

Medical data presentation and acquisition application auto-generation on OpenEHR

Citation for published version (APA):

Hu, D., Wang, L., Yin, Z., Lu, X., & Duan, H. (2014). Medical data presentation and acquisition application auto-generation on OpenEHR. In *Proceedings of the 2014 International Conference on Information Technology and Management Engineering, ITME2014, 26-27 April, Hong Kong*

Document status and date:

Published: 01/01/2014

Document Version:

Accepted manuscript including changes made at the peer-review stage

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Medical Data Presentation and Acquisition Application Auto-generation on OpenEHR

HU Di, WANG Li, YIN Zi-ming, LU Xu-dong, and DUAN Hui-long

College of Biomedical Engineering and Instrument Science of Zhejiang University, Hangzhou, China

hudixidian@gmail.com

Abstract.“Two-level” Modeling was proposed by openEHR to achieve modeling Healthcare Information Model directly by domain experts, and “Two-level” modeling based Medical Data Platform has achieved meeting data storage needs dynamically by reading the openEHR domain model. On the basis of “Two-level” modeling based Medical Data Platform, a methodology of Medical Data Presentation and Acquisition (MDPA) Application Auto-generation on openEHR was proposed. Firstly, the openEHR domain model was extended and a Data Application Template was designed. Secondly, a What You See Is What You Get (WYSIWYG) Data Application Template Designer for domain experts was developed. Thirdly, a MDPA Application Framework was implemented. By dynamically loading the Data Application Template and the openEHR domain model, the MDPA Application Framework could generate the corresponding MDPA Application.

Keywords:“Two-level” Modeling • Data Application Template • What You See Is What You Get • Medical Data Presentation and Acquisition

1 Introduction

With the development of medical informatics, a variety of digital medical instruments and systems have accumulated a large amount of medical data. By making use of the medical data, the quality of medical treatment services has been greatly improved. On the other hand, the development of medical informatics makes the medical staff rely on the medical applications. However, due to the rapid development of medical knowledge and the complexity of medical business, the traditional software development method can not fully meet needs. If the medical staff could take part in the software development process and customize the medical applications by themselves, their needs could be met faster.

“Two-level” Modeling was proposed by openEHR [1]. It achieved modeling Healthcare Information Model directly by domain experts. The first level is the Information Level and the second is the Domain Knowledge Level. Reference Model (RM) on the Information Level conveys the general structures while the

archetypes on the Domain Knowledge Level define all kinds of healthcare concepts. The RM is stable, which makes the storage structure and other underlying structures not change along with knowledge updating or demands changing. Domain knowledge consisting of archetypes and templates can be defined directly by the domain experts, which has “put the clinician back in the driver's seat” [2].

Based on “Two-level” Modeling, the OpenEHR Medical Data Platform has been implemented. It can meet the data storage needs by dynamically reading the openEHR domain model at run time. WANG Li et al. have designed and implemented an Archetype-driven Biomedical Data Platform on Relational Database (ABDP) which could achieve flexible data storage by Archetype Relational Mapping (ARM). On the basis of ARM, the ABDP implemented Archetype Query Language (AQL) [3] through a set of web services to provide flexible data access. At the same time, Sergio Miranda Freire et al. have implemented an OpenEHR Data Platform on XML Database [4] and Aastha Madaan et al. have implemented one on NoSQL Database [5].

On the basis of the OpenEHR Data Platform which can map openEHR domain model to actual data storage, a methodology of auto-generating Medical Data Application from domain model is needed to meet the medical staffs' data needs easily.

Thilo Schuler et al. studied auto-generating Graphical User Interface (GUI) expressed by Mozilla XML User Interface Language (XUL) from openEHR archetypes. In the study, a set of generic controls, one for each data type, was customized. By traversing the archetypes' data items, the mapping GUI were generated [6]. Koray Atalag et al. studied generating Data Entry Forms from openEHR templates. The implemented application could map data items in templates to GUI elements and adjust the GUI elements according to the GUI Directives that added into the templates [7].

In these studies, the generating GUI could not be adjusted flexibly or arranged precisely. On the other side, there were nothing about data interaction between GUI and the data storage.

In this paper, a methodology of Medical Data Presentation and Acquisition (MDPA) Application Auto-generation on openEHR was proposed. Firstly the openEHR template was extended the information of GUI and Data Manipulation, and a Data Application Template was designed. Secondly, a What You See Is What You Get (WYSIWYG) Data Application Template Designer for domain experts was developed. Thirdly, a MDPA Application Framework was implemented which could dynamically load the Data Application Template, openEHR templates and archetypes and generate the MDPA Application that meet the data storage, presentation and acquisition needs.

