Author's response to reviews

Title: Archetype Relational Mapping - A practical openEHR Persistence Solution

Authors:

li wang (daowangli@icloud.com)
Lingtong Min (ericmin1987@zju.edu.cn)
Xudong Lu (lvxd@zju.edu.cn)
Huilong Duan (duanhl@zju.edu.cn)

Version: 6 Date: 12 August 2015

Author's response to reviews: see over
Dear Editors:

Thanks for your review comments. The comments are very important for us to improve our paper, the revised parts according to the comments as follows:

1. We cite the CJBE paper in the background section and provide a short explanation of how this manuscript extends the CJBE paper as follows: “This work extends the basic ARM method introduced in a previous proof-of-concept work by providing a more sophisticated mapping approach from templates and archetypes definition to map archetypes to relational tables and details about performance optimization rules.”

2. The title of the paper is revised from “Archetype Relational Mapping - An openEHR Persistence Solution” to “Archetype Relational Mapping - A practical openEHR Persistence Solution”.

Reviewer 1:

Major Compulsory Revisions

1. The background section is focused on openEHR, since the paper focuses on this specification. However, I miss referencing also other dual-model based approaches such as ISO 13606 or HL7 CDA.

Thanks for this review comment. Introduce to similar works and approaches on ISO 13606 and HL7 will make this paper much more complete. We add related data persistence works on ISO 13606 and HL7 in the background section. “There are also similar works on data persistence of other two-level modelling based approaches such as EN13606 or HL7 Version 3. In a proof-of-concept work of EN13606, the data storage is developed by applying the Object Relational Mapping (ORM) approach to the reference model of EN13606. We investigated this approach in an earlier work. The deep inheritance and complicated relationships of the EN13606 reference model causes a lot of JOIN operation during data query. And classes near the top hierarchy are heavily over loaded with data, for example, DATA_VALUE is the basic class of all data type classes and contains the common attributes of all the instances of all data type classes. It cannot operate in a real-time setting. In IBM Clinical Genomics medical research product, a hybrid data model is developed based on HL7 Version 3 Reference Information Model (RIM). The hybrid data model combines elements of ORM and EAV approaches which suits medical research data warehouse well where the data is sparse and flexible, but it also has similar problems as other ORM and EAV approach and does not improve the performance enough to support clinical transactions.”
2. In the background section, third paragraph, archetypes are described as modeling clinical concepts of the knowledge domain. However, archetypes rather describe clinical recording scenarios (lab test, diagnostic report, etc.). Clinical concepts are defined by ontologies/terminologies (e.g. cholesterol, heart attack, liver cancer, etc.). This is also repeated in the following paragraph where it is stated that specialists define clinical concepts with archetypes; and again within the archetype preparation section, where it is claimed that concepts are mapped to existing archetypes. Please, revise this issue in the whole document. If you consider that archetypes define clinical concepts, please clarify what you refer to by using clinical concept.

We agree that the phrase “clinical concepts” here contains certain ambiguity. Archetypes are used to model the information structures of the clinical concepts. We add “Archetypes define coherent, whole informational concepts from the clinical domain” in the third paragraph, and use “archetypes model the structure of clinical concepts” instead of “archetypes defines clinical concepts” in this paper.

3. In the database mapping subsection, in Table 1, what does “#” within a column mean? Please add the explanation within the corresponding text or table caption. In the same subsection, in 8) you write: “an alternative is to use the ontology of the data item”, I think that you mean to use the textual name or description provided within the archetype ontology section. Please correct that sentence.

Thanks for this review comment. We add explanation for “#” within the corresponding text: “For fields with non-preliminary types in each archetype basic data type, the corresponding SQL type is noted as “#” and the mappings should be looked up recursively in Table 1.” We think your comment is much more accurate and revise the sentence “an alternative is to use the ontology of the data item” as advised.

4. In the ARM mapping