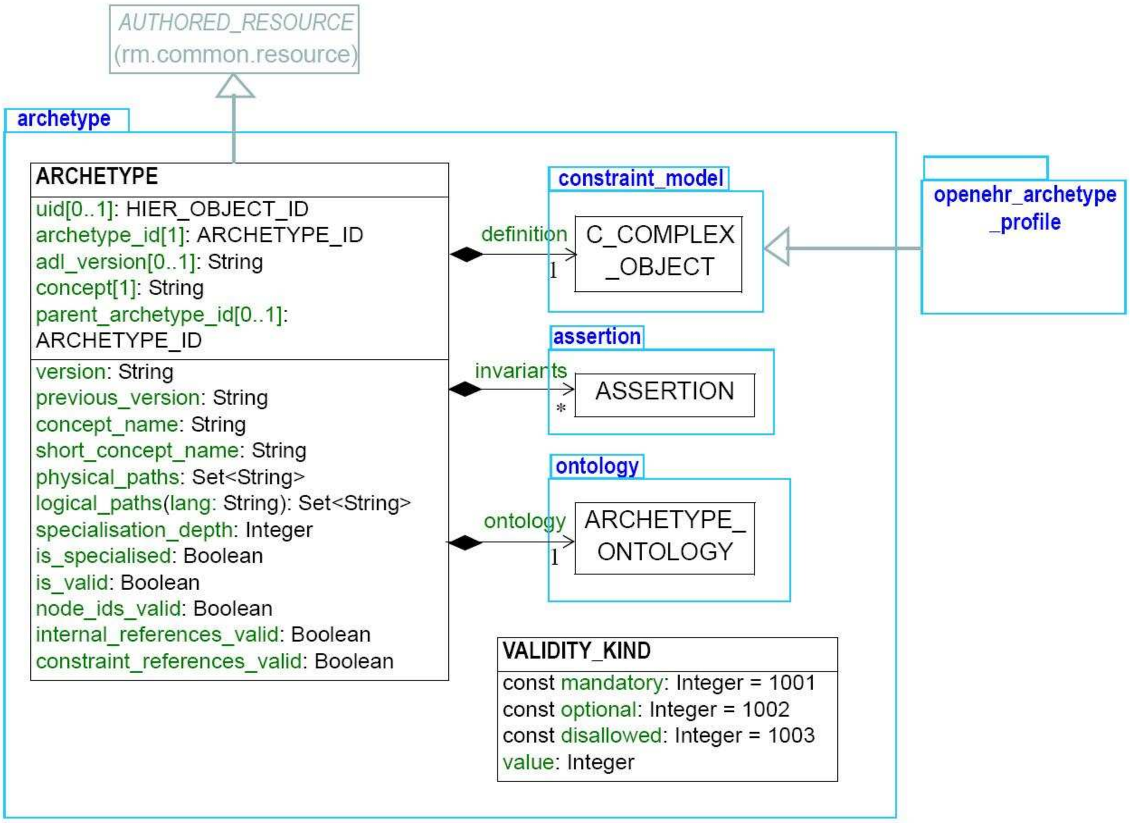

AUTHORED_RESOURCE
(rm.common.resource)
archetype
ARCHETYPE
uid[0..1]: HIER_OBJECT_ID
archetype_id[1]: ARCHETYPE_ID
adl_version[0..1]: String
concept[1]: String
parent_archetype_id[0..1]:
ARCHETYPE_ID
version: String
previous_version: String
concept_name: String
short_concept_name: String
physical_paths: Set<String>
logical_paths(lang: String): Set<String>
specialisation_depth: Integer
is_specialised: Boolean
is_valid: Boolean
node_ids_valid: Boolean
internal_references_valid: Boolean
constraint_references_valid: Boolean
constraint_model
definition
1
C_COMPLEX
_OBJECT
assertion
invariants
*
ASSERTION
ontology
ontology
1
ARCHETYPE_
ONTOLOGY
VALIDITY_KIND
const mandatory: Integer = 1001
const optional: Integer = 1002
const disallowed: Integer = 1003
value: Integer
openehr_archetype
_profile

Figure 7.8: openehr.am.archetype Package (taken from [77]).

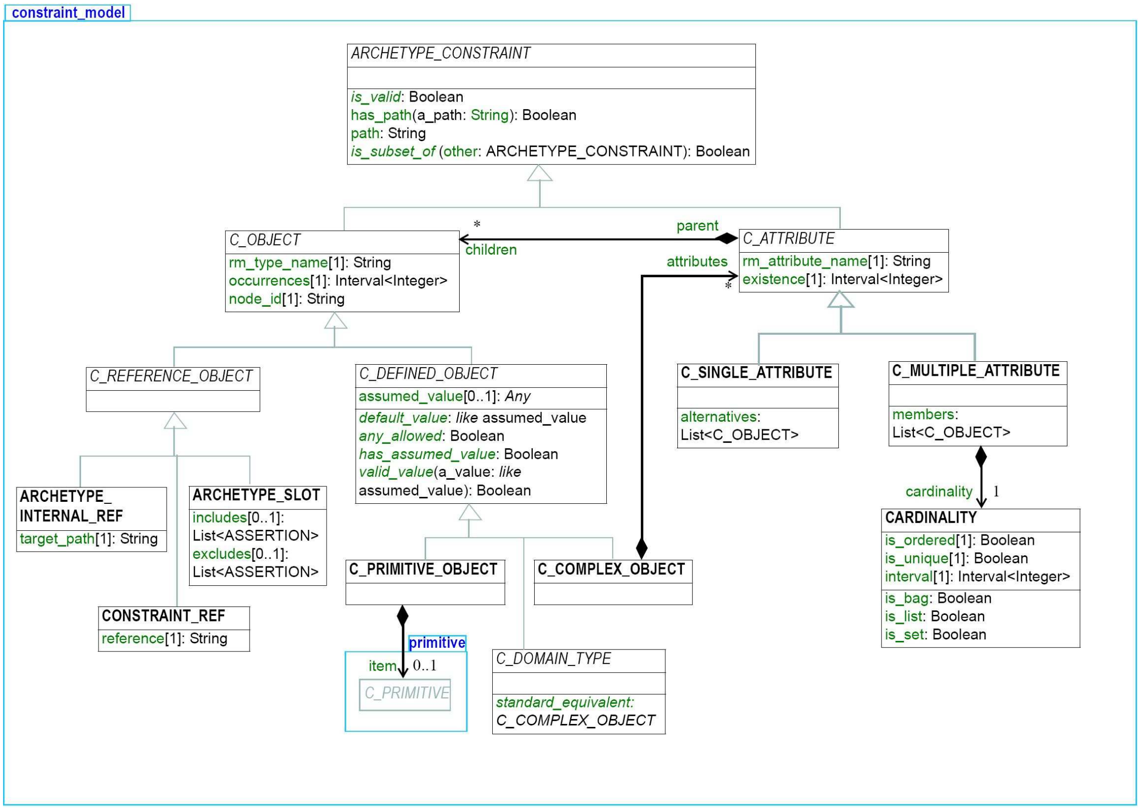


Figure 7.9: openehr.am.archetype.constraint_model Package (taken from [77]).

null, it means that the old db object has no matching object in the new model, thus it should be left out. On the other hand, if the old db object is null in a diff-node, the new db object did not appear in the old model, therefore, it should be created. In case both db objects are present in a diff-node, they match each other and no action is necessary, because there was no difference found between the old and the new model at this point.

Since there is no aparatus to synchronise archetypes in ADL, the Schemagic tool was extended only to support loading and comparison of archetypes stored in ADL files. The resulting diff-model is stored in an XML file for further processing once a manipulating language for archetypes is introduced (with similar functionality to the SQL DML, Data Manipulation Language).

7.5 Semantic Interoperability Platform based on HL7 v3 Messages

In this section an approach to semantic interoperability among heterogeneous EHR systems based on HL7 v3 standard is described. The research presented here was done in the frame of the ITDCSH project (see p. 29).

The primary result of the ITDCSH project was a proposal of a semantic interoperability platform (Figure 7.10) based on the international communication standard.

In [31] the author describes the difference between the human semantic interoperability and computable semantic interoperability (CSI), and states four necessary but not the only conditions (the "four pillars") for establishing the CSI:

1. a common information model that spans all domains of interest,
2. unambiguous semantics of each transferred data element defined by its data type,
3. usage of vocabulary – binding the information models to domain-specific terminologies,
4. a formal top-down message development process with restricted optionality.

To meet the conditions of semantic interoperability (1.-4.), the following steps were performed:

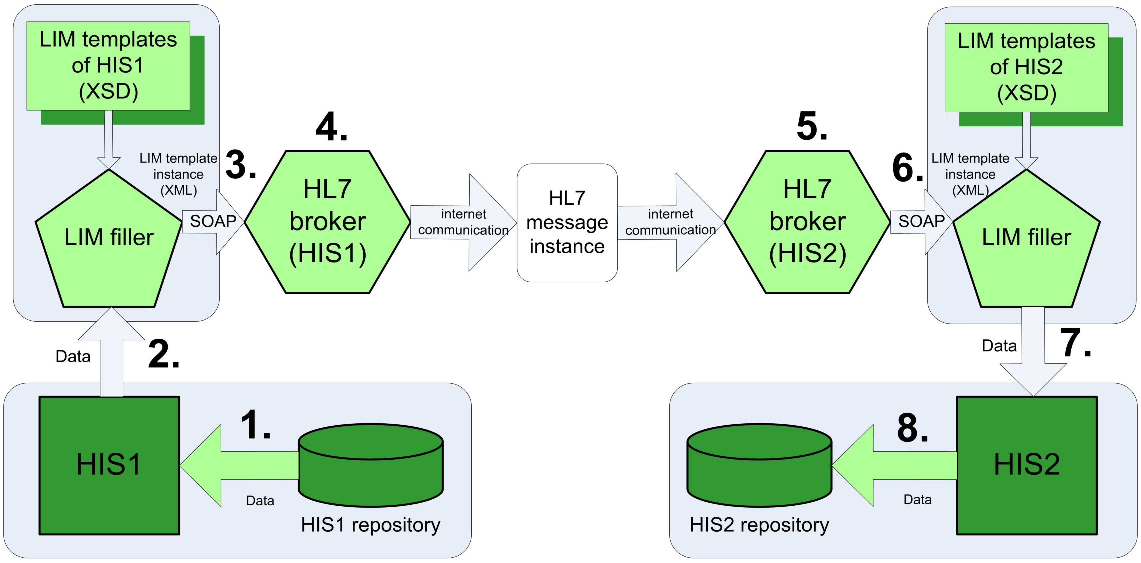


Figure 7.10: Proposal of a semantic interoperability platform based on the international communication standard.

1. The first condition addresses the usage of a common model. The ”world of HL7” introduces the RIM. In this solution, the information stored in the incorporated EHRs was modeled by creating the Local Information Models (LIMs). Its classes are derived from the core classes of RIM. This approach represents a slight diversion from the general HL7 v3 message development process, which is discussed later.

2. The second condition was fulfilled by using the data types defined by HL7 v3 for all attributes in the LIM model. An important role was played by the data type ”coded with equivalents” (CE) and its derived subtypes, allowing us to connect the concepts from our model with given code-lists.

3. The third condition requires the usage of vocabulary. Concepts from LIMs (based on MDMC) had to be mapped to a code-list that is supported by HL7. The Logical Observation Identifier Names and Codes (LOINC) [45] coding system was used as the preferred one, since it contained a significant subset of concepts used in LIM models. The remaining concepts were encoded by the SNOMED Clinical Terms (SNOMED CT) [12] and ICD-10 [78].

4. The last condition requires using a strict methodology. HL7 v3 provides methods for defining data interchange structures using only RIM elements bound to domain-specific values. After preparing the LIM models and their binding to the coding systems, tree-like structures from the LIM (LIM templates) had to be derived as the next step in the process of the message definition. Each LIM template represents one integrated part of the EHR the LIM describes, e.g. physical examination, medication and ECG data. The HL7 HMD [35] and LIM templates serve the same purpose, i.e. they define the hierarchy, sequence and optionality of classes specified in HL7 R-MIM (Refined Message Information Model) and the LIM, respectively.

The steps summarised above resulted in developing a proposal of SIP, which was the main result of the ITDCSH project. The proposal consists of the LIM filler module, HL7 broker and original HISes. Numbers in Figure 7.10 represent the data flow in a situation when HIS1 sends data to HIS2. First, the requested data are gathered from HIS1. This is done by the LIM filler that has a connection to the HIS repository. Next, the LIM filler takes the suitable LIM template which contains the correct concepts to represent the communicated data. The LIM filler asigns data values to unassigned member variables of classes in the LIM template, thus creating a LIM message. The HL7 broker receives the LIM message via the SOAP protocol used by the LIM filler module. Again, another transformation is performed; in this case the HL7 broker produces appropriate HL7 message instances, which are sent in a secure way to the receiving HL7 broker. There the process of data transformation runs backwards. The HL7 broker attached to HIS2 creates LIM messages supported by the LIM filler of HIS2 and sends them via SOAP. The receiving LIM filler recognizes the incoming LIM message and extracts the data into a form suitable for storage in the HIS2 repository. Finally, the data are stored in HIS2. In this example the requesting and confirmation mechanisms were left out for simplicity reasons.

LIM fillers and HL7 broker components were developed to support data transformations of a given HIS. Both components will be described in more detail in the following text.

Local Information Model (LIM) is an information model representing the content of a given EHR system. This information model derives its classes from those defined in HL7 v3 RIM (Reference Information Model see p. 39).

LIM template is a pattern defining the tree-like structure of instances of LIM classes. Each LIM template represents one integrated part of the EHR system the LIM describes, e.g. physical examination, medication and ECG data.

7.5.1 The LIM Filler Module

Transferred data are converted between the local EHR storage and a LIM message, which is an instance of a LIM template, on each side of the communication. This task is performed by the module called the LIM filler. The LIM filler is adjusted for each local EHR-S to produce LIM messages according to the local EHR structure.

The LIM filler module can be an EHR-S plug-in or a standalone application. It works in two modes: sending and receiving. In the sending mode, it creates LIM messages on user's demand and sends them to the HL7 broker. In the second mode, it polls the HL7 broker for new messages. In case of a new message downloaded, it extracts the data from the message and acts according to the particular storyboard or just stores the data of the patient in the local EHR.

The LIM filler must respect the security aspects of the communication protocol. It communicates with the HL7 broker through the secured HTTP channel using the SOAP protocol. The LIM messages must be digitally signed by both parties involved and the signatures must be checked before extracting the data from the LIM message.

7.5.2 The HL7 Broker

Although the market offers several HL7 v2 brokers (e.g. MergeCOM-3^{TM}, Interfaceware Chameleon^{TM}), in the early phase of the project (in 2005) no HL7 v3 broker was available, therefore a new one was implemented for our solution. The HL7 broker is a fundamental part of our solution. For the EHR-S, the attached HL7 broker serves as a gateway to the world of HL7 v3 and separates it from details of HL7.

The vendors of the EHR-S only need to specify the LIM templates based on their

LIM models describing their database structure and to implement a simple SOAP client to push, poll and process LIM messages from/to the HL7 broker. Besides the LIM templates, the vendors must upload all internally used enumerations with appropriate descriptions to our web-based vocabulary mapping tool.

The HL7 broker counterbalances the implementation effort between the EHR-S vendors and the HL7 broker producers. The easier the transformations on EHR system's side are, the more work needs to be done on the HL7 broker's side. The HL7 broker needs to be configured specifically for its EHR-S. It was necessary to preload all LIM templates (as XSD schemata of all the LIM messages) defined by the vendors. For each HL7 storyboard the HL7 broker requires processing instructions (i.e., what to do when the specific LIM message arrives, when the HL7 Application Role is triggered or when an error occurs). It was also necessary to map the EHR's enumerations to the HL7 Vocabulary or other nomenclatures.

The core of the HL7 broker is responsible for the transcription of the information from LIM to HL7, and vice versa. It requires processing instructions in the XML language involving the helper class.

The communication between the EHR-S and the HL7 broker was implemented using Web Services based on the SOAP over the HTTPS protocol. The HL7 broker provides only three methods (`sendLimMsg()`, `getLimMsg()`, `ackLimMsg()`) for transfer of data in the form of LIM messages.

A simple "push to broker" and "periodically poll the broker" communication schema was introduced, which enabled the implementation of the SOAP server only on the broker side and the SOAP client only on the EHR-S side. The SOAP client is typically implemented in a simpler and cheaper way than the SOAP server. Developers can also track all the communication between the EHR-S and the HL7 broker with a tracking application written in Java. This has proved to be very useful, especially in the initial phase of the implementation of the SOAP client in the EHR-S. Along with the transmitted LIM message, a cryptographically signed digest of the message is transmitted and confirmed, thus each side has proved authenticity of the transmission.

The HL7 broker may be invoked by the EHR-S call or by the HL7 v3 counterpart.

According to the LIM message type or the HL7 interaction ID combined with the LIM session ID or the HL7 Message Response ID, the broker will decide whether it is the beginning of a new session (and which storyboard to start with), or whether the session is already running. Afterwards, an appropriate action is chosen (e.g. assembling a message or raising an exception).