2 Method

The methodology of MDPA Application Auto-generation on openEHR proposed in this paper is shown in Figure 1. Domain experts use the Data Application Tem-

plate Designer to read archetypes from the Archetype Repository for editing Data Application Templates and openEHR templates. The MDPA Application Framework generates the MDPA Application by loading archetypes and templates. The MDPA Application Framework registers archetypes and templates to the OpenEHR Data Platform to generate storage tables. At runtime, the MDPA Application manipulates the database using AQL and acquires results in dADL [8] format.

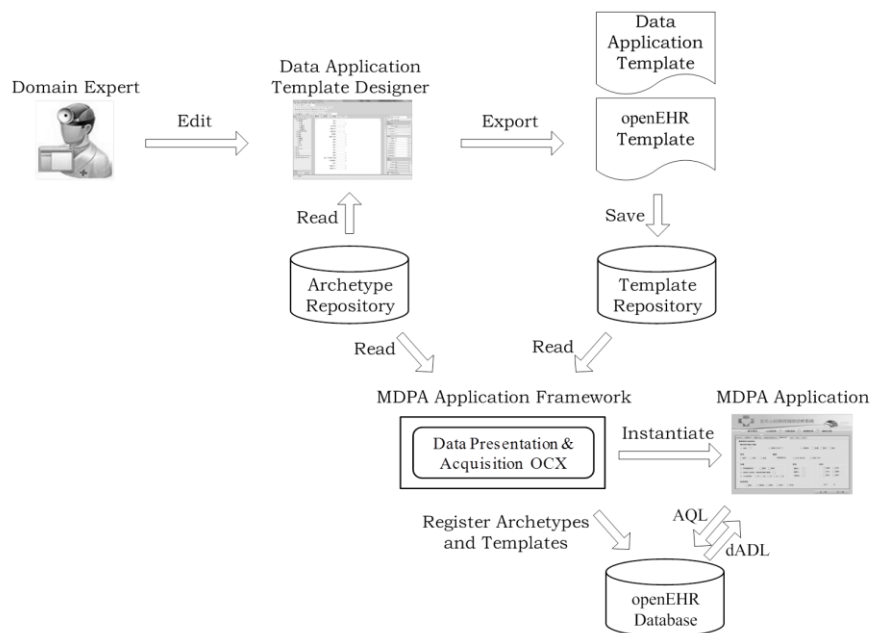


Fig. 1. Methodology of MDPA Application Auto-generation on openEHR

2.1 Data Application Template

OpenEHR templates would add constraints to archetypes and combine archetypes according to local needs. By making use of openEHR templates, medical applications could generate default data structures at runtime. Because openEHR templates would add only constraints rather than semantics to archetypes, they couldn't express the information of GUI or Data Manipulation. In this paper, the openEHR template was extended the information of GUI and Data Manipulation, and a Data Application Template was designed.

To express all kinds of information produced in activities of scientific research and clinical treatments, a series of basic data types were defined in the openEHR RM. The designed mapping from RM data types to GUI elements is shown in table 1. The XML tags used to express GUI elements are shown in table 2.

Table 1. Mapping from RM data types to GUI elements

RM Data Type	GUI Elements
Text	Label + Textbox
CodedText	Label + Combobox
Quantity	Label + Textbox+ Label
Count	Label + Textbox
DateTime	Label + Datetime
Duration	Label + Textbox+ Label
Ordinal	Label + Combobox
Boolean	Checkbox
Interval	Label + Textbox + Label + Textbox / Label + Datetime + Label + Datetime
MultiMedia	Label + Imagegrid
Proportion	Textbox + Label + Textbox
Identifier	Label + Textbox

Table 2. Graphical User Interfaces tags

GUI Element	Tag
Textbox	<textbox/>
Label	<label/>
Combobox	<combobox/>
Checkbox	<checkbox/>
Radiobox	<radiobox/>
Datetimectrl	<datetime/>
Flame	<frame/>
Groupbox	<groupbox/>
Group	<group/>
Page	<page/>
Image	<imagegrid/>
Line/ Curve/ Rectangle/ ...	<sketch/>

GUI Elements in the Data Application Template use attributes to express all the features. The structure of the Data Application Template is shown below:

```

<GUIElement1 Attribute1="Value1" ...>
  <AttributeNode1 Attribute2="Value2" ...>
    </AttributeNode1>
  <GUIElement2 ...>
    ...
  </GUIElement2>
</GUIElement1>

```

2.2 Data Application Template Designer

The interface of the Data Application Template Designer was divided into Toolbar, Archetype Panel, Page Area and Property Panel and the use flow of the Data Application Template Designer is shown in figure 2. In this way, the Data Application Template could be developed by domain experts in the WYSIWYG edit mode.

