Methods of Flexible Information Storage in MUDR^{II} Health Record

Josef Spidlen, Petr Hanzlicek, Antonin Riha, Jana Zvarova

European Center for Medical Informatics, Statistics and Epidemiology – Cardio,
Institute of Computer Science, Academy of Sciences of the Czech Republic,
Prague, Czech Republic

Abstract

An important research task of the EuroMISE Center – Cardio is the applied research in the field of electronic health record (EHR). In this frame we have proposed a mathematical meta-description of a flexible information storage model and described global system architecture of a 3-layer EHR application. In the paper both of this proposals are briefly described. We have tested the functionality of our solution implementing a pilot EHR application named "MUltimedia Distributed Record" (MUDR). According to our experience and test results gained from the MUDR EHR usage we describe an advanced solution MUDR^{II}, which can be applied as a kernel of an EHR application. Since we are a research center and not a commercial company, in our kernel we do not handle all parts needed for a large commercial use in a local environment. To give publicity to our solutions we negotiate with two companies about the application of our EHR kernel into their hospital information systems (HIS) used in the Czech Republic.

Keywords:

Health Records, Information Storage and Retrieval.

Introduction

The European Center for Medical Informatics, Statistics and Epidemiology – Cardio (EuroMISE Center – Cardio) is focused on new approaches to the electronic health record (EHR) design, including electronic medical guidelines and intelligent systems for data mining and decision support [1]. The research in the field of data storage, data acquisition and data mining [2] was inspired by several European projects [3], mostly by the I4C and TripleC projects [4,5]. Studying the CEN TC 251 standards [6] together with cooperation with physicians resulted in a list of 15 requirements on EHR [7]. The proposed model was mostly influenced by two of them:

- structured way of data storage combined with free text,
- dynamically extensible and modifiable set of collected attributes without any change of the database structure.

The main goal of our work 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 free-text based health record. We didn't want to choose a particular database or an operating system; we tried to propose an open information storage meta-model with various implementation possibilities as inspirations and recommendations for EHR software vendors on the market. To test the functionality of our solutions we have developed a pilot EHR application called "MUltimedia Distributed Record" (MUDR) [8].

Flexible Information Storage Model

Because of the requirement of a dynamically extensible and modifiable set of collected attributes, it is complicated to use a classical relational database structure with columns corresponding to the gathered features as the basis of the information storage. In our solution, two main structures described by the graph theory expressions are used instead. The collected attributes and relations among them are stored in a so-called knowledge base. Another graph structure named "data-files" is used to store the patient data itself.

The Knowledge Base

The main function of the knowledge base is to store the hierarchy of the collected attributes and relations among them. We define the *MUDR KB* knowledge base as an oriented graph:

$$MUDR\,KB = (\mathbf{V}_{kb}, \mathbf{E}_{kb}).$$

The graph vertices, representing symptoms, elements $n \in \mathbf{V}_{kb}$, are called knowledge nodes. Each node is a quaternion:

$$n = (\gamma, \varphi, \omega, \epsilon_{nd}),$$

where γ is a unique identifier among all knowledge nodes, φ is a name constructed as a mnemotechnical string identifier, ω is a data type of the node and ε_{nd} contains administrative data about the node such as identification of the user who has entered or deleted the node. Later in this text we use the square brackets [] to index the items of ordered sets.

Edges $e \in \mathbf{E}_{kb}$, representing relations between symptoms, are also defined by quaternions:

$$e = (\alpha, \beta, \tau, \epsilon_{ed}),$$

where $\alpha, \beta \in \boldsymbol{V}_{kb}$ determine starting and ending vertices of the edge, τ is the edge type determining the type of the relation between α and β and ε_{ed} contains administrative information about the edge e (e.g. the creator of the edge).

A special edge type called *inferior* (*inf*) can be distinguished. The edge of this type leads from a parent vertex to a child vertex in the graph. Thereby a hierarchical relation among knowledge nodes is set. Using other edge types we can add additional medical knowledge into the knowledge base like equivalence, contraindications, usable scales, etc. Their detailed description goes out of the scope of this text. The knowledge nodes and edges of type inferior create an oriented forest with a few trees. These trees are called *knowledge base domains*. Each domain associates nodes used to a similar aim. The domain $\boldsymbol{V}_s$, $\boldsymbol{V}_s \subseteq \boldsymbol{V}_{kb}$, is used to store collected attributes of a patient. Nodes in this domain are called *semantic types*. Other domains can be used for example to store the International Classification of Diseases and Related Health Problems (ICD10) or the Anatomical Therapeutic Chemical Classification of drugs (ATC).