While transcribing the information from the received message to the new one, the core uses heavily the helper class and the associated Vocabulary mapping service. The process of the transcription is logged and the result of the transcription can be propagated back to the sender (Control LIM message or HL7 ACK message).

The transport of HL7 v3 messages is based on simple imitation of MLLP (Minimal Lower Layer Protocol) over SOAP. HL7 messages are serialized according to XML ITS R1.

7.5.3 Examples of Implemented Storyboard, LIM and HL7 Messages

Querying in our solution is based on the HL7 storyboards. As an example of such a storyboard, one from the ”Patient Administration” domain called ”Patient Registry Find Candidates Query” (artifact code PRPA_ST201305) was implemented to search for the patient's administrative data. A UML sequence diagram representing the activities performed according to the ”patient query” storyboard with an added ADAMEKj query and WinMedicalc responses is shown in Figure 7.11.

Queries that are produced by the incorporated EHR-Ss and passed to the HL7 broker are composed of ”empty” LIM templates with only some attributes containing values, which are recognized as the parameters of a query.

Our solution enables composing the queries to all the domains covered by the LIM templates. The HL7 broker executes the appropriate storyboard that leads to the acquisition of the data queried by the LIM filler module.

In order to consolidate the reader's apprehension of the HL7 messaging and the queries described in the previous text, an example message is given, which corresponds to a ”Patient Registry Find Candidates Response” interaction (artifact code

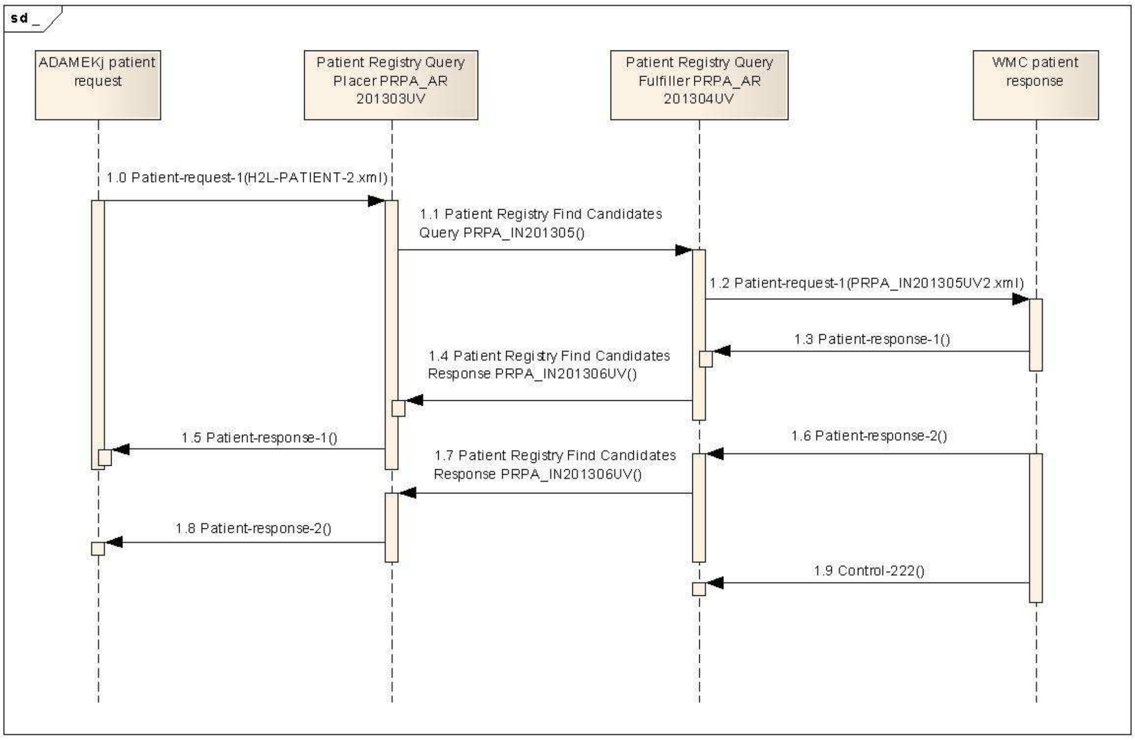
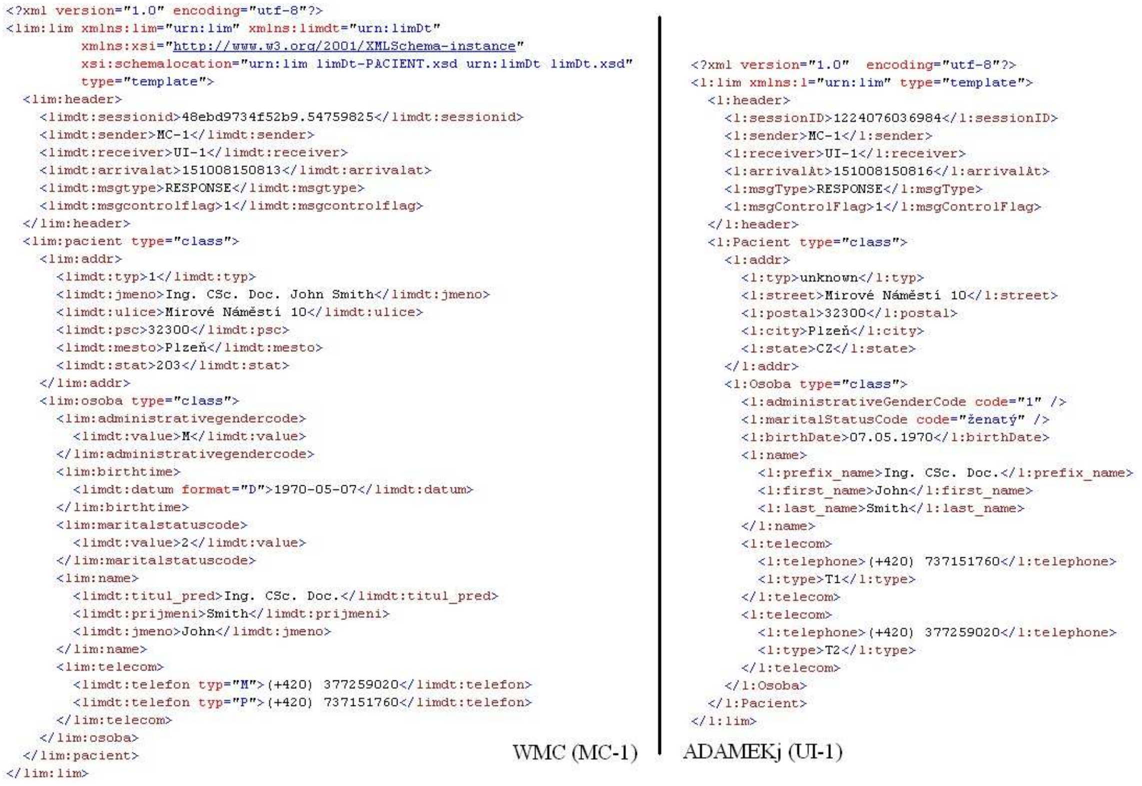

sd _
ADAMEKj patient request
Patient Registry Query Placer PRPA_AR 201303UV
Patient Registry Query Fulfiller PRPA_AR 201304UV
WMC patient response
1.0 Patient-request-1(H2L-PATIENT-2.xml)
1.1 Patient Registry Find Candidates Query PRPA_IN201305()
1.2 Patient-request-1(PRPA_IN201305UV2.xml)
1.3 Patient-response-1()
1.4 Patient Registry Find Candidates Response PRPA_IN201306UV()
1.5 Patient-response-1()
1.6 Patient-response-2()
1.7 Patient Registry Find Candidates Response PRPA_IN201306UV()
1.8 Patient-response-2()
1.9 Control-222()

Figure 7.11: A sequence diagram of the Patient Registry Find Candidates Query.



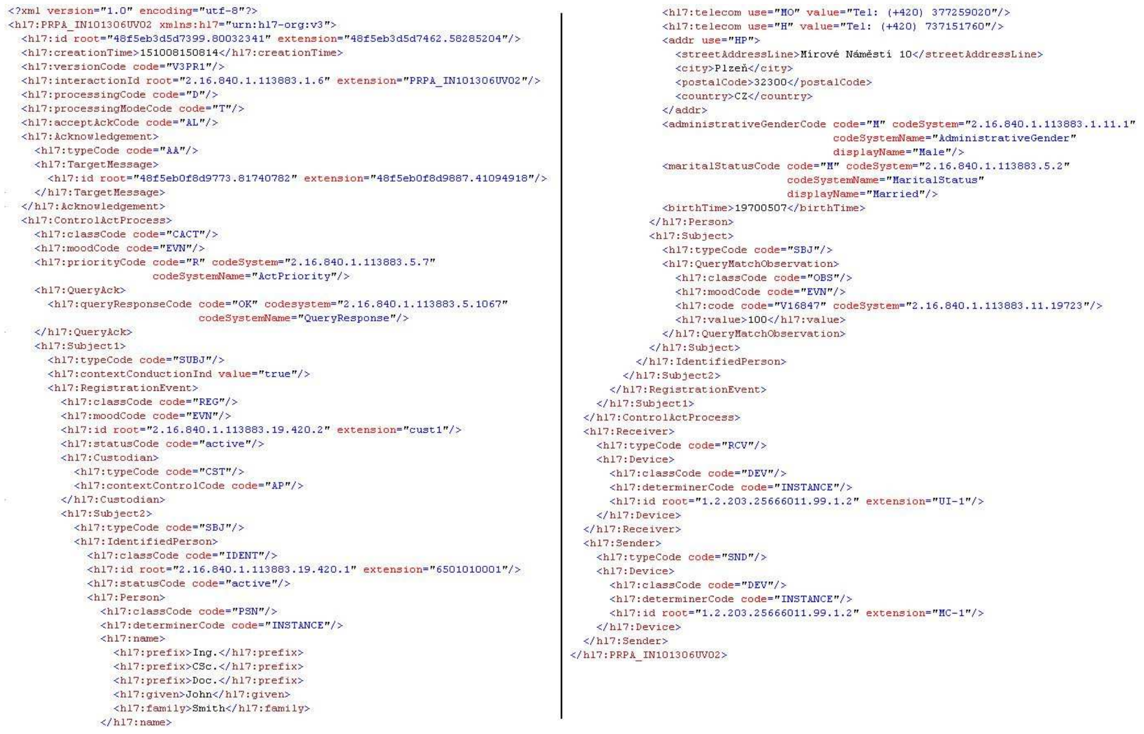
```
<?xml version="1.0" encoding="utf-8"?>
<lim:lim xmlns:lim="urn:lim" xmlns:limdt="urn:limDt"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemalocation="urn:lim limDt-PACIENT.xsd urn:limDt limDt.xsd"
         type="template">
  <lim:header>
    <limdt:sessionid>48ebd9734f52b9.54759825</limdt:sessionid>
    <limdt:sender>MC-1</limdt:sender>
    <limdt:receiver>UI-1</limdt:receiver>
    <limdt:arrivalat>151008150813</limdt:arrivalat>
    <limdt:msgtype>RESPONSE</limdt:msgtype>
    <limdt:msgcontrolflag>1</limdt:msgcontrolflag>
  </lim:header>
  <lim:pacient type="class">
    <lim:addr>
      <limdt:typ>1</limdt:typ>
      <limdt:jmeno>Ing. CSc. Doc. John Smith</limdt:jmeno>
      <limdt:ulice>Mirové Náměstí 10</limdt:ulice>
      <limdt:psc>32300</limdt:psc>
      <limdt:mesto>Plzeň</limdt:mesto>
      <limdt:stat>203</limdt:stat>
    </lim:addr>
    <lim:osoba type="class">
      <lim:administrativegendercode>
        <limdt:value>M</limdt:value>
      </lim:administrativegendercode>
      <lim:birthtime>
        <limdt:datum format="D">1970-05-07</limdt:datum>
      </lim:birthtime>
      <lim:maritalstatuscode>
        <limdt:value>2</limdt:value>
      </lim:maritalstatuscode>
      <lim:name>
        <limdt:titul_pred>Ing. CSc. Doc.</limdt:titul_pred>
        <limdt:prijmeni>Smith</limdt:prijmeni>
        <limdt:jmeno>John</limdt:jmeno>
      </lim:name>
      <lim:telecom>
        <limdt:telefon typ="M">(+420) 377259020</limdt:telefon>
        <limdt:telefon typ="P">(+420) 737151760</limdt:telefon>
      </lim:telecom>
    </lim:osoba>
  </lim:pacient>
</lim:lim>
```

WMC (MC-1)

```
<?xml version="1.0"  encoding="utf-8"?>
<l:lim xmlns:l="urn:lim" type="template">
  <l:header>
    <l:sessionID>1224076036984</l:sessionID>
    <l:sender>MC-1</l:sender>
    <l:receiver>UI-1</l:receiver>
    <l:arrivalAt>151008150816</l:arrivalAt>
    <l:msgType>RESPONSE</l:msgType>
    <l:msgControlFlag>1</l:msgControlFlag>
  </l:header>
  <l:Pacient type="class">
    <l:addr>
      <l:typ>unknown</l:typ>
      <l:street>Mirové Náměstí 10</l:street>
      <l:postal>32300</l:postal>
      <l:city>Plzeň</l:city>
      <l:state>CZ</l:state>
    </l:addr>
    <l:Osoba type="class">
      <l:administrativeGenderCode code="1" />
      <l:maritalStatusCode code="ženatý" />
      <l:birthDate>07.05.1970</l:birthDate>
      <l:name>
        <l:prefix_name>Ing. CSc. Doc.</l:prefix_name>
        <l:first_name>John</l:first_name>
        <l:last_name>Smith</l:last_name>
      </l:name>
      <l:telecom>
        <l:telephone>(+420) 737151760</l:telephone>
        <l:type>T1</l:type>
      </l:telecom>
      <l:telecom>
        <l:telephone>(+420) 377259020</l:telephone>
        <l:type>T2</l:type>
      </l:telecom>
    </l:Osoba>
  </l:Pacient>
</l:lim>
```

ADAMEKj (UI-1)

Figure 7.12: LIM messages containing patient administrative data.



```xml
<?xml version="1.0" encoding="utf-8"?>
<hl7:PRPA_IN101306UV02 xmlns:hl7="urn:hl7-org:v3">
  <hl7:id root="48f5eb3d5d7399.80032341" extension="48f5eb3d5d7462.58285204"/>
  <hl7:creationTime>151008150814</hl7:creationTime>
  <hl7:versionCode code="V3PR1"/>
  <hl7:interactionId root="2.16.840.1.113883.1.6" extension="PRPA_IN101306UV02"/>
  <hl7:processingCode code="D"/>
  <hl7:processingModeCode code="T"/>
  <hl7:acceptAckCode code="AL"/>
  <hl7:Acknowledgement>
    <hl7:typeCode code="AA"/>
    <hl7:TargetMessage>
      <hl7:id root="48f5eb0f8d9773.81740782" extension="48f5eb0f8d9887.41094918"/>
    </hl7:TargetMessage>
  </hl7:Acknowledgement>
  <hl7:ControlActProcess>
    <hl7:classCode code="CACT"/>
    <hl7:moodCode code="EVN"/>
    <hl7:priorityCode code="R" codeSystem="2.16.840.1.113883.5.7"
                      codeSystemName="ActPriority"/>
    <hl7:QueryAck>
      <hl7:queryResponseCode code="OK" codesystem="2.16.840.1.113883.5.1067"
                             codeSystemName="QueryResponse"/>
    </hl7:QueryAck>
    <hl7:Subject1>
      <hl7:typeCode code="SUBJ"/>
      <hl7:contextConductionInd value="true"/>
      <hl7:RegistrationEvent>
        <hl7:classCode code="REG"/>
        <hl7:moodCode code="EVN"/>
        <hl7:id root="2.16.840.1.113883.19.420.2" extension="cust1"/>
        <hl7:statusCode code="active"/>
        <hl7:Custodian>
          <hl7:typeCode code="CST"/>
          <hl7:contextControlCode code="AP"/>
        </hl7:Custodian>
        <hl7:Subject2>
          <hl7:typeCode code="SBJ"/>
          <hl7:IdentifiedPerson>
            <hl7:classCode code="IDENT"/>
            <hl7:id root="2.16.840.1.113883.19.420.1" extension="6501010001"/>
            <hl7:statusCode code="active"/>
            <hl7:Person>
              <hl7:classCode code="PSN"/>
              <hl7:determinerCode code="INSTANCE"/>
              <hl7:name>
                <hl7:prefix>Ing.</hl7:prefix>
                <hl7:prefix>CSc.</hl7:prefix>
                <hl7:prefix>Doc.</hl7:prefix>
                <hl7:given>John</hl7:given>
                <hl7:family>Smith</hl7:family>
              </hl7:name>
              <hl7:telecom use="MO" value="Tel: (+420) 377259020"/>
              <hl7:telecom use="H" value="Tel: (+420) 737151760"/>
              <addr use="HP">
                <streetAddressLine>Mirové Náměstí 10</streetAddressLine>
                <city>Plzeň</city>
                <postalCode>32300</postalCode>
                <country>CZ</country>
              </addr>
              <administrativeGenderCode code="M" codeSystem="2.16.840.1.113883.1.11.1"
                                        codeSystemName="AdministrativeGender"
                                        displayName="Male"/>
              <maritalStatusCode code="M" codeSystem="2.16.840.1.113883.5.2"
                                 codeSystemName="MaritalStatus"
                                 displayName="Married"/>
              <birthTime>19700507</birthTime>
            </hl7:Person>
            <hl7:Subject>
              <hl7:typeCode code="SBJ"/>
              <hl7:QueryMatchObservation>
                <hl7:classCode code="OBS"/>
                <hl7:moodCode code="EVN"/>
                <hl7:code code="V16847" codeSystem="2.16.840.1.113883.11.19723"/>
                <hl7:value>100</hl7:value>
              </hl7:QueryMatchObservation>
            </hl7:Subject>
          </hl7:IdentifiedPerson>
        </hl7:Subject2>
      </hl7:RegistrationEvent>
    </hl7:Subject1>
  </hl7:ControlActProcess>
  <hl7:Receiver>
    <hl7:typeCode code="RCV"/>
    <hl7:Device>
      <hl7:classCode code="DEV"/>
      <hl7:determinerCode code="INSTANCE"/>
      <hl7:id root="1.2.203.25666011.99.1.2" extension="UI-1"/>
    </hl7:Device>
  </hl7:Receiver>
  <hl7:Sender>
    <hl7:typeCode code="SND"/>
    <hl7:Device>
      <hl7:classCode code="DEV"/>
      <hl7:determinerCode code="INSTANCE"/>
      <hl7:id root="1.2.203.25666011.99.1.2" extension="MC-1"/>
    </hl7:Device>
  </hl7:Sender>
</hl7:PRPA_IN101306UV02>
```

Figure 7.13: The dump of communication according to PRPA_ST201305 storyboard – the XML representation of a Patient Registry Find Candidates Response.