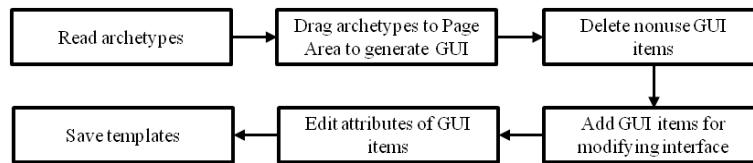


Fig. 2. Use flow of the Data Application Template Designer

2.3 Medical Data Presentation and Acquisition Application Framework

The Medical Data Presentation and Acquisition (MDPA) Application Framework is based on the OpenEHR Data Platform, as is shown in figure 3.

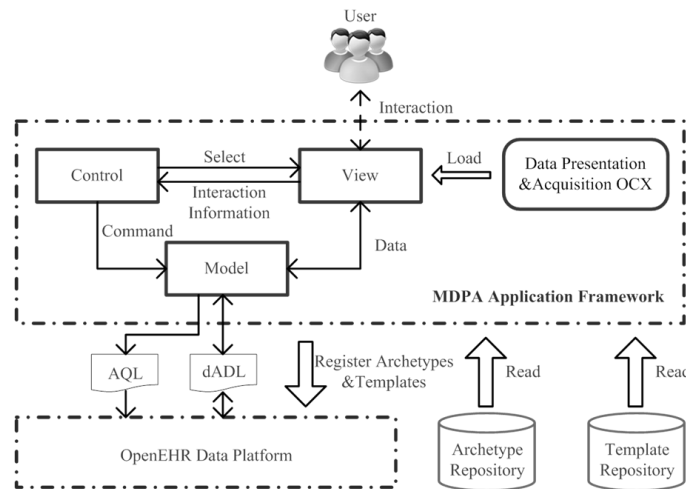


Fig. 3. Structure of the MDP Application Framework

Data Presentation and Acquisition OCX.

The Data Presentation and Acquisition OCX was developed using Object Linking and Embedding (OLE) Control eXtension (OCX) technology. It is used for reading the Data Application Template, displaying GUI and achieving Data Presentation and Acquisition. The upper application acquires the information expressed in the Data Application Template through OCX interfaces.

Medical Data Presentation and Acquisition Application Framework.

The MDPA Application Framework was developed in the Browser/Server (B/S) mode and set up in the structure of ASP.NET MVC3.0. The Model is the data's logic implementation. It acquires the state from and saves it into the OpenEHR Data Platform. The View is used to display interfaces. By loading the Data Presentation and Acquisition OCX, the View reads the Data Application Template, displays GUI and achieves Data Presentation and Acquisition. The Controller is used for processing the User Interaction Signals. In the end, the MDPA Application Framework achieves generating the MDPA Application that meet needs by reading the domain model consisting of a Data Application Template, openEHR templates and archetypes.

3 Result

The methodology in this paper was applied in the development of the Data Presentation and Acquisition Application for Alzheimer's Disease to be evaluated. By making use of the application, medical staffs who get permission can register patients, view visit records, and input results of examination and diagnosis.

3.1 Related Archetypes Defining

The archetypes related to the Alzheimer's Disease have not been contained in the public Archetype Repository, Clinical Knowledge Manager (CKM). Taking the existing archetypes as examples, we defined eight archetypes, each was about physical examinations, related diseases and drugs, and a set of questionnaires including activities of daily living (ADL), Clinical Dementia Rating (CDR), Geriatric Depression Scale (GDS), Mini Mental State Exam (MMSE), Montreal Cognitive Assessment (MoCA), and other cognition tests.

3.2 Corresponding Templates Developing

By using the Data Application Template Designer, the archetypes related to the Alzheimer's Disease were selected and displayed on the Archetype Panel. When the archetypes had been dragged from the Archetype Panel to the Page Area, the mapping GUI elements were generated, as is shown in figure 4.

The GUI Elements used for optimizing the interface were chosen from the Toolbar and created on the Page Area. The attributes of the GUI elements on the Page Area were set and their layout was modified in the WYSIWYG edit mode. After being developed, the Data Application Templates' interfaces are shown in figure 5.