Nodes with the same parent are called siblings – $n_1 \Diamond n_2$.

$$n_1 \Diamond n_2 \overset{def}{\iff} \exists\, e_1, e_2 \in \mathbf{E}_{kb}, e_1[\alpha] = e_2[\alpha], e_1[\beta] = n_1, e_2[\beta] = n_2, e_1[\tau] = e_2[\tau] = \text{"}inf\text{"}.$$

Nodes without any parents are called knowledge domain roots. For $n \in \boldsymbol{V}_{kb}$ we define:

$$n \in \mathbf{R}_{kb} \overset{def}{\iff} (\forall e \in \mathbf{E}_{kb}, e[\tau] = \text{"}inf\text{"} \implies e[\beta] \neq n).$$

We require that the name φ is unique among siblings. The same we require among domain roots, so:

$$(n_1 \Diamond n_2) \lor (n_1, n_2 \in \mathbf{R}_{kb}) \implies \implies (n_1 = n_2) \lor (n_1[\varphi] \neq n_2[\varphi]).$$

This uniqueness enables constructing a dot-separated node full name, which can be used to identify a knowledge node among other knowledge nodes.

The node data type is important for the semantic types first of all. We distinguish basic data types like numbers, strings or boolean variables, multimedia data types like pictures, audios and videos and reference data types. In the latest MUDR^{II} version we have added some useful data types like enumerations and the multimedia data types group was extended. Now, the Multipurpose Internet Mail Extension Type (RFC 2048) is used to determine the multimedia type.

The Data Files

Patient data themselves are stored in a graph structure *MUDR DF*, realized as an oriented forest:

$$MUDR\ DF = (\mathbf{D}_{df}, \mathbf{E}_{df}).$$

Information about one patient corresponds exactly to one tree in the forest. A graph vertex $d \in \boldsymbol{D}_{df}$ is a quaternion:

$$d = (\delta, \sigma, \lambda, \epsilon_{df}).$$

In this case δ is a unique identifier among all data files, $\sigma \in \boldsymbol{V}_s$ is a semantic type of the data file, λ is a data file value and ε_{df} contains administrative information like the identification of the person who has entered or confirmed the stored value. The domain of the value is implicitly determined by the semantic type, including the data type within.

The edges $e \in \boldsymbol{E}_{df}$ do not have any types, they express the "parent – child" hierarchical relation among stored data, so we can simply write $\boldsymbol{E}_{df} \subseteq \boldsymbol{D}_{df} \times \boldsymbol{D}_{df}$. For the *MUDR DF* we require two expressions to be valid:

- $\forall d \in \mathbf{D}_{df}\ (\exists d' \in \mathbf{D}_{df}, (d', d) \in \mathbf{E}_{df} \lor d[\sigma] \in \mathbf{R}_{kb})$
- $d, d' \in \mathbf{D}_{df}, (d', d) \in \mathbf{E}_{df} \implies$ $\implies \exists e \in \mathbf{E}_{kb}\ e[\alpha] = d[\sigma], e[\beta] = d'[\sigma], e[\tau] = \text{"}inf\text{"}$

The first condition means that each data file has a parent or its semantic type is a domain root. The second one expresses that if a data file d has a parent d', than the semantic type of d is the child of the semantic type of d'. It simply means that the *MUDR DF* corresponds to the *MUDR KB*. A small example can be seen in the Figure 1.

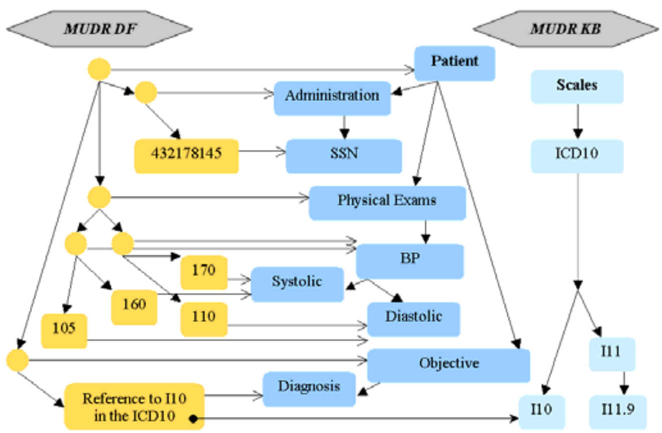

Figure 1 - Fragment of the MUDR Information Storage Model

MUDR Architecture

Classical MUDR Architecture

The MUDR EHR is based on a three-layer architecture with a data layer, an application layer and a user interface. This decomposition enables separating different system modules to small functional parts, which makes the system more flexible. In our system we define the global architecture with communication interfaces based predominately on XML and HTTP. We also define an interface for connecting medical guidelines formalized in the form of dynamic libraries. These libraries extend the capabilities of the MUDR EHR.