PRPA_IN201306) in the XML form (Figure 7.13) and the two associated LIM messages (Figure 7.12). The first LIM message (left side in Figure 7.12) serves as the data source of the LIM-HL7 transformation on the sender's side (e.g. WMC EHR-S); the second LIM message (right side in Figure 7.12) is the result of the HL7-LIM transformation done on the receiver's side (e.g. ADAMEKj). The presented messages contain administrative data of Mr. John Smith, who was subject to a query issued by the ADAMEKj system.

The proposal of a SIP platform aimed on creating fundamental principles for structured healthcare data exchange to enable national and cross-border communication. In this context a set of requirements was established:

- international communication standards and nomenclatures usage,
- automated interactions among systems – based on well-defined rules and data flows,

- clinical Domain independency – the clinical domain should not be hard-wired into the solution,
- configurability – the transformations of data stored in EHR into the form required by the selected communication standard should be defined in a configuration file, rather than the programming language,
- extensibility and openness – a new communicating system may be added in a convenient manner.

To adress the first requirement, the national standard was evaluated. Despite the exchange of the patient data between different healthcare institutions in the Czech Republic being realized mostly by the DASTA standard, it has limited functionality, completeness and with little regard to international efforts (standards and nomenclatures). Because of its bottom-up evolution, it does not contain an appropriate information model and has a low coverage of the healthcare domain (except for laboratory examinations). These reasons excluded this standard from putting into the standards' candidates list.

The candidates for the primary communication standard of the SIP were openEHR, ENV 13606 (EN 13606 was not available in 2004 yet), DICOM and HL7 v3. The openEHR is more suitable for building an EHR-S [38] rather than messaging, and furthermore, it was not as advanced as it is nowadays. The relationship between openEHR Specifications and HL7 v3 is described in [79]. The ENv 13606 had too few implementations in the real world and was in the gradual process of revision in a slightly different EN13606, which was not finished at that time. The DICOM is focused mainly on the imaging domain and was not suitable for the planned usage. As the best and the most promising solution, the HL7 v3 was chosen because of its wide international acceptance, an elaborate methodology [35], and a long and successful history of its previous version. This decision led to the satisfaction of the first criterion – the international communication standard and nomenclatures usage.

Figure 7.10 shows that both EHR-Ss communicate without any human intervention. They are controlled by queries placed by the user of EHR-S initiating communication. In this way, the **automated interactions between these systems** exist.

An important feature of our solution is the introduction of LIM models and LIM templates. The HL7 broker enables composing the queries to all the domains covered by the LIM templates. All the necessary information about the given clinical domain is covered by the configuration file of the HL7 broker. Consequently, the **clinical domain independency** and **configurability** criteria are fulfilled.

The **extensibility** of the solution was achieved by using the international communication standard and introducing the HL7 broker. The communicating systems do not need to know anything about the data structure of their counterparts. The new communicating EHR-S can be added after accomplishing the following steps: the creation of the LIM describing information stored in the EHR, deriving the LIM templates from the LIM model, the implementation of a transforming module (e.g. LIM filler) for data conversions between the EHR database and the LIM templates, and finally the creation of the HL7 broker's definition file specific to the new EHR-S. The **openness** of our solution was provided by making our results available to all interested parties.

7.6 Clinical Data Interoperability Based on the openEHR Approach

In the previous section, an approach to semantic interoperability platform based on HL7 v3 standard was presented. During the implementation process the complexity of the HL7 v3 standard emerged as a problem, which supported an idea of proposing an alternative interoperability platform based on an openEHR approach.

The experiment was conducted in order to compare the communication schema based on HL7 v3 messages with the data exchange using openEHR archetypes and extracts [80]. The main motivation for accomplishing such a comparison were the difficulties encountered during implementation of the communication among EHR-Ss using HL7 v3 messages. The main shortcoming was the usage of HL7 balloted storyboards. Storyboards describe the dynamic aspect of the communication and define the factual form and thus the content of messages is exchanged. And here emerges the main problem. HL7 storyboards are "short-message" oriented, which was not exactly matching our needs. The nature of our communication was rather document-oriented, which caused complicated transfer of LIM messages (e.g. physical examination) via several HL7 v3 messages originating in several storyboards.

The use of openEHR extracts on the other hand makes the data exchange straightforward. The data compliant with defined LIM templates were more suitable for openEHR extracts than for HL7 v3 messages, i.e. the task was to transfer ”documents” rather than short messages. Another possible solution would incorporate HL7 CDA documents [81], which might be a future work for the next research project.

The solution based on the openEHR approach had to fulfill the computable semantic interoperability conditions summarised on p. 100 in the same manner as the HL7 based solution did.

1. the openEHR archetypes used in this solution were gathered from the CKM repository, which only contains archetypes based on the openEHR reference model. This fulfills the first condition of using the common information model that spans all domains of interest.

2. The unambiguous semantics of each transferred data element was described by data types defined by the openEHR Foundation in the document ”Data Types Information Model” [82].

3. The usage of vocabulary was satisfied by utilising the mapping done on MDMC and several coding systems in Section 7.1.1. MDMC concept codes were used to fix the semantic meaning of concepts in an archetype via its ontology section.

4. A formal message development process is described in the ”Extract Information Model” [80]. Apparently the openEHR approach does not use messages, but rather so-called EHR extracts are used for transporting of clinical data among EHR-Ss.

According to [42] archetypes are distinct, structured models of domain concepts, such as ”blood pressure”. They sit between lower layers of knowledge resources in a computing environment, such as clinical terminologies and ontologies, and actual data in production systems. Their primary purpose is to provide a reusable, interoperable way of managing generic data so that it conforms to particular structures and semantic constraints. Consequently, they bind the terminology and ontology concepts to information model semantics, in order to make statements about what valid data structures look like. ADL provides a solid formalism for expressing, building and using these entities computationally. Every ADL archetype is written with respect to a particular information model, often known as a ”reference model”.

Archetypes are applied to data via the use of templates, which are defined at a local level. Templates generally correspond closely to screen forms, and may be re-usable at a local or regional level. Templates do not introduce any new semantics to archetypes, they simply specify the use of particular archetypes, and default data values.

The third artifact which governs the functioning of archetypes and templates at runtime is a local palette, which specifies which natural language(s) and terminologies are in use in the locale. The use of a palette removes irrelevant languages and terminology bindings from the archetypes, retaining only those relevant to actual use. Figure 7.14 illustrates the overall environment in which archetypes, templates, and a locale palette exist.

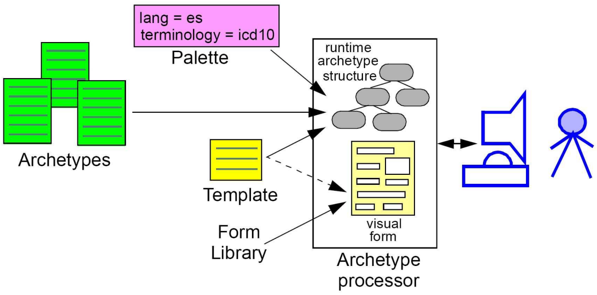


Figure 7.14: Archetypes, templates and palletes.

According to [83] templates include the following semantics:

- archetype ”chaining”: the choice of archetypes to make up a larger structure, specified via indicating identifiers of archetypes to fill slots in higher-level archetypes;
- local optionality: narrowing of some or all 0..1 constraints to either 1..1 (mandatory) or 0..0 (removal) according to local needs;
- tightened constraints: tightening of other constraints, including cardinality, value ranges, terminology value sets, and so on;

- default values: choice of default values for use in the templated structure at runtime.

At runtime, templates are used with archetypes to create data and to control its modification.

The main advantages [84] of the openEHR approach are the functional and semantical interoperability. The functional interoperability represents the correct communication between two or more systems. This is also covered by other approaches like HL7 v2.x. The openEHR approach also offers the semantic interoperability. It is the ability of two or more computer systems to exchange information which can be comprehended unambiguously by both, humans and computers.

In the following text steps leading to creation of an EHR with the Harmonised Clinical Content (HCC) will be proposed. Concepts of the clinical content of an EHR are usually "hiden" in the object model, database schema or in meta-models developed during the information system creation. These clinical concepts had to be extracted from the information models and upgraded to a higher level of modelling hierarchy, thus enabling the clinical content to be manipulated explicitly, allowing for a harmonisation claims. The process of enabling the creation of EHRs with HCC has following steps:

1. **map clinical concepts** to an international coding system or ontology (SNOMED CT, LOINC, etc.),

2. **find archetypes** in a repository or knowledge base that sufficiently cover encoded concepts,

3. **underlying reference model** may be openEHR RM or HL7 v3 RIM thanks to OWLmt Mapping Engine [85] that is capable of transforming one to another one by using pre-defined mappings.

An example of implementation of step 1 is shown in Table 7.3. In [46] the controlled and uncontrolled archetype development is described as well as techniques for ensuring maximal reusability of created constructs, curtailing their complexity and minimizing their number. Such a practice would perfectly support the second step. This step was

Description of encoded concept	**Code**	**Coding system**
Measurement of breath frequency in one minute	9279-1	LOINC
Measurement of heart beats in one minute	8893-0	LOINC
Measurement of blood temperature	8328-7	LOINC
Measurement of intravascular diastolic pressure	8462-4	LOINC
Amount of proteins in blood sample	2885-2	LOINC
Subjective complaints of the patient are described	10154-3	LOINC
Treatment of Ischemic Heart Disease	C0585894	UMLS CUI
Detection of Left ventricular hypertrophy	C0149721	UMLS CUI
Coughing after administration of ACE inhibitors	C0740723	UMLS CUI
Sequelae of cerebrovascular disease	I61	ICD10
Angina Pectoris	I20	ICD10
Hyperplasia of prostate	N40	ICD10

Table 7.3: Mapping concepts of MDMC to LOINC, UMLS and ICD-10 coding systems.

accomplished using the archetype repository [44] and some found archetypes are put down in Table 7.4 together with matching classes from the LIM model developed in former HL7 v3 based solution, since the experiment described on page 116 is parialy based on solution from Section 7.5. The third step involved a decision as to which underlying reference model the archetypes would use. In this solution, the archetypes use the openEHR Information Model, since they origin from the CKM repository managed by the openEHR Foundation.

Which form of communication was better? HL7 has a much better documentation, thus a straightforward development process. OpenEHR extract, templates and the Template Definition Language (TDL) [83] are not documented well yet, but they are much simpler than the HL7 v3 modelling methodology and suitable for smaller developer

LIM class	Archetype ID
Subjective Complaints Description (Observation-cls)	openEHR-EHR-CLUSTER.issue.v1
Patient Height Measurement (Observation-cls)	openEHR-EHR-OBSERVATION.height.v1
Body temperature measurement (Observation-cls)	openEHR-EHR-OBSERVATION. body_temperature.v1
Heart rate measurement (Observation-cls)	openEHR-EHR-OBSERVATION. heart_rate-pulse.v1
Breath frequency measurement (Observation-cls)	openEHR-EHR-OBSERVATION. respiration.v1
Waist circumference measurement (Observation-cls)	openEHR-EHR-OBSERVATION.waist_hip.v1
Laboratory examination (Act-cls)	openEHR-EHR-OBSERVATION.lab_test.v1
Smoking state determination (Observation-cls)	openEHR-EHR-OBSERVATION. substance_use-tobacco.v1

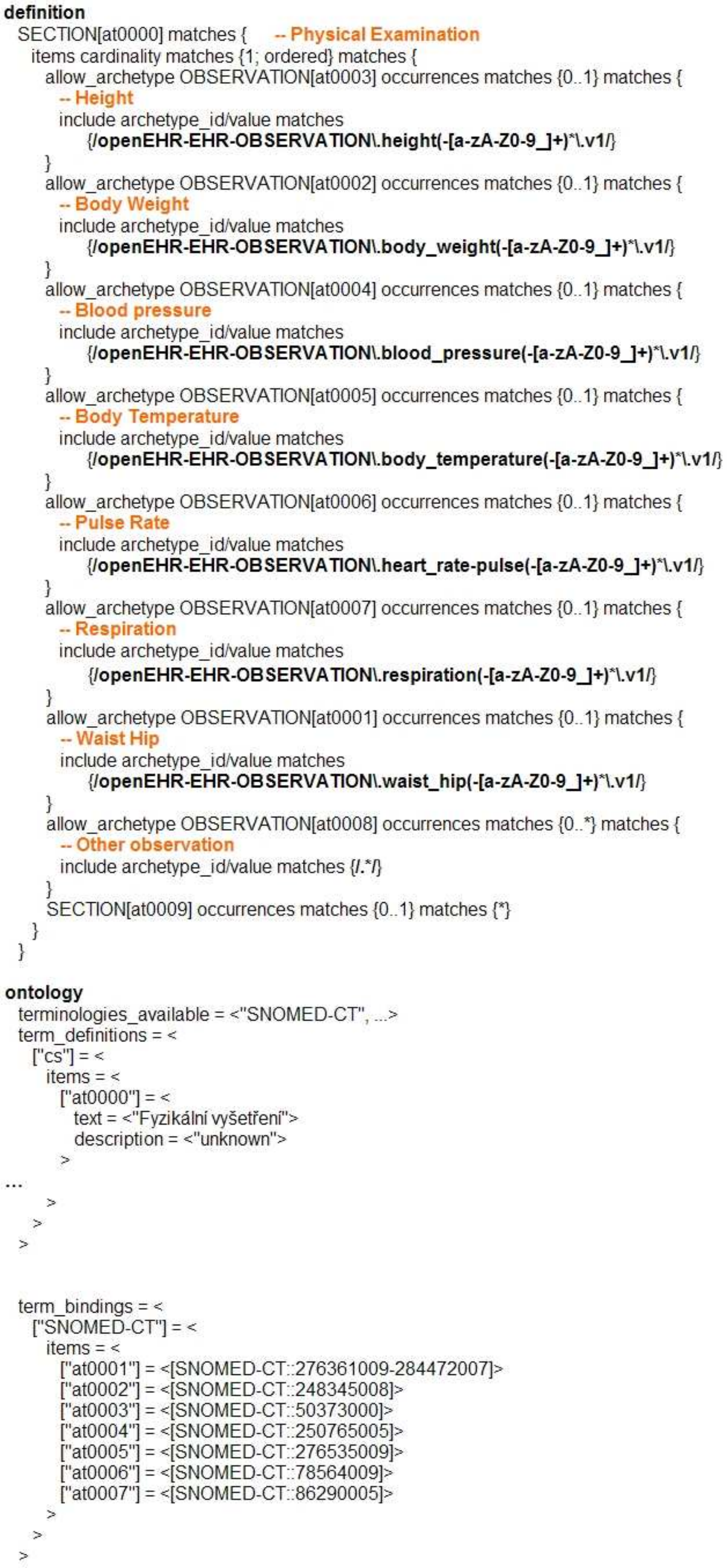
Table 7.4: Some archetypes matching the MDMC concepts modeled as LIM models based on HL7 v3 RIM.

teams.

Figure 7.15 shows an `openEHR-EHR-SECTION.physical_examination.v1` archetype, which was created by author using the Ocean Archetype Editor tool. This archetype was developed to be a root archetype of composed openEHR template (OET). The archetype is derived from the SECTION class from the openEHR Reference Model. According to the Archetype Building Guide: a SECTION is an organizing class, contained within a COMPOSITION. Archetypes of Sections standardize the organisation of information within a Composition. Examples of Sections are: Physical examination - organized by the System, History - organized by the present complaint, social history, review of systems etc.