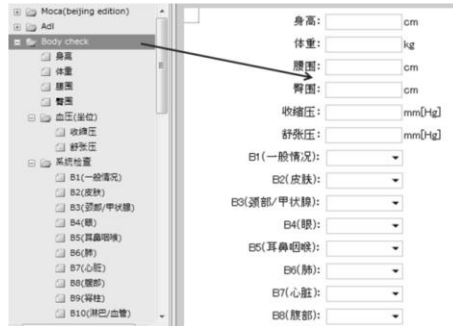


Fig. 4. Archetypes were dragged and mapping GUI elements were generated

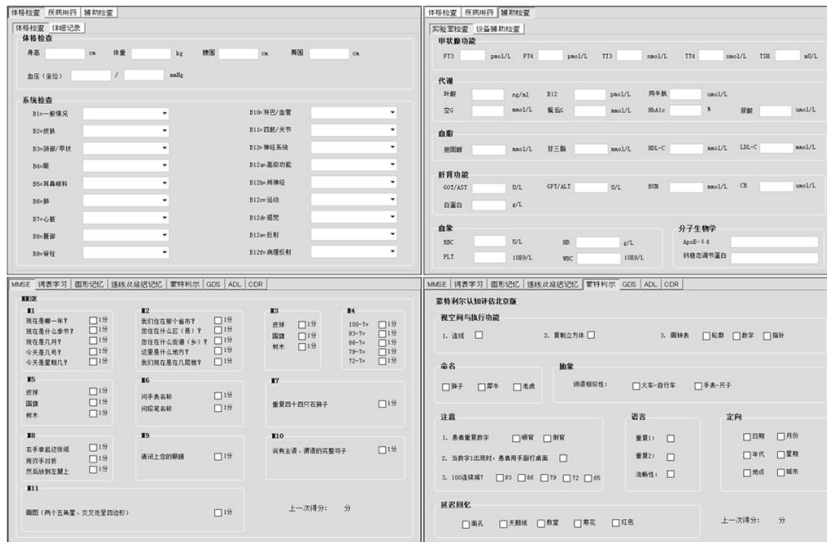


Fig. 5. Interfaces of the Data Application Templates corresponding to the Data Presentation and Aquisition Application for Alzheimer's Disease

3.3 Data Presentation and Aquisition Application for Alzheimer's Disease

The corresponding archetypes and templates were read from repositories by the MDPA Application Framework in this paper. After registering the archetypes and templates to the OpenEHR Data Platform, the Data Presentation and Aquisition Application for Alzheimer's Disease that meet needs of data storage, presentation and acquisition was generated.

If the demands changed, the only thing needed to be done was to modify the Data Application Templates and the openEHR templates by the Data Application Template Designer. The MDPA Application Framework was initialized again using the modified templates, and the new application that meet needs would be generated.

4 Summary

On the basis of “Two-level” Modeling and the OpenEHR Data Platform, a methodology of MDPA Application Auto-generation on openEHR was proposed. The openEHR template was extended the information of GUI and Data Manipulation, and a Data Application Template was designed. A WYSIWYG Data Application Template Designer for domain experts was developed and a MDPA Application Framework was implemented. By reading the Data Application Template and the openEHR domain model, the MDPA Application Framework could generate the MDPA Application that meet the data storage, presentation and acquisition needs. This methodology was proved to be effective in the development practice of the Data Presentation and Acquisition Application for Alzheimer's Disease.

Acknowledgement

This work was supported by the National High-tech R&D Program of China (2012AA02A601) and National Science and Technology Major Project of China (2013ZX03005012).

References

1. Kalra, D., Beale, T., and Heard, S.: The openEHR foundation, *Studies in health technology and informatics*, 115, 153-173 (2005)
2. Beale, T., Heard, S., Kalra, D., and Lloyd, D.: OpenEHR architecture overview, *The OpenEHR Foundation* (2006)
3. Beale, T., Heard, S., Kalra, D., and Lloyd, D.: Archetype query language (aql), *OpenEHR Foundation* (2005)
4. Freire, S. M., Sundvall, E., Karlsson, D., and Lambrix, P.: Performance of XML Databases for Epidemiological Queries in Archetype-Based EHRs, in *Scand Conf Health Inform*, Vol. 13, pp. 51-57 (2012)
5. Madaan, A., Chu, W., Daigo, Y., and Bhalla, S.: Quasi-Relational Query Language Interface for Persistent Standardized EHRs: Using NoSQL Databases, in *Databases in Networked Information Systems*, Springer, pp. 182-196 (2013)
6. Schuler, B. T., Gardeb, S., Heard, S., and Beale, T.: Towards automatically generating graphical user interfaces from openEHR archetypes, in *Ubiquity: Technologies for Better Health in Aging Societies: Proceedings of MIE2006*, Vol. 124, IOS Press, pp. 221 (2006)
7. Atalag, K., and Yang, H. Y.: From openEHR Domain Models to Advanced User Interfaces: A Case Study in Endoscopy, in *Health Informatics New Zealand Conference* (2010)
8. Beale, T., and Heard, S.: The openEHR Archetype Model: Archetype Definition Language, ADL 1.4, *The openEHR release*, 1 (2007)