MUDR^{II} Extensions

MUDR^{II} architecture extends the basic three-layer architecture in the way shown in the Figure 2. The communication between the MUDR DB Server and the MUDR Application Layer Service uses the MUDR DB Connection Module, which enables implementing the data layer on various database platforms. The application layer chooses the right module for the database currently used. Using this scheme, the application layer communicates transparently with the data layer using always the identical interface.

The main difference lies in the application layer interface, used to provide the functionality of the application layer. The application layer integrates communication modules to communicate with different client types. The main communication module is called MUDR WS. This module provides objects using the MUDR .NET Remoting API (MUDRNRAPI). Using .NET Remoting, the remote call of methods of the shared objects is possible. This functionality is utilized by another application layer component called MUDR Web Service. This web service makes the MUDR Web Service Application Interface (MUDRWSAPI) accessible to common clients. A typical general practitioner (GP) uses a PC on his table to work with the MUDR EHR remotely. The communication is enabled by the HTTP Server installed on the application server. The com-mands and parameters are encoded using the SOAP standard.

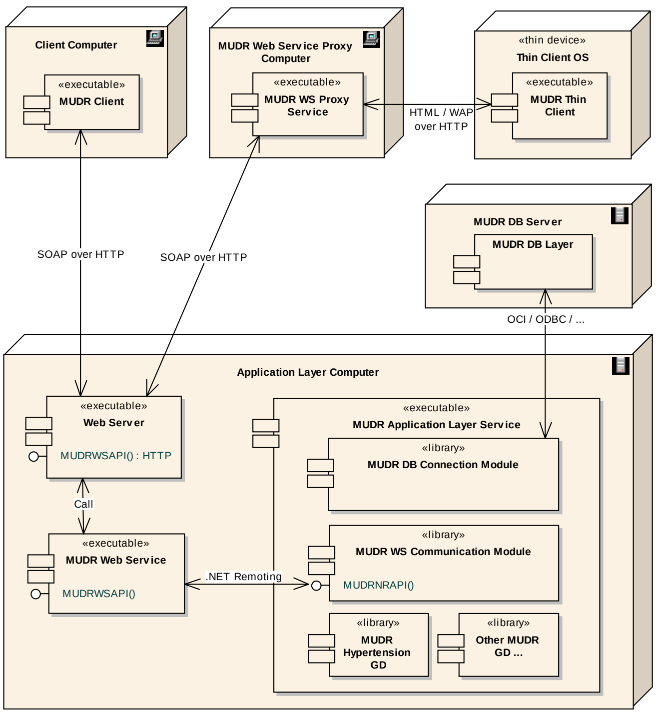

Figure 2 – MUDR^{II} Architecture

For the eventual usage of MUDR thin clients in the form of HTML and WAP browsers we use a MUDR WS Proxy Service. This service is implemented as a Common Gateway Interface (CGI) program; it provides classical HTML or WAP pages to thin clients and appears in the role of a classical MUDR Web Service client at the other side.

Results

Pilot Implementation

To test our ideas of a flexible information storage model and the functionality of presented EHR architecture we have implemented a pilot EHR application. Our implementation consists of many different modules.

The Oracle 9*i* database is used as the MUDR database layer platform. The Oracle database enables using features like nested tables or object references to implement the *MUDR KB* and *MUDR DF* structures as described above. Simultaneously we are proposing a way of implementation of these structures using a classical relational database.

To implement the application layer we have chosen the Microsoft Windows platform. The application layer runs as a win32 service developed using the C++ language with the Microsoft XML Parser. The communication is established using the Apache web server together with small proper CGI scripts. At the application layer we have connected two formalized medical guideline libraries in the form of win32 DLL libraries. One implements the 1999 WHO/ISH Guidelines for the management of hypertension [9] and the other one implements support for genetic data evaluation [10].

Two thick clients were implemented [11] as the user interfaces. We have also developed two CGI scripts to enable the usage of thin clients. One enables working with the MUDR EHR using the Internet Explorer browser and the other one emulates the MUDR EHR client on the Nokia 9110i browser [12].

Obtained Results

We have tested our implementation together with physicians in the ambulance of the EuroMISE Center where service is provided by cardiologists from two cooperating Czech hospitals. We have confirmed, that our information storage model is really flexible; it allows a dynamical change of the knowledge base and of the set of collected attributes. We have also verified the functionality of MUDR/MUDR^{II} architecture, which enables an easy decomposition of the system modules, an easy way of data sharing and the remote data access.

But we have found out that the bottleneck of these systems lies in the user interface. It is a very difficult problem to design an EHR user interface, which would be comfortable enough [13]. The problem in our case is more difficult because of the dynamically changing knowledge base content. To ensure a large use of an EHR system we should provide a user interface, which would be easy and fast to control.