The slots of the `openEHR-EHR-SECTION.physical_examination.v1` archetype contain following archetypes found in the CKM repository:

- openEHR-EHR-OBSERVATION.height.v1,
- openEHR-EHR-OBSERVATION.body_weight.v1,
- openEHR-EHR-OBSERVATION.blood_pressure.v1,



```
definition
  SECTION[at0000] matches {     -- Physical Examination
    items cardinality matches {1; ordered} matches {
      allow_archetype OBSERVATION[at0003] occurrences matches {0..1} matches {
        -- Height
        include archetype_id/value matches
            {/openEHR-EHR-OBSERVATION\.height(-[a-zA-Z0-9_]+)*\.v1/}
      }
      allow_archetype OBSERVATION[at0002] occurrences matches {0..1} matches {
        -- Body Weight
        include archetype_id/value matches
            {/openEHR-EHR-OBSERVATION\.body_weight(-[a-zA-Z0-9_]+)*\.v1/}
      }
      allow_archetype OBSERVATION[at0004] occurrences matches {0..1} matches {
        -- Blood pressure
        include archetype_id/value matches
            {/openEHR-EHR-OBSERVATION\.blood_pressure(-[a-zA-Z0-9_]+)*\.v1/}
      }
      allow_archetype OBSERVATION[at0005] occurrences matches {0..1} matches {
        -- Body Temperature
        include archetype_id/value matches
            {/openEHR-EHR-OBSERVATION\.body_temperature(-[a-zA-Z0-9_]+)*\.v1/}
      }
      allow_archetype OBSERVATION[at0006] occurrences matches {0..1} matches {
        -- Pulse Rate
        include archetype_id/value matches
            {/openEHR-EHR-OBSERVATION\.heart_rate-pulse(-[a-zA-Z0-9_]+)*\.v1/}
      }
      allow_archetype OBSERVATION[at0007] occurrences matches {0..1} matches {
        -- Respiration
        include archetype_id/value matches
            {/openEHR-EHR-OBSERVATION\.respiration(-[a-zA-Z0-9_]+)*\.v1/}
      }
      allow_archetype OBSERVATION[at0001] occurrences matches {0..1} matches {
        -- Waist Hip
        include archetype_id/value matches
            {/openEHR-EHR-OBSERVATION\.waist_hip(-[a-zA-Z0-9_]+)*\.v1/}
      }
      allow_archetype OBSERVATION[at0008] occurrences matches {0..*} matches {
        -- Other observation
        include archetype_id/value matches {/.*/}
      }
      SECTION[at0009] occurrences matches {0..1} matches {*}
    }
  }

ontology
  terminologies_available = <"SNOMED-CT", ...>
  term_definitions = <
    ["cs"] = <
      items = <
        ["at0000"] = <
          text = <"Fyzikální vyšetření">
          description = <"unknown">
        >
...
      >
    >
  >


  term_bindings = <
    ["SNOMED-CT"] = <
      items = <
        ["at0001"] = <[SNOMED-CT::276361009-284472007]>
        ["at0002"] = <[SNOMED-CT::248345008]>
        ["at0003"] = <[SNOMED-CT::50373000]>
        ["at0004"] = <[SNOMED-CT::250765005]>
        ["at0005"] = <[SNOMED-CT::276535009]>
        ["at0006"] = <[SNOMED-CT::78564009]>
        ["at0007"] = <[SNOMED-CT::86290005]>
      >
    >
  >
```

Figure 7.15: Definition and Ontology sections of the `openEHR-EHR-SECTION.physical_examination.v1` archetype, which is a root archetype in the Physical Examination OET.

- openEHR-EHR-OBSERVATION.body_temperature.v1,
- openEHR-EHR-OBSERVATION.heart_rate-pulse.v1,
- openEHR-EHR-OBSERVATION.respiration.v1,
- openEHR-EHR-OBSERVATION.waist_hip.v1.

Finally, concepts covered by archetypes slots were encoded in the ontology section by codes from SNOMED CT (the lower part of Figure 7.15).

Once having the archetype describing exchanged data ready, the actual data transportation can be established. The openEHR approach distinguishes various types of Extracts like EHR_EXTRACT, GENERIC_EXTRACT and synchronisation_EXTRACT. The document [80] describes the architecture of the openEHR EHR Extract Information Model. This model is equivalent in scope to CEN EN 13606 (see p. 35). For this experiment the EHR_EXTRACT was used. The EHR Extract is defined in terms of a general model of Folders / Compositions / Demographics / other items, following the top level structure of the openEHR EHR. The contents of each of these parts of the Extract are in the form of a serialised variant of the VERSIONED_OBJECT class.

The structure of the EHR Extract is depicted in Figure 7.16 and the inclusion of the particular SECTION class from the openEHR-EHR-SECTION.physical_examination.v1 archetype into an EHR Extract instance will be as follows. The class rm.extract.common.EXTRACT contains an attribute chapters:List<EXTRACT_CHAPTER>. The rm.extract.common.EXTRACT_CHAPTER contains content:EXTRACT_ENTITY_CONTENT to record the content of the given chapter. The rm.extract.common.EXTRACT_ENTITY_CONTENT is a container of extracted and serialised content, intended to be subtyped into e.g. EHR_EXTRACT_CONTENT. The class rm.extract.ehr_extract.EHR_EXTRACT_CONTENT contains among others a list of compositions in the form of X_VERSIONED_OBJECT<COMPOSITION>. Finally, the SECTION class defined in openEHR-EHR-SECTION.physical_examination.v1 archetype will be a value of the content:CONTENT_ITEM attribute. Such an assignment is valid because the class rm.composition.content.navigation.SECTION is derived from the rm.ehr.composition.content.CONTENT_ITEM class.

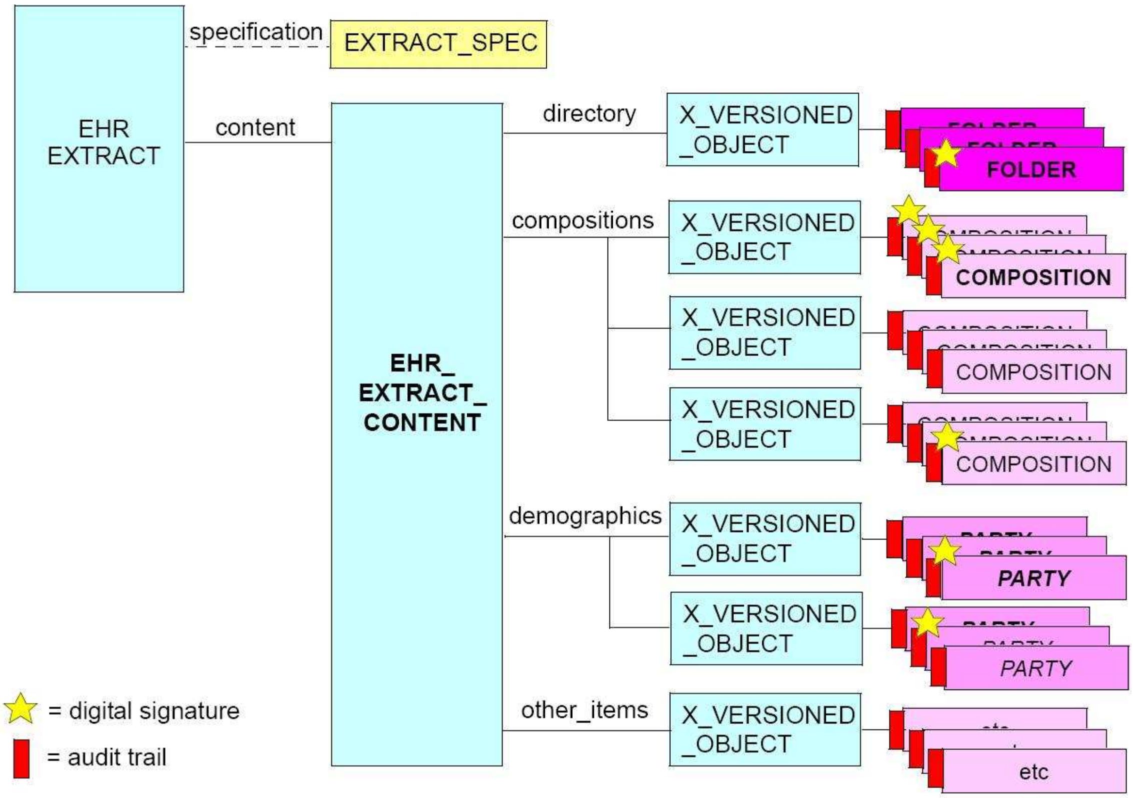


Figure 7.16: openEHR EHR Extract structure.

The components used in the experiment as well as the data flow are depicted in Figure 7.17. The flow of data in the openEHR based solution is as follows. The former LIM Filler module attached to ADAMEKj EHR creates the LIM message containing data about a physical examination. The LIM message is then transformed by XSL templates into the EHR Extract conforming to the `openEHR-EHR-SECTION.physical_examination.v1` archetype (Figure 7.15). Both the LIM message and the EHR Extract instance are encoded in XML. The XML form of the EHR Extract is realised as a dADL language serialised into XML as described in [42]. The EHR Extract instance is sent via SOAP to the former HL7 Broker, which was reconfigured for the purpose of this experiment. The HL7 broker then sends the EHR Extract instance to the recieving system. In this experiment the HL7 broker does not transform the openEHR Extracts in any way. It serves as an inbox/outbox for the communicating EHR-S.

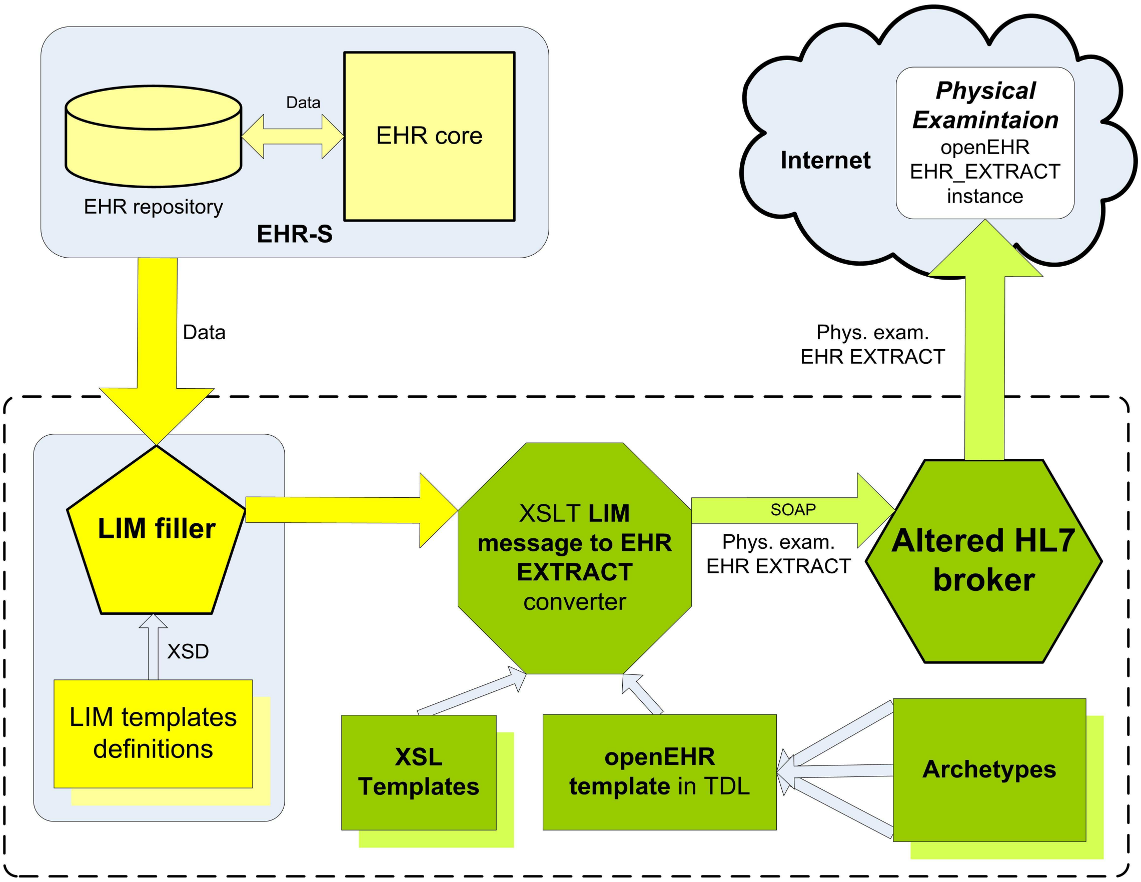


Figure 7.17: New communication schema based on openEHR archetypes, templates and extracts.

7.7 Archetype Modelling Methodology

To propose a modelling guideline or some sort of methodology for the archetypes' creation, the content of EHR they describe must be clarified. According to the "EHR information model" [43], these are commonly used types of clinical information: *Basic information* (e.g. date of birth, sex, height, weight, pregnancy), *Problem list*, *Medications list*, *Therapeutic precautions* (allergies and alerts), *Patient preferences*, *Patient consents*, *Family history*, *Social history/situation*, *Lifestyle*, *Vaccination record* and *Care plan*.

This is a high level structure of EHR which can be found in a more or less complete form in many contemporary systems. The ADAMEKj also corresponds with this structure. This fact suggests an idea that if the modelling process of EHR adheres to the commonly agreed guidelines or methodologies, which are independent on the implementation aspects (i.e. used approach - object oriented programming, archetypes, relational database etc.), much better interoperable EHRs can be implemented. This implies the creation of ontologies of various domains in medicine.

Within the frame of the research in the EuroMISE Centre, a MUDR KB tree based on concepts of the MDMC was created. Another KB tree was created for the domain of dentistry (described in Section 6.3 on p. 60). MUDR KB trees can be considered as some kind of ontologies, because they comprise vertices interconnected with various kinds of edges. The tree structure is formed only by the edge of the *superior* type. From the modelling point of view, MUDR KB trees do not depend on any kind of implementation technology. After creation of the KB tree in the MUDR Knowledge Base Editor tool, it can be imported to MUDR or it can serve as a basis for further modelling process. Hence, the process of enabling the creation of EHRs with harmonized clinical content formulated on page 112 can be extended by the *ontology creation* step at the beginning of the process.

Although the ontology creation process can end up with various ontologies describing more or less the same domain, they can be aligned and merged [86] thank to the research conducted in the field of knowledge modelling.

However, openEHR archetypes do not have merging and aligning techniques developed. Nevertheless, in the frame of this thesis, an ADL archetype comparison tool was developed (presented on p. 94), which could help to compare two archetypes developed by different authors, but describing the same concept. Such a tool could be fully utilised in a standards development organisation (SDO) which would be in charge of managing a central archetype repository.

The archetype creating and modelling methodology cannot be defined by any research team. This should be proposed by a SDO thus enabling creation of an extensible and open centralised repository of clinical content modeled in the form of archetypes. The openEHR Foundation is approaching a solution of this task via their approval process in the Clinical Knowledge Manager containing commonly accepted clinical concepts.

Chapter 8

Discussion

It turned out that at the beginning of the clinical modelling proces, it is good to start with a statistical model, which consists of a list of collected attributes, their data types and containts (e.g. the range of values). The statistical model is then transformed by a domain expert into a formal model describing the clinical field. The final stage of the clinical model creation should comprise mapping clinical concepts to established coding systems in order to fix the semantic meaning of modelled concepts. This step should result in minimising the ambiguities in the model, which is robust enough to be used as a modelling basis for the implementation of communication standards.

However, the CEN EN 13606 standard is gaining on popularity and is in the center of interest in various research projects [87]; in this thesis the openEHR approach was used mainly for its continual improvement and development process, better documentation and broader acceptance by experts. Since the openEHR approach is tightly related to the EN 13606, the conversion of our solution remains possible.

Introduction of enterprise programming technologies, such as Hibernate framework, into the development process of medical applications (concerning ADAMEKj EHR), resulted in shifting of the modelling procedures from the database layer to an object-oriented programming language. Although actual data are still stored in the relational database, the object model can be mapped to relational tables by the tools of Hibernate framework called the Hibernate Mapping files (HBM) (see Fig. 8.1).

This approach showed that established modern tools for enterprise programming are

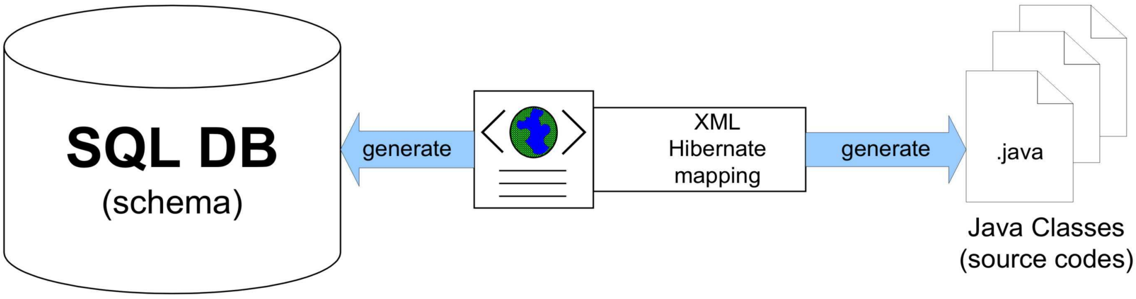


Figure 8.1: The process of generating database schema and Java source codes from Hibernate mapping file (HBM).

well applicable in the area of healthcare as well. Today's standard enterprise technologies like the Spring Framework based on the Java2 Enterprise Edition are going to be the most effective development platforms. They allow us to develop EHR systems of various sizes, i.e. from single-user systems to enterprise-level ones, and of various degrees of universality, i.e. from simple applications tailored to one's needs to universal, extensible and sophisticated systems. ADAMEKj EHR served as a testing repository of clinical data employed in the development process of the HL7 v3 standard-based semantic interoperability platform.

EHR in dentistry typically stores information about the status of denture and surrounding tissues and as well as treatment procedures. Since the set of concepts describing dental defects and corresponding recovery procedures is limited and well-defined, it is the right candidate to be stored in a structured form. The relatively high degree of structural exuberancy can be observed in many paper-based forms which contain so-called dental cross, which was also included in the DentCross component.

The "natural" acceptance of data in a structured form makes the dentistry the most suitable field for the structured EHR application. The MUDRLite EHR with interactive Dental Cross component offers a transparent record of the whole dentition and individual accomplished examinations in a concentrated form. The dental information recorded in a common graphical structure accelerates the dentist's decision-making and provides a more complex view of gathered information.

For the purpose of this work the part of Disaster Victim Identification (DVI) [88] form recording dental status of the victim was studied in more detail. It is tightly

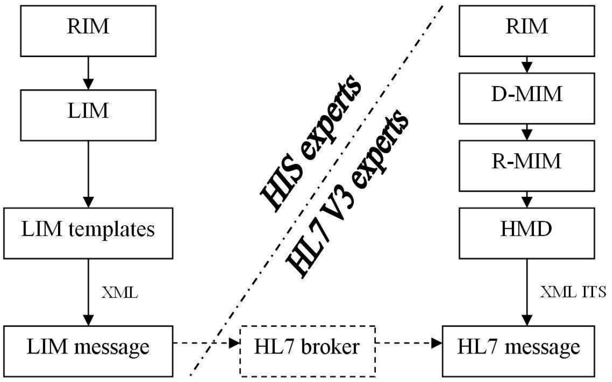


Figure 8.2: The messaging development process, recommended by HL7 v3 on the right and our solution on the left side.

connected with dental EHR, which was developed as part of this PhD thesis (see p. 75), and can serve as a repository of ante-mortem (before death) records. The comparison of dental cross structures (ante- and post-mortem) is successfully utilized in disaster victims identification [69].

The development process of message interchange, recommended by HL7 v3, was altered by splitting the implementation effort between HIS developers and HL7 standard implementers (see Figure 8.2). This new approach might help the developers to overcome the initial frustration which is caused by the overwhelming size of the HL7 standard (RIM, amount of artifacts, domains etc.). The development of individual LIMs, closely related to the internal information structure of the particular HIS, with the simple communication interface between HIS and HL7 broker based on LIM messages, SOAP and Web Services, seems to be more manageable for smaller developer teams than a strict adherence to HL7 v3 methodology. This modelling methodology is utilised only by developers (experts on HL7 v3) of the core component of the solution – the HL7 v3 Broker. Thank to the fact that LIMs are based on HL7 v3 RIM, they are not that difficult to transform into HL7 v3 messages as the original models would be.

During the development and implementation of the platform for semantic interoperability, it was necessary to use the simulated patient data, as the use of real patient data is not allowed for such purpose due to legislative reasons. Results of performed tests were not affected by the fact that the data were simulated and are valid for real patient data as well.

Using LIM models and LIM fillers resulted in considerable universality of the solution that consequently does not depend on the communication standard used (although the LIM is based on HL7 v3 RIM). This independency is supported by the fact that contemporary modern communication standards have some important characteristics in common: basic reference model, user defined models derived from that reference model using strict methodology, and finally, a kind of templates helping in creating a new message or document. A comparison of contemporary communication standards can be found in [89].

The message interchange based on EHR Extracts from openEHR is a very interesting and relatively unexplored field, as the openEHR approach is primarily oriented on description of the development of future-proof EHR systems. This kind of messaging is based on a simple idea – instead of rendering the definitions contained in templates as screen forms, these definitions could be used to build EHR_EXTRACT instances, which are consequently exchanged among EHR-Ss. Such a concept is close to the document interchange via HL7 v3 CDA, which, however, has one major drawback – a lower degree of data structuring. Although the EHR_EXTRACT_REQUESTs are already developed a formal language for querying EHR data is still not part of the openEHR specification. This gap should be filled in by a query language developed by Ocean Informatics, named EHR Query Language (EQL) [90].

Despite many positives the ADL language has some drawbacks that are summarised in [91]. The main problem is that because of its generality, the language does not provide any component that guarantees the consistency of clinical information. It can only offer the consistency on the archetype level, i.e. the conformance of ADL/AOM (Archetype Object Model) [77] principles. Therefore, in order to process the ADL content, two elements are necessary: an ADL parser for capturing AOM objects and the parser of the particular reference model to guarantee the clinical correctness of the ADL content. Atuhors in [91] suggest using OWL language for supporting semantic processing and activities.

During the research process in the area of clinical content of EHRs, the following types of content harmonisation were identified:

- Harmonisation of the content on database schema level – it involves the synchronisation of various versions of EHR from the same vendor; for this purpose the Schemagic tool was developed. Relational database schemas and MUDR knowledge trees are supported.

- Harmonisation of the development process on the knowledge level – the target of this type of content harmonisation is to converge to the state when all vendors would have access to common source of knowledge; for this purpose the knowledge repository (e.g. openEHR archetype repository) would be used. A comparison of knowledge concepts can be promoted by the extension of Schemagic supporting archetypes.

- Data interchange among EHR systems produced by various vendors – for this purpose the communication standard would be implemented, acting as a transformation tool.

The communication via HL7 v3 messaging standard or openEHR EHR Extracts is in real life the result of a huge effort of many people – domain experts, developers, medical stuff etc. Therefore, having EHRs with HCC available would reduce the complexity of communication frameworks and various translator and mapping modules. The data interchange would be much more straightforward and that is worth studying rigorously.

Our research in agreement with scientific literature suggests the following solution for harmonisation of clinical content:

- the archetype development process should be formalised in a form of guidelines,

- a centralised authority should be created with the task to manage a centralised repository of archetypes,

- commonly acceptable indentification of archetypes should enable more precise results during the searching for clinical concepts.

8.1 Recognised Needs of Clinical Content harmonisation

The IT-014 Health Informatics Committee published a paper calling for a need for clinical content harmonisation in [92] as an "Action: (IT-014-09)". There is a need for clinical content harmonisation, even if this only starts with a single repository/resource/library/registry/index for existing clinical models of all types.

In SemanticHEALTH Report [2] future expectations are summarised: "... regardless of the type of vision one may develop, semantic interoperability is not a phenomenon to be expected to happen over night. Rather, it will emerge gradually and in the most optimistic of assumptions it may remain an incomplete phenomenon. The SemanticHEALTH vision is characterised by a large number of changes at both the technical and the use case level. Note however, that even in this vision, no full semantic interoperability or a complete harmonisation of either EHR models or terminologies can be expected."

8.2 Related Work

In this section some projects with similar aims are discussed. An elaborate work regarding semantic interoperability can be found in [3], where the development framework (not the implementation itself) for semantically interoperable health information systems is described. However, our research is oriented mainly on the realisation of the interoperability platform rather than pure theoretical issues.

The procedure of semantic interoperability achievement among EHR systems storing clinical information in various proprietary formats was studied in project ARTEMIS [93]. The resulting solution contained an idea of wrapping and exposing the existing healthcare applications as Web Services [94].

Several independent teams of scientists study the possibilities of archetypes implementation into local (proprietary) EHR systems [54], [53], [87], [95]. These efforts are closely related to the experiment (see Section 7.6) performed in the frame of this thesis.

Encoding of clinical content by means of SNOMED CT terminology is described in

[96]. The MDMC concepts were mapped not only to SNOMED CT, but also LOINC and ICD-10 coding systems were utilised (Section 7.1.1). HL7 v3 based queries were studied in Section 7.5 and more detailed research on this topic can be found in [97].

Similarly to our research results, the National eHealth Transition Authority (NEHTA) recommends [98] that Australia adopt, develop and, over time, migrate to a document/services-centric implementation of HL7 v3, based on HL7 CDA and joint HL7/OMG Health Services Specification Project. In parallel with pursuing these targets, NEHTA is continuing to track the developments in openEHR/EN 13606 archetypes and associated tools for capturing and representation of clinical content.

Chapter 9

Conclusion

The research accomplished in the frame of this PhD thesis analysed two approaches to clinical content harmonisation. The real benefits of the proposed methods can be learned only when HL7 v3 standard and/or openEHR archetypes will be widely used. The usage of the HL7 v3 is hampered by its great complexity discouraging users who have experienced a different style of work with the former HL7 v2. This fact is referenced also in some foreign studies, such as [98].

The main step for a wider use of HL7 v3 in the Czech Republic should be the implementation of functionality, which is currently provided by the DASTA national standard, the inclusion of NCLP on the list of HL7-supported code systems, or better the mapping of the NCLP to an established international nomenclature like SNOMED CT. The next fundamental step would be obtaining the translation of the international nomenclature into the Czech language.

Generally, for a wider acknowledgement of archetypes the acceptance of the EN 13606 standard may be benefitial. Moreover, extension of this standard or specifying a new one which would define archetype based EHR systems, would help broader acceptance of 2-level modelling and knowledge separating approaches.

The UMLS Knowledge Source can be exploited in clinical content mapping in the future. Further support and contribution to harmonisation of clinical ontologies are also one of the future goals. The primary objective for the future, however, should be reaching an agreement about clinical concepts used in EHR-S with well-defined semantics by means of international terminologies. Clinical content coordination demands should be addressed to the EuroRec institute via national proREC centres in order to push the idea of semantic interoperability further.

Appendix A

Local Information Model of ADAMEKj EHR

In this appendix the whole Information Model of the ADAMEKj EHR in form of Local Information Models (LIMs) is presented. All its classes are derived from the HL7 v3 RIM and following figures conain UML Class Diagrams, which adhere to common ”class colors” used in HL7 RIM.

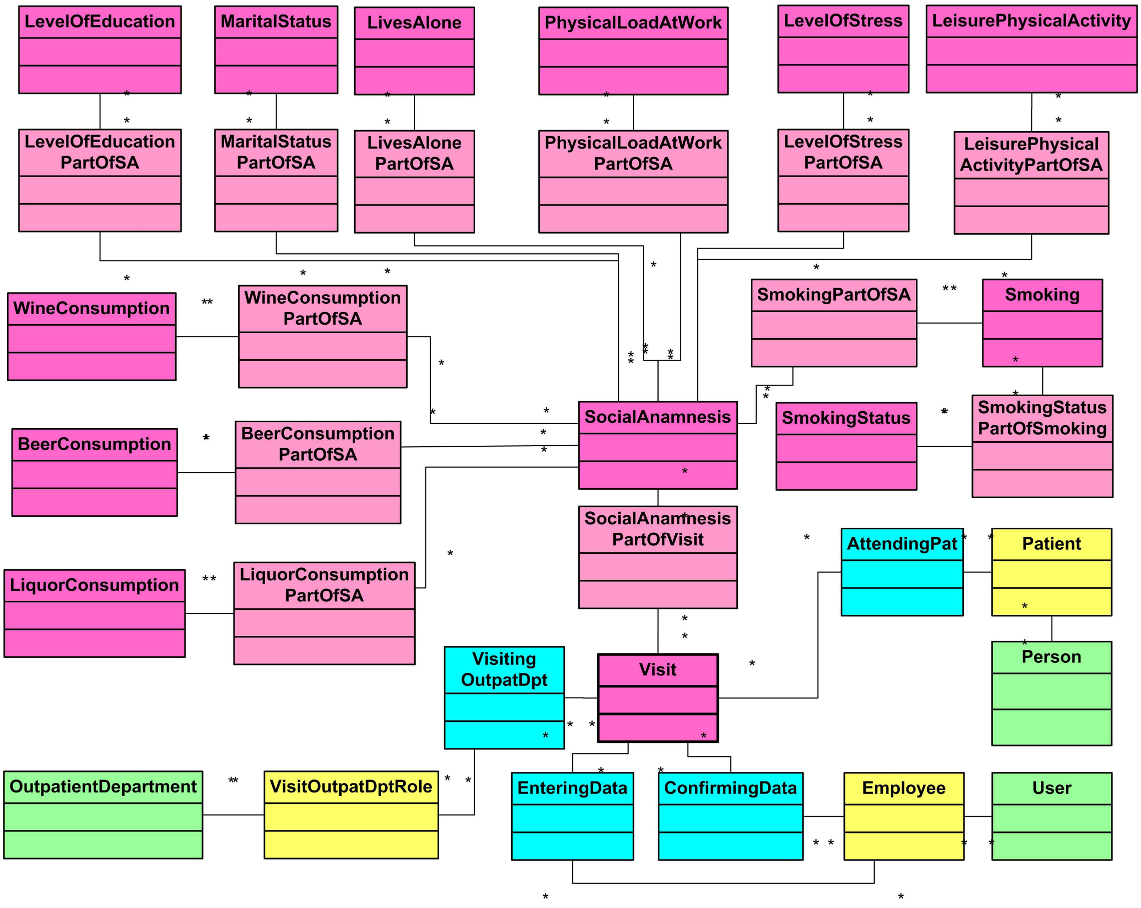


Figure A.1: Social anamnesis of a patient modeled as LIM.

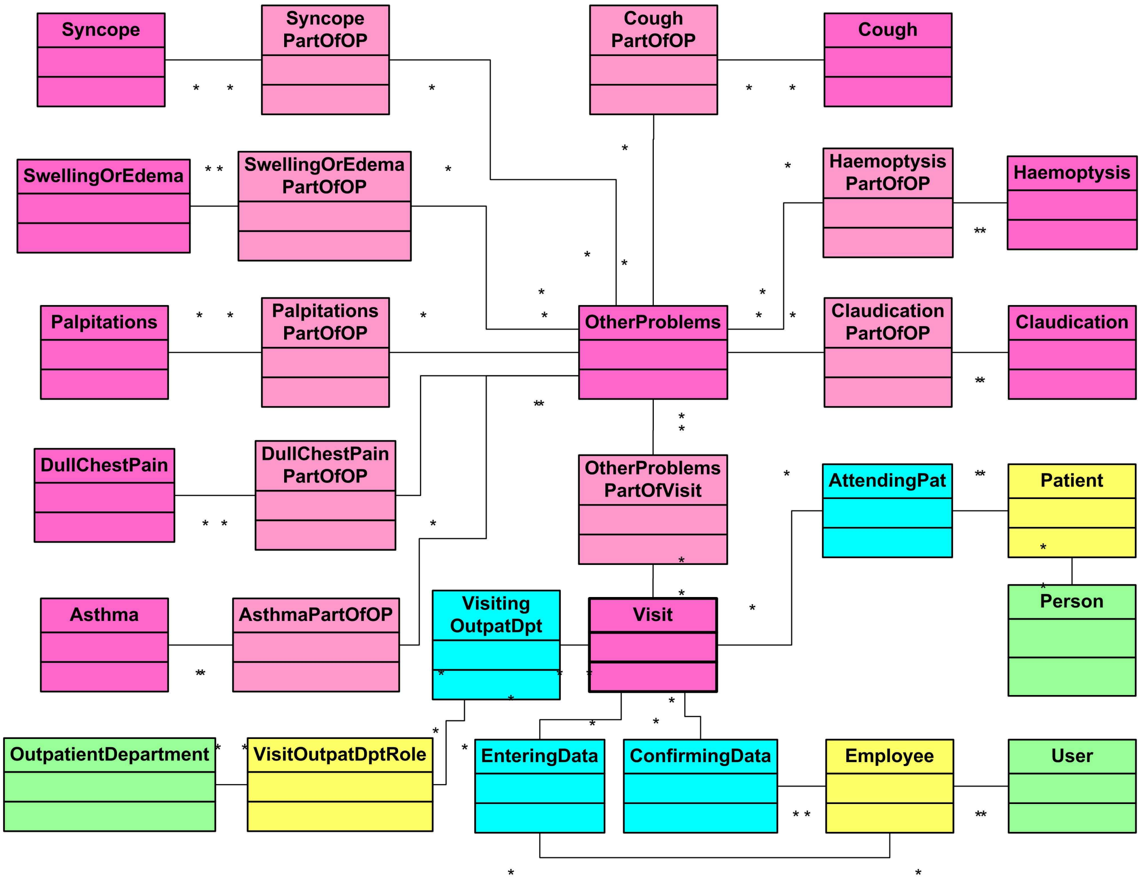


Figure A.2: Current patient's problems of eventual cardiovascular origin modeled as LIM.