Physicians and GPs in the Czech Republic are used to write their reports in the form of free texts. The structured way of storing information is very important for the additional information processing using for example data mining or knowledge discovery techniques. Unfortunately not all information can be structured, so our system enables storing structured as well as unstructured information.

Discussion

Nowadays, most hospital and clinical information systems vendors provide systems with a small possibility of structuring health record - very often only the administrative information and laboratory examinations are structured well but it is important to structure also other parts of the health record. We provide a flexible solution but it cannot be a stand-alone EHR application. It is unacceptable to force the users to enter the information twice, once into the EHR and then into the information system again. It is necessary to implement the EHR application as a part of a complex information system, which would bring benefit to the users – save their time and offer them additional information using data mining and knowledge discovery techniques.

Conclusion

Our interest is to increase the quality of EHR systems, to simplify data sharing and data migration among various EHR systems and to help in overcoming the classical free-text based health record. This is the way, which would increase the quality of healthcare, which brings benefit for the patient first of all. To do this, we have suggested common general principles of EHR system architecture and a model of flexible information storage without the implementation details. This makes possible to use various ways of implementation.

We also present an EHR solution MUDR^{II}. It is an open universal solution, which can be applied as the EHR kernel of a HIS. We have not implemented special modules needed for a large commercial use in a particular environment, e.g. modules for the communication with health insurance companies, laboratories, pharmacies or various modalities and public administration offices. There are many standards used in different countries and so it is impossible to give a worldwide universal solution.

Nowadays, we have started the negotiation about the implementation of our EHR kernel in hospital and clinical information systems of two big Czech companies. We hope this will put our solutions into the real practical use, which should bring advantages for the health care in the Czech Republic. Later we would like our EHR kernel to be used in a different HIS software worldwide.

Acknowledgments

The work was partially supported by the grant no. LN00B107 of the Ministry of Education of the Czech Republic.

References

[1] Rauch J: Mining for Statistical Association Rules. The Fifth Pacific/Asia Conference on Knowledge Discovery and Data Mining Industrial track and Workshop Proceeding Red. Joseph Fong and Michael Ng Hong Kong 2001, pp. 149-158.

[2] Riha A, Svatek V, Nemec P, Zvarova J: Medical Guideline as Prior Knowledge in Electronic Healthcare Record Mining. Data Mining III, WIT Press 2002, ISBN 1-85312-925-9, ISSN 1470-6326, pp. 809-818.

[3] Iakovidis I: Towards Personal Health Record: Current Situation, Obstacles and Trends in Implementation of Electronic Healthcare Record in Europe. Int J Med Inform 52, Nos. 1-3, 1998, pp. 105 –115.

[4] Pierik FH, Ginneken AM, Timmers T, Stam H, Weber RF: Restructuring Routinely Collected Patient Data: ORCA Applied to Andrology. Yearbook of Medical Informatics 98, Schattauer, Stuttgart 1998.

[5] Ginneken AM. The Computerized Patient Record: Balancing Efort and Benefit, Int J Med Inform 65, 2002, pp. 97-119.

[6] CEN/TC251, ENV 13606, parts 1 – 4.

[7] Zvarova J, Hanzlicek P, Spidlen J: Electronic Health Record in Cardiology: Pilot Application in the Czech Republic, Proceedings of Medical Informatics Symposium in Taiwan 2002, pp. 10-13.

[8] Hanzlicek P: Development of Universal Electronic Health Record in Cardiology. Proceedings of MIE 2002. Amsterdam IOS Press, 2002. ISBN 1-58603-279-8, ISSN 0926-9630, pp. 356-360.

[9] World Health Organization – International Society of Hypertension: Hypertension Guidelines, http://new.euromise.org/mgt/who1999/who1999.html.

[10] Spidlen J, Adaskova J, Heroutova H, Zvarova J, Mazura I: Statistical Processing of Genetic Information in the MUDR Health Record, International Anthropological Congress, Anthropology and Society 2003, Programme Abstracts, Charles University, Prague, Faculty of Science, ISBN 80-86561-06-2, p.190.

[11] Hanzlicek P, Spidlen J, Heroutova H: User interface of MUDR Electronic Health Record, Medical Informatics Europe MIE2003 Proceedings CD.

[12] Hanzlicek P, Spidlen J, Zvarova J: The Internet in Connecting Electronic Health Record Mobile Clients, Technology and Health Care, Vol. 10, Num 6, 2002, ISSN 0928-7329, pp. 502-503.

[13] Ginneken AM, Verkoien MJ, A Multi-disciplinary Approach to a User Interface for Structured Data Entry. MEDINFO 2001, pp. 693-697.

Address for correspondence

Josef Spidlen, EuroMISE Center – Cardio,

Institute of Computer Science AS CR,
Pod Vodarenskou vezi 2, 182 07 Prague 8, Czech Republic

Email: spidlen@euromise.cz