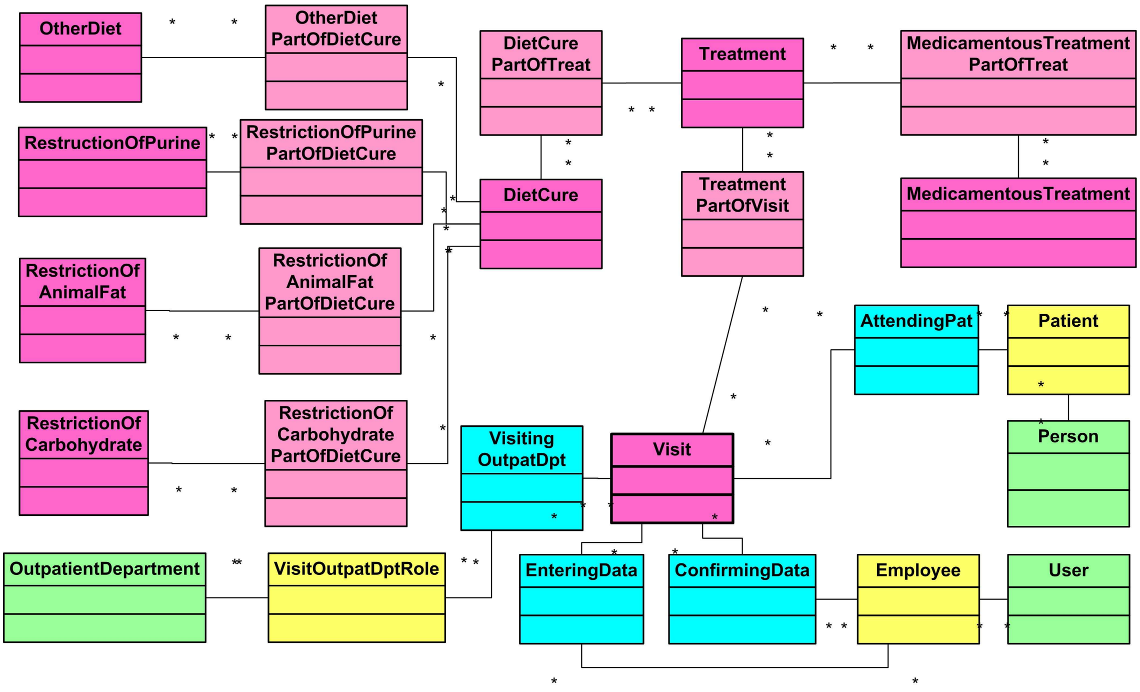


Figure A.3: LIM for patient’s treatment procedures.

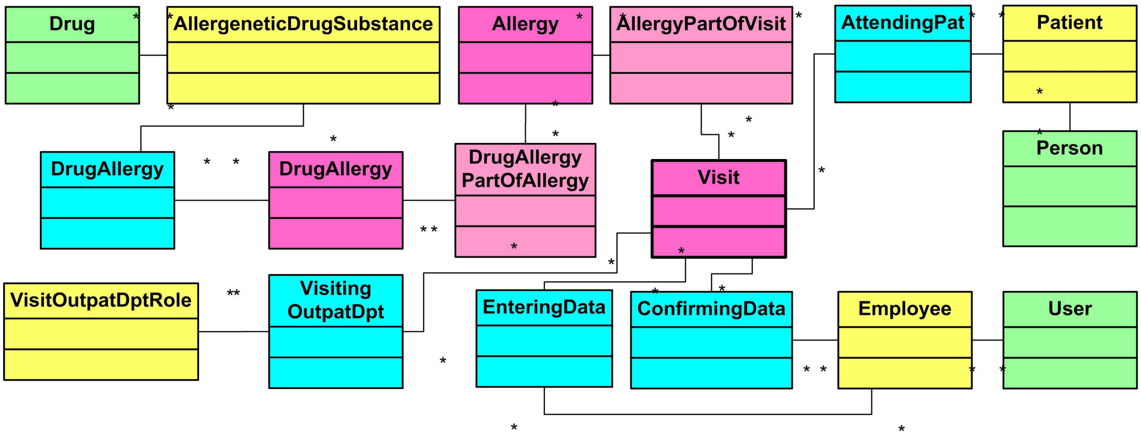


Figure A.4: LIM for Patient’s Allergies.

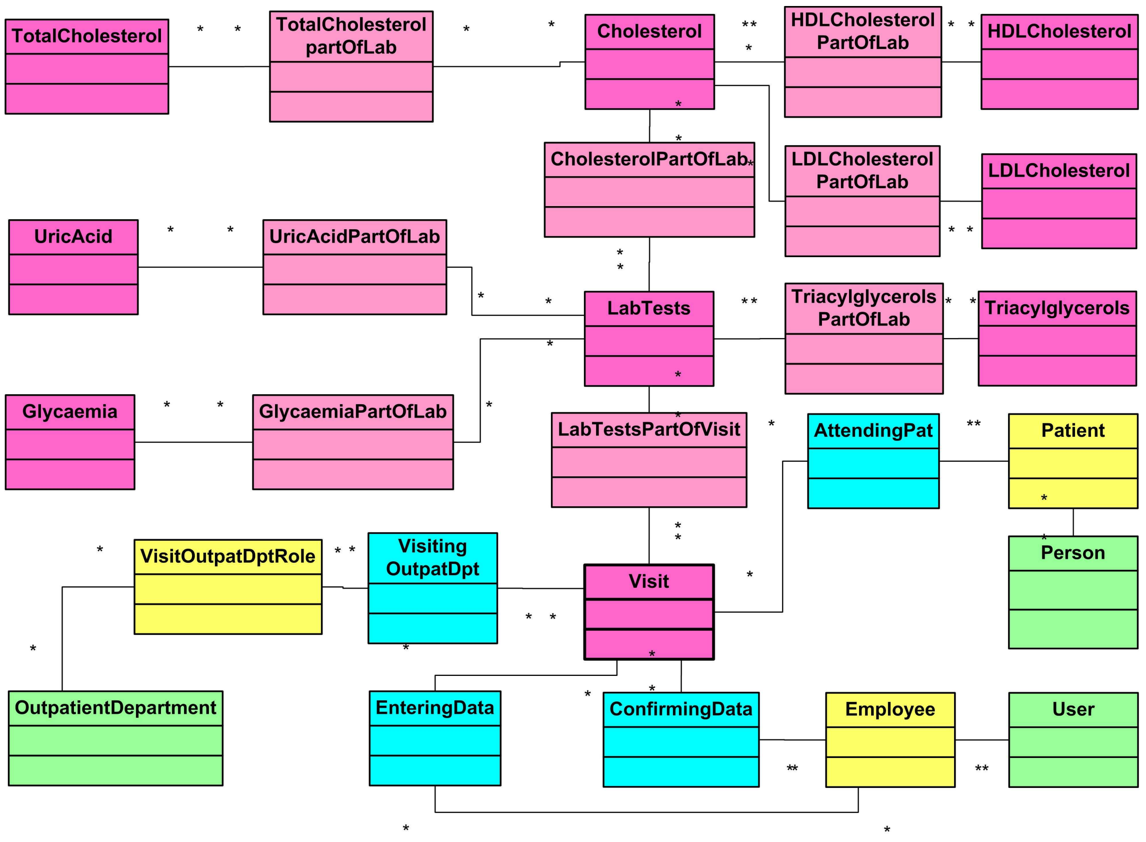


Figure A.5: LIM for Laboratory Results aimed on cardiologic risk factors.

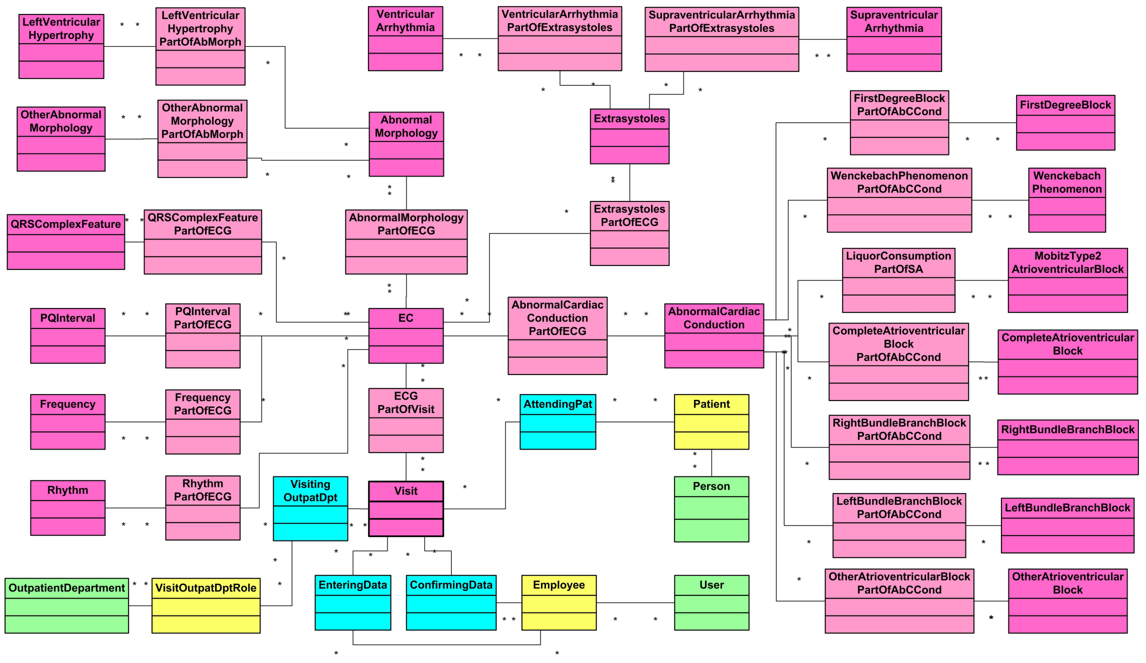

LeftVentricular Hypertrophy
LeftVentricular Hypertrophy PartOfAbMorph
Ventricular Arrhythmia
VentricularArrhythmia PartOfExtrasystoles
SupraventricularArrhythmia PartOfExtrasystoles
Supraventricular Arrhythmia
OtherAbnormal Morphology
OtherAbnormal Morphology PartOfAbMorph
Abnormal Morphology
Extrasystoles
FirstDegreeBlock PartOfAbCCond
FirstDegreeBlock
WenckebachPhenomenon PartOfAbCCond
Wenckebach Phenomenon
QRSComplexFeature
QRSComplexFeature PartOfECG
AbnormalMorphology PartOfECG
Extrasystoles PartOfECG
LiquorConsumption PartOfSA
MobitzType2 AtrioventricularBlock
PQInterval
PQInterval PartOfECG
EC
AbnormalCardiac Conduction PartOfECG
AbnormalCardiac Conduction
CompleteAtrioventricular Block PartOfAbCCond
CompleteAtrioventricular Block
Frequency
Frequency PartOfECG
ECG PartOfVisit
AttendingPat
Patient
RightBundleBranchBlock PartOfAbCCond
RightBundleBranchBlock
Rhythm
Rhythm PartOfECG
Visiting OutpatDpt
Visit
Person
LeftBundleBranchBlock PartOfAbCCond
LeftBundleBranchBlock
OutpatientDepartment
VisitOutpatDptRole
EnteringData
ConfirmingData
Employee
User
OtherAtrioventricularBlock PartOfAbCCond
OtherAtrioventricular Block

Figure A.6: LIM for patient’s ECG examination.

Appendix B

Mappings of MDMC to SNOMED CT, LOINC and UMLS

The complete table representing the mapping of concepts from MDMC to various coding systems (SNOMED CT, LOINC, ICD-10) could not be presented here, mainly because of its size. The resulting table consists of more than 500 rows and therefore it is available in MS Excel format (file `mapping.mdmc.to.codes.xls`) in the results repository located at `http://neo.euromise.cz/nagy/pgs`.

Appendix C

OpenEHR Archetypes Matching Concepts from MDMC

Description of MDMC concept	**Archetype name**
Allergies	
drug allergy	openEHR-EHR-EVALUATION.adverse.v1
other allergy	openEHR-EHR-EVALUATION.adverse.v1
Physical examination	
height	openEHR-EHR-OBSERVATION.height.v1
weight	openEHR-EHR-OBSERVATION.body_weight.v1
waist	openEHR-EHR-OBSERVATION.waist_hip.v1
hips size	openEHR-EHR-OBSERVATION.waist_hip.v1
body temperature	openEHR-EHR-OBSERVATION.body_temperature.v1
blood pressure	openEHR-EHR-OBSERVATION.blood_pressure.v1
pulse rate	openEHR-EHR-OBSERVATION.heart_rate-pulse.v1
respiratory rate	openEHR-EHR-OBSERVATION.respiration.v1
Laboratory results	
Uric acid measurement	openEHR-EHR-OBSERVATION.lab_test-urine_protein.v1
triacyglycerols	openEHR-EHR-OBSERVATION.lab_test-lipids.v1
LDL Cholesterol	openEHR-EHR-OBSERVATION.lab_test-lipids.v1
HDL Cholesterol	openEHR-EHR-OBSERVATION.lab_test-lipids.v1
Total Cholesterol	openEHR-EHR-OBSERVATION.lab_test-lipids.v1
glycaemia	openEHR-EHR-OBSERVATION.lab_test-blood_glucose.v1
ECG	
ECG meassurement	openEHR-EHR-OBSERVATION.ecg.v1

Table C.1: OpenEHR Archetypes Matching Concepts from MDMC (part 1)

Treatment	
medication	openEHR-EHR-ACTION.medication.v1
prescribed drug	openEHR-EHR-ITEM_TREE.medication-vaccine.v1
Current problems	
asthma	openEHR-EHR-OBSERVATION.respiration.v1
chest pain	openEHR-EHR-CLUSTER.exam-chest.v1
edema	openEHR-EHR-CLUSTER.oedema.v1
Personal anamnesis	
actual problems	openEHR-EHR-SECTION.problem_list.v1
diabetes mellitus	openEHR-EHR-EVALUATION.problem.v1
hipertension	
hyperlipoproteinemia	
ischemic heart disease	
cerebrovascular accident	
ischemic peripheral artery disease	
aneurysm of aorta	
other relevant diseases of patient	
date of death	openEHR-DEMOGRAPHIC-CLUSTER.person_death_data_iso.v1
cause of death	
Social anamnesis	
marital status	openEHR-DEMOGRAPHIC-CLUSTER.person_additional_data_br.v1
physical load in job	openEHR-EHR-CLUSTER.level_of_exertion.v1
physical activity out of job	openEHR-EHR-CLUSTER.level_of_exertion.v1
smoking	openEHR-EHR-OBSERVATION.substance_use-tobacco.v1
alcohol	openEHR-EHR-EVALUATION.substance_use_summary-alcohol.v1
beer intake	openEHR-EHR-OBSERVATION.substance_use-alcohol.v1
wine intake	
hard liquor intake	

Table C.2: OpenEHR Archetypes Matching Concepts from MDMC (part 2)

Appendix D

Database Model of the ADAMEKj Application

The data model for ADAMEKj application is entirely based on the MDMC and its E-R model is shown in Figure D.1. Since the size of available paper (A4) is far too small for this model to display properly, its electronic version (in the form of a JPEG image `adamekj.dm.jpg`) is located in the results repository located at `http://neo.euromise.cz/nagy/pgs`.

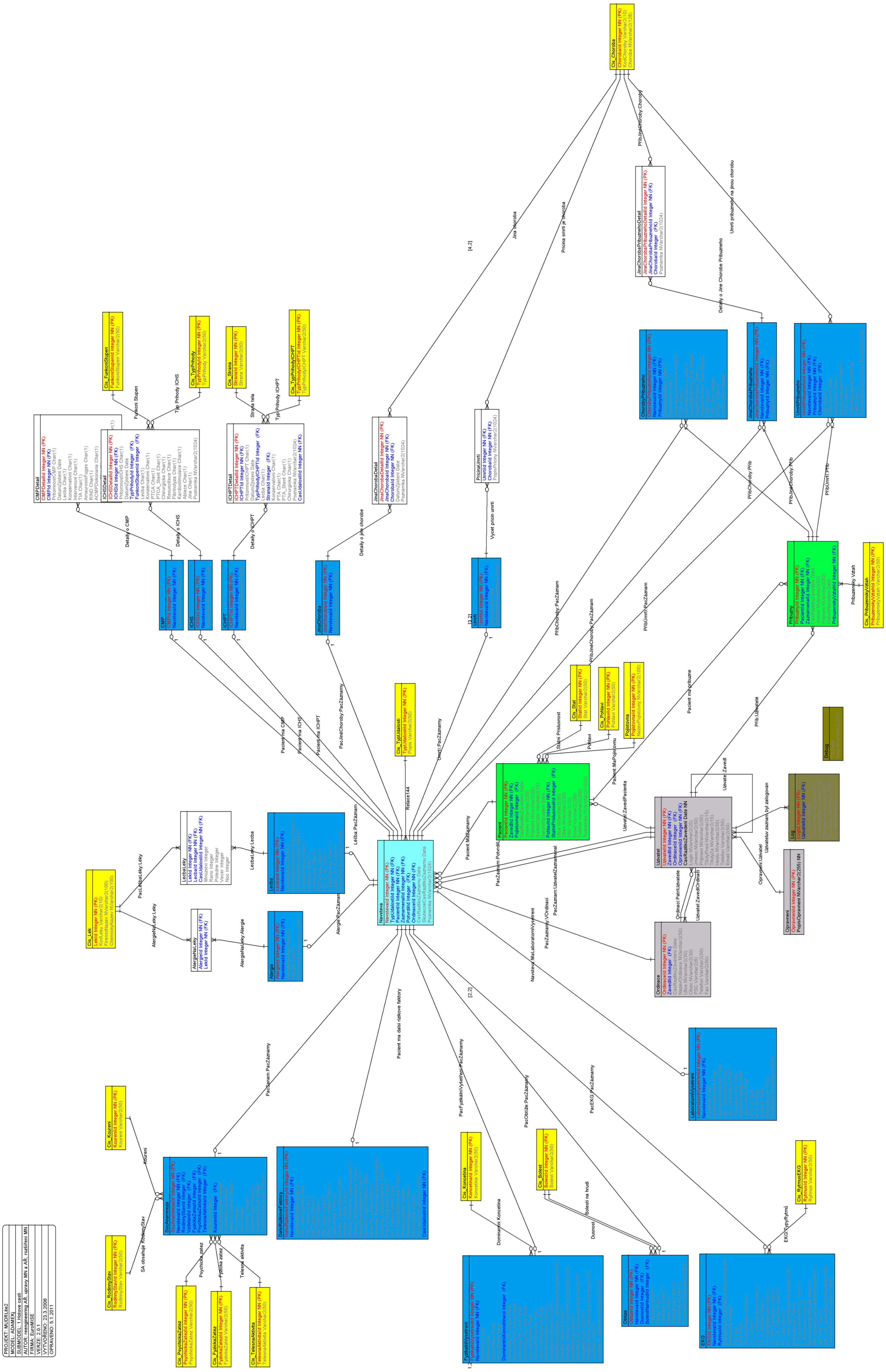


Figure D.1: The E-R model of MDMG implemented in ADAMEKj application.

Appendix E

Database Model of the DentCross Component

E-R model of the database layer in the MUDRLite EHR serving as a storage of data for the DentCross custom component is depicted in Figure E.1 and E.2. For better readibility the figure was split into two parts.

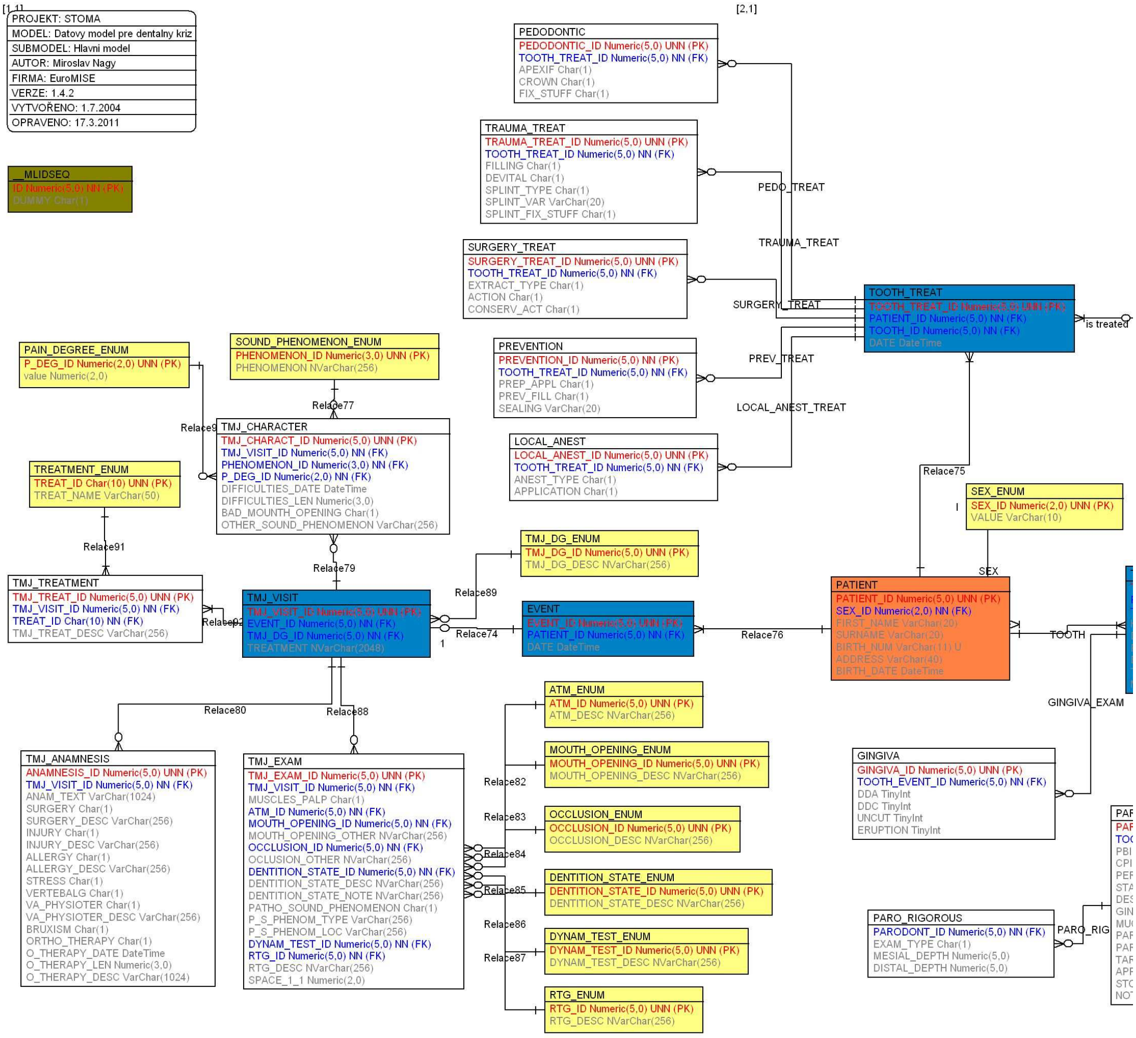

[1,1]
PROJEKT: STOMA
MODEL: Datovy model pre dentalny kriz
SUBMODEL: Hlavni model
AUTOR: Miroslav Nagy
FIRMA: EuroMISE
VERZE: 1.4.2
VYTVOŘENO: 1.7.2004
OPRAVENO: 17.3.2011
__MLIDSEQ
ID Numeric(5,0) NN (PK)
DUMMY Char(1)
[2,1]
PEDODONTIC
PEDODONTIC_ID Numeric(5,0) UNN (PK)
TOOTH_TREAT_ID Numeric(5,0) NN (FK)
APEXIF Char(1)
CROWN Char(1)
FIX_STUFF Char(1)
TRAUMA_TREAT
TRAUMA_TREAT_ID Numeric(5,0) UNN (PK)
TOOTH_TREAT_ID Numeric(5,0) NN (FK)
FILLING Char(1)
DEVITAL Char(1)
SPLINT_TYPE Char(1)
SPLINT_VAR VarChar(20)
SPLINT_FIX_STUFF Char(1)
PEDO_TREAT
TRAUMA_TREAT
SURGERY_TREAT
SURGERY_TREAT_ID Numeric(5,0) UNN (PK)
TOOTH_TREAT_ID Numeric(5,0) NN (FK)
EXTRACT_TYPE Char(1)
ACTION Char(1)
CONSERV_ACT Char(1)
SURGERY_TREAT
TOOTH_TREAT
TOOTH_TREAT_ID Numeric(5,0) UNN (PK)
PATIENT_ID Numeric(5,0) NN (FK)
TOOTH_ID Numeric(5,0) NN (FK)
DATE DateTime
is treated
PAIN_DEGREE_ENUM
P_DEG_ID Numeric(2,0) UNN (PK)
value Numeric(2,0)
SOUND_PHENOMENON_ENUM
PHENOMENON_ID Numeric(3,0) UNN (PK)
PHENOMENON NVarChar(256)
PREVENTION
PREVENTION_ID Numeric(5,0) NN (PK)
TOOTH_TREAT_ID Numeric(5,0) NN (FK)
PREP_APPL Char(1)
PREV_FILL Char(1)
SEALING VarChar(20)
PREV_TREAT
Relace77
Relace9
TMJ_CHARACTER
TMJ_CHARACT_ID Numeric(5,0) UNN (PK)
TMJ_VISIT_ID Numeric(5,0) NN (FK)
PHENOMENON_ID Numeric(3,0) NN (FK)
P_DEG_ID Numeric(2,0) NN (FK)
DIFFICULTIES_DATE DateTime
DIFFICULTIES_LEN Numeric(3,0)
BAD_MOUNTH_OPENING Char(1)
OTHER_SOUND_PHENOMENON VarChar(256)
LOCAL_ANEST_TREAT
LOCAL_ANEST
LOCAL_ANEST_ID Numeric(5,0) UNN (PK)
TOOTH_TREAT_ID Numeric(5,0) NN (FK)
ANEST_TYPE Char(1)
APPLICATION Char(1)
Relace75
TREATMENT_ENUM
TREAT_ID Char(10) UNN (PK)
TREAT_NAME VarChar(50)
SEX_ENUM
SEX_ID Numeric(2,0) UNN (PK)
VALUE VarChar(10)
I
TMJ_DG_ENUM
TMJ_DG_ID Numeric(5,0) UNN (PK)
TMJ_DG_DESC NVarChar(256)
Relace91
Relace79
SEX
Relace89
TMJ_TREATMENT
TMJ_TREAT_ID Numeric(5,0) UNN (PK)
TMJ_VISIT_ID Numeric(5,0) NN (FK)
TREAT_ID Char(10) NN (FK)
TMJ_TREAT_DESC VarChar(256)
Relace92
TMJ_VISIT
TMJ_VISIT_ID Numeric(5,0) UNN (PK)
EVENT_ID Numeric(5,0) NN (FK)
TMJ_DG_ID Numeric(5,0) NN (FK)
TREATMENT NVarChar(2048)
EVENT
EVENT_ID Numeric(5,0) UNN (PK)
PATIENT_ID Numeric(5,0) NN (FK)
DATE DateTime
Relace74
1
Relace76
PATIENT
PATIENT_ID Numeric(5,0) UNN (PK)
SEX_ID Numeric(2,0) NN (FK)
FIRST_NAME VarChar(20)
SURNAME VarChar(20)
BIRTH_NUM VarChar(11) U
ADDRESS VarChar(40)
BIRTH_DATE DateTime
TOOTH
ATM_ENUM
ATM_ID Numeric(5,0) UNN (PK)
ATM_DESC NVarChar(256)
GINGIVA_EXAM
Relace80
Relace88
MOUTH_OPENING_ENUM
MOUTH_OPENING_ID Numeric(5,0) UNN (PK)
MOUTH_OPENING_DESC NVarChar(256)
GINGIVA
GINGIVA_ID Numeric(5,0) UNN (PK)
TOOTH_EVENT_ID Numeric(5,0) NN (FK)
DDA TinyInt
DDC TinyInt
UNCUT TinyInt
ERUPTION TinyInt
TMJ_ANAMNESIS
ANAMNESIS_ID Numeric(5,0) UNN (PK)
TMJ_VISIT_ID Numeric(5,0) NN (FK)
ANAM_TEXT VarChar(1024)
SURGERY Char(1)
SURGERY_DESC VarChar(256)
INJURY Char(1)
INJURY_DESC VarChar(256)
ALLERGY Char(1)
ALLERGY_DESC VarChar(256)
STRESS Char(1)
VERTEBALG Char(1)
VA_PHYSIOTER Char(1)
VA_PHYSIOTER_DESC VarChar(256)
BRUXISM Char(1)
ORTHO_THERAPY Char(1)
O_THERAPY_DATE DateTime
O_THERAPY_LEN Numeric(3,0)
O_THERAPY_DESC VarChar(1024)
TMJ_EXAM
TMJ_EXAM_ID Numeric(5,0) UNN (PK)
TMJ_VISIT_ID Numeric(5,0) NN (FK)
MUSCLES_PALP Char(1)
ATM_ID Numeric(5,0) NN (FK)
MOUTH_OPENING_ID Numeric(5,0) NN (FK)
MOUTH_OPENING_OTHER NVarChar(256)
OCCLUSION_ID Numeric(5,0) NN (FK)
OCLUSION_OTHER NVarChar(256)
DENTITION_STATE_ID Numeric(5,0) NN (FK)
DENTITION_STATE_DESC NVarChar(256)
DENTITION_STATE_NOTE NVarChar(256)
PATHO_SOUND_PHENOMENON Char(1)
P_S_PHENOM_TYPE VarChar(256)
P_S_PHENOM_LOC VarChar(256)
DYNAM_TEST_ID Numeric(5,0) NN (FK)
RTG_ID Numeric(5,0) NN (FK)
RTG_DESC NVarChar(256)
SPACE_1_1 Numeric(2,0)
Relace82
Relace83
OCCLUSION_ENUM
OCCLUSION_ID Numeric(5,0) UNN (PK)
OCCLUSION_DESC NVarChar(256)
Relace84
Relace85
DENTITION_STATE_ENUM
DENTITION_STATE_ID Numeric(5,0) UNN (PK)
DENTITION_STATE_DESC NVarChar(256)
Relace86
Relace87
DYNAM_TEST_ENUM
DYNAM_TEST_ID Numeric(5,0) UNN (PK)
DYNAM_TEST_DESC NVarChar(256)
RTG_ENUM
RTG_ID Numeric(5,0) UNN (PK)
RTG_DESC NVarChar(256)
PARO_RIGOROUS
PARODONT_ID Numeric(5,0) NN (FK)
EXAM_TYPE Char(1)
MESIAL_DEPTH Numeric(5,0)
DISTAL_DEPTH Numeric(5,0)
PARO_RIG

Figure E.1: The first part of the E-R model of database layer for DentCross Component.

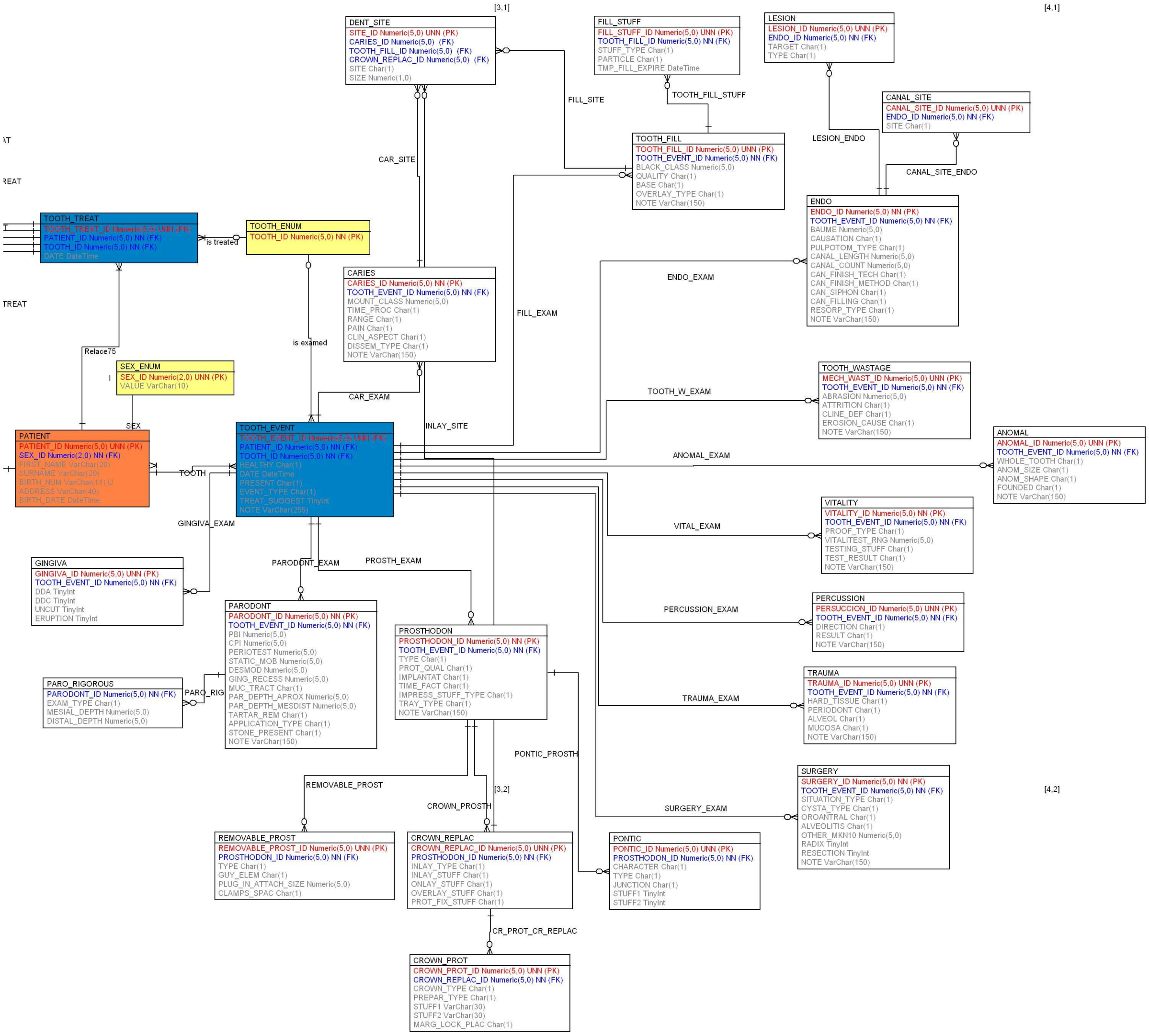


Figure E.2: The second part of the E-R model of database layer for DentCross Component.

Bibliography

[1] Deloitte Development LLC. Clinical Content: The Essential Currency of Clinical Information Systems. Deloitte Inc., 2007.

[2] Stroetmann VN, Kalra D, Lewalle P, Rector A, et al. Semantic Interoperability for Better Health and Safer Healthcare Deployment and Research Roadmap for Europe. SemanticHEALTH Report, January 2009.

[3] Lopez D.M, Blobel G.M.E. A development framework form semantic interoperable health information systems. Int. J. of Medical Informatics 2009; 78:83-103.

[4] Blobel BG, Engel K, Pharow P. Semantic interoperability - HL7 Version 3 compared to advanced architecture standards. Methods Inf Med. 2006; 45(4):343-53.

[5] Kalra D, Blobel BG. Semantic interoperability of EHR systems. Stud Health Technol Inform. 2007;127:231-45.

[6] Technical report. ISO/TR 20514 – Health informatics – Electronic health record – Definition, scope, and context. ISO. 2005.

[7] Veltman KH. Syntactic and Semantic Interoperability: New Approaches to Knowledge and the Semantic Web. The New Review of Information Networking 2001; 7: 159-84.

[8] European Committee for standardisation (CEN), Technical Committee CEN/TC 251: European Standard ENV 13606, ”Health informatics - Electronic healthcare record communication”

[9] Health Level Seven, Inc. (homepage on the internet) Health Level 7. Available from: http://www.hl7.org (cited Mar 24, 2011).

[10] The Commission of The European Communities. Commission Recommendation of 2 July 2008 on cross-border interoperability of electronic health record systems. Official Journal of the European Union, 2008; 190: 37-43. Available from: http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2008:190:0037:0043:EN:PDF (cited Mar 24, 2011).

[11] U.S. National Library of Medicne. SNOMED CT. Available from: http://www.nlm.nih.gov/research/umls/Snomed/snomed_main.html (cited Mar 24, 2011).

[12] The International Health Terminology Standards Development Organisation (homepage on the internet). SNOMED CT. Available from: http://www.ihtsdo.org/snomed-ct (cited Mar 24, 2011).

[13] Spidlen J. Database representation of medical information and medical guidelines (In Czech). Master Thesis, Faculty of Mathematics and Physics, Charles University in Prague, Prague, 2002.

[14] Kolesa P. The Analysis and Implementation of EHR Application Server. (In Czech). Master Thesis, Faculty of Mathematics and Physics, Charles University in Prague, Prague, 2004.

[15] Dioszegi M. Implementation of Modern Approaches to EHR (In Slovak). Bachelor Thesis, Faculty of Mathematics and Physics, Charles University in Prague, Prague, 2007.

[16] Spidlen J. Electronic Health Record and Telemedicine. PhD Thesis, 1. Faculty of Medicine, Charles University in Prague, Prague, 2005.

[17] Spidlen J, Hanzlicek P, Zvarova J. MUDRLite – Health Record Tailored to Your Particular Needs. In: Duplaga M, et al. (eds). Transformation of Healthcare with Information Technologies. Amsterdam: IOS Press; 2004. pp 202-209.

[18] ISO OSI model. Wikipedia The Free Encyclopedia. Available from: http://en.wikipedia.org/wiki/OSI_model (cited Mar 24, 2011).

[19] CEN EN 13606. Health informatics – Electronic health record communication – Part 1: Reference model. European Committee for standardisation, 2007.

[20] Martínez-Costa C, Menarguez-Tortosa M, Fernandez-Breis JT. An approach for the semantic interoperability of ISO EN 13606 and OpenEHR archetypes. Journal of Biomedical Informatics 2010, 43: 736–746

[21] Smith B, Ceusters W. HL7 RIM: An Incoherent Standard. In Ubiquity: Technologies for Better Health in Agigng Societies. Amsterdam: IOS Press, 2006; 124: 133-138.

[22] U.S. National Library of Medicine. Unified Medical Language System (UMLS) (homepage on the internet). Available from: http://www.nlm.nih.gov/research/umls/ (cited Mar 24, 2011).

[23] HL7 Harminization Meetings. Available from: http://www.hl7.org/events/harmonization/index.cfm (cited Mar 24, 2011).

[24] Ministry of Health of the Czech Republic (homepage on the internet), Data Standard of MH CR - DASTA and NCLP. Available from: http://ciselniky.dasta.mzcr.cz (cited Mar 24, 2011).

[25] Institute of Health Information and Statistics of the Czech Republic (homepage on the internet). Available from: http://www.uzis.cz (cited Mar 24, 2011).

[26] European Committee for standardisation (CEN), Technical Committee CEN/TC 251: European Standard EN 13606, "Health informatics - Electronic health record communication"

[27] NEMA - Medical Imaging & Technology Alliance (homepage on the internet), DICOM. Available from: http://dicom.nema.org (cited Mar 24, 2011).

[28] IZIP Ltd. (homepage on the internet), Internet Access to Patient Health Care Information. Available from: http://www.izip.cz (cited Mar 24, 2011).

[29] epSOS – a European eHealth Project (homepage on the internet). Available from: http://www.epsos.eu (cited Mar 24, 2011).

[30] IZIP Ltd. The Electronic Patient Health Account. Available from: http://www.ehealthconference2006.org/pdf/Sykorova_art.pdf (cited Mar 24, 2011).

[31] Mead Ch.N. Data Interchange Standards Healthcare IT – Computable Semantic Interoperability: Now Possible but Still Difficult, Do We Really Need a Better Mousetrap? J. of Healthcare Information Management 2006; 20:71-8.

[32] EurMISE.org. The project of the "Information Society" programme. (homepage on the internet). Available from: http://www.euromise.org/research/news.html (cited Mar 24, 2011).

[33] openEHR (homepage on the internet). openEHR – future-proof and flexible EHR specifications. Available from: http://www.openehr.org (cited Mar 24, 2011).

[34] Blobel B. Application of the component paradigm for analysis and design of advanced health system architectures. International Journal of Medical Informatics 2000; 60: 281–301.

[35] Hinchley A. Understanding Version 3. Munich, Alexander Mönch Publishing 2005.

[36] HL7 Inc. (homepage on the internet), HL7 Version 3 - January 2009. Available from: http://www.hl7.org/v3ballot/html/welcome/environment (cited Mar 24, 2011).

[37] Beale T, Heard S. Architecture Overview. The openEHR Foundation. Rev. 1.1.1, 2008.

[38] Garde S, Knaup P, Hovenga E.J.S, Heard S. Towards Semantic Interoperability for Electronic Health Records: Domain Knowledge Governance for openEHR Archetypes. Methods Inf Med 2007; 46: 332–343.

[39] Beale T, Heard S. Archetype Definitions and Principles. The openEHR Foundation. Rev. 1.0, 2007. Available from:

http://www.openehr.org/releases/1.0.2/architecture/am/archetype_principles.pdf (cited Mar 24, 2011).

[40] Jane Currey. HL7 Templates – A means to Manage Complexity. HL7 International, 2005. Available from: http://www.hl7.org/Library/Committees/template/HL7 Templates - Jane Currey.ppt (cited Mar 24, 2011).

[41] Beale T. Archetypes: Constraint-based domain models for future-proof information systems. In: Baclawski K, Kilov H, editors. Eleventh OOPSLA Workshop on Behavioral Semantics: Serving the Customer. Northeastern University, Boston, 2002, 16-32.

[42] Beale T, Heard S. Archetype Definition Language (ADL). The openEHR Foundation. Rev. 1.3.1, 2004.

[43] Beale T, Heard S, Kalra D, Lloyd D. EHR Information Model. The openEHR Foundation. Rev. 5.1.1, 2008.

[44] openEHR Foundation. Clinical Knowledge Manager. Available from: http://www.openehr.org/knowledge/ (cited Mar 24, 2011).

[45] Logical Observation Identifiers Names and Codes — LOINC (homepage on the Internet). Available from: http://loinc.org (cited Mar 24, 2011).

[46] Heard S, et al. Templates and Archetypes: how do we know what we are talking about? Available from: http://www.openehr.org/publications/archetypes/templates_and_ archetypes_heard_et_al.pdf (cited Mar 24, 2011).

[47] van Ginneken AM, Stqam H, van Mulligen EM, de Wilde M, van Mastrigt R, van Bemmel JH. ORCA: the versatile CPR. Methods Inf Med 1999; 38: 332-338.

[48] Hanzlicek P, Spidlen J, Heroutova H, Nagy M. User Interface of MUDR Electronic Health Record. Int J Med Inform 2005; 74: 221-227.

[49] Spidlen J, Hanzlicek P, Riha A, Zvarova J. Flexible Information Storage in MUDRII EHR. Int J Med Inform 2006; 75: 201-208.

[50] Hanzlicek P, Spidlen J, Nagy M. Universal electronic health record MUDR. Studies in health technology and informatics, Amsterdam: IOS Press 2004, 105: 190-201.

[51] Chalmers J, et al. WHO-ISH Hypertension Guidelines Committee. 1999 Guidelines for Management of Hypertension. J Hypertens 1999; 17:151-185.

[52] Entity-relationship model. Wikipedia The Free Encyclopedia. Available from: http://en.wikipedia.org/wiki/Entity-relationship_model (cited Mar 24, 2011).

[53] Chen R, Klein GO, Sundvall E, Karlsson D, Ahlfeldt H. Archetype-based conversion of EHR content models: pilot experience with a regional EHR system. BMC Med Inform Decis Mak. 2009; 9: 33.

[54] Connected Digital Health in Denmark. Discussion of the clinical content/archetype track. Sign-off discussion, SFI/archetype track, Available from: http://www.sst.dk/~/media/Indberetning%20og%20statistik/Terminologi/Modelarbejde/Arketyper/Engelske%20rapporter/SFI_archetype_track.ashx (cited Mar 24, 2011).

[55] Leslie H, Heard S Garde S, McNicoll I. Engaging Clinicians in Clinical Content: Herding Cats or Piece of Cake? Stud Health Technol Inform. 2009; 150:125-9.

[56] Tomeckova M et al. Minimal data model of cardiological patient. (in Czech) Cor et Vasa 2002; 4: 123.

[57] Mares R, Tomeckova M, Peleska J, Hanzlicek P, Zvarova J. User interface for patients' database systems – an example of application for data collection using minimal data model of cardiological patient (in Czech). Cor et Vasa 2002; 4: 76.

[58] EuroMISE. Centre of Biomedical Informatics (CBI) (homepage on the internet). Available from: http://www.euromise.org/cbi/cbi.html (cited Mar 24, 2011).

[59] MySQL table patcher. Available from: http://bisqwit.iki.fi/source/sqlupdate.html (cited Mar 24, 2011).

[60] Best SoftTool. SQLdbCompare tool. Available from: http://bestsofttool.com/SQLDBCompare/SDC_Introduction.aspx (cited Mar 24, 2011).

[61] Spectral Core. Omega Sync Pro. Available from: http://www.spectralcore.com/omegasync/ (cited Mar 24, 2011).

[62] AdeptSQLTools. The Adept SQL Diff tool. Available from: http://www.adeptsql.com (cited Mar 24, 2011).

[63] Nagy M. Synchronisation of Relational Schemas. Master Thesis at Faculty of Mathematics and Physics, Charles University in Prague, 2005.

[64] EuroMISE Centre Schemagic project. Available from: http://schemagic.sourceforge.net (cited Mar 24, 2011).

[65] Freeman E. Head First Design Patterns. O'Reilly, 2004.

[66] McDonald C, Huff S, Mercer K, Hernandez JA, Vreeman DJ. Logical Observation Identifiers Names and Codes (LOINC) Users' Guide. The Regenstrief Institute, Indianopolis, 2010.

[67] Hanzlicek P, Preckova P, Zvarova J. Semantic Interoperability in the Structured Electronic Health Record. Ercim News 2007, 69:52-3.

[68] EuroMISE s.r.o. Modeller Application. Available from: http://infspol.euromise.com/ (cited Mar 24, 2011).

[69] Zvarova J, Dostalova T, Hanzlicek P, Teuberova Z, Nagy M, Seydlova M, Eliasova H, Simkova H. Electronic Health Record for Forensic Dentistry. Methods of Information in Medicine 2008. 47: 8–13.

[70] Muller L, Psutka J, Smidl L. Design of Speech Recognition Engine. TSD 2000, Lecture Notes in Artificial Intelligence, Heidelberg: Springer; 2000.

[71] Bauer Ch, King G. Hibernate in Action. Manning, 2004.

[72] Walls C, Breidenbach R. Spring in Action. Manning, 2005.

[73] Spring Richclient project (homepage on the internet). Available from: http://spring-rich-c.sourceforge.net/1.0.0/index.html (cited Mar 24, 2011).

[74] Sun Corporation. Java EE at a Glance. Sun Corporation, Available from: http://java.sun.com/javaee/, (cited Mar 24, 2011).

[75] Tidwell D. XSLT. O'Reilly, 2001.

[76] openEHR Foundation. The openEHR Java Reference Implementation Project. Available from: http://www.openehr.org/projects/java.html (cited Mar 24, 2011).

[77] Beale T. Archetype Object Model. The openEHR Foundation. Rev. 2.0.2, 2008.

[78] WHO (homepage on the Internet). International Classification of Diseases. Available from: http://www.who.int/classifications/icd/en (cited Mar 24, 2011).

[79] The openEHR Foundation (homepage on the Internet). Relationship between openEHR Specifications and HL7 v3 Standards. Available from: http://www.openehr.org/206-OE.html (cited Mar 24, 2011).

[80] Beale T, Frankel H. Extract Information Model. The openEHR Foundation. Rev. 2.0, 2007.

[81] Dolin RH, et al. The HL7 Clinical Document Architecture. J Am Med Inform Assoc. 2001; 8(6):552-69.

[82] Beale T, Heard S, Kalra D, Lloyd D. Data Types Information Model. The openEHR Foundation. Rev. 5.1.1, 2008.

[83] Beale T, Heard S. openEHR Templates. The openEHR Foundation. Rev. 1.0, 2009.

[84] Goek M. Introducing an openEHR-Based Electronic Health Record System in a Hospital. Master Thesis. Department of Medical Informatics, University of Goettingen, Germany. 2008.

[85] Artemis project. OWLmt – OWL mapping tool. 2005. Available from: http://www.srdc.metu.edu.tr/artemis/owlmt (cited Mar 24, 2011).

[86] Burger A, Davidson D, Baldock R. Anatomy Ontologies for Bioinformatics. Computational Biology, Berlin: Spinger, 2008. Ontology Alignment and Merging, 133-149.

[87] D. Moner, J. Maldonado, D. Bosca, C. Angulo, M. Robles, D. Perez, P. Serrano. CEN EN 13606 Normalisation Framework Implementation Experiences. In: B.Blobel, E. Hvannberg, V. Gunnarsdottir (Eds). Seamless Care - Safe Care, IOS Press 2010. 136-142.

[88] Interpol. Disaster victim identification guide. Lyon, France: Interpol; 2005. Available from: http://www.interpol.com/public/disastervictim/ (cited Mar 24, 2011).

[89] Eichelberg M, Aden T, Riesmeier J, Dogac A, Laleci GB. A Survey and Analysis of Electronic Healthcare Record Standards. ACM Comp Surv, 2005 Dec; 37(4): 277 - 315.

[90] Ma Ch, Frankel H, Beale T, Heard S. EHR Query Language (EQL) – A Query Language for Archetype-Based Health Records. MEDINFO proceedings, IOS Press, 2007, 397-401.

[91] Martínez-Costa C, Menarguez-Tortosa M, Fernandez-Breis JT, Maldonado JA. A model-driven approach for representing clinical archetypes for Semantic Web environments. Journal of Biomedical Informatics 2009, 45: 150–164.

[92] IT-014 Health Informatics Committee. Action Items. ISO/TC215 Health Informatics Plenary and Working Group Meeting, October 2009.

[93] Software R&D Center, Middle East Technical University. Artemis Project Homepage. Available from: http://www.srdc.metu.edu.tr/webpage/projects/artemis/home.html (cited Mar 24, 2011).

[94] Bicer V et al. Archetype-Based Semantic Interoperability of Web Service Messages in the Health Care Domain. Int'l Journal on Semantic Web & Information Systems. 2005; 1(4): 1-22.

[95] Hoy D, Hardiker NR, McNicoll I.T, Westwell P, Bryansa A. Collaborative development of clinical templates as a national resource. International Journal of Medical Informatics. 2009; 78(1): S3-S8.

[96] Qamar R. Semantic mapping of clinical model data to biomedical terminologies to facilitate interoperability. PhD thesis, University of Manchester, 2008.

[97] Orgun B. Interoperability in heterogeneous medical information systems using smart mobile agents and HL7 (EMAGS). Master Thesis, Macquarie University, 2003.

[98] National eHealth Transition Authority. Standards for eHealth Interoperability: An eHealth Transition Strategy. NEHTA, Sidney, 2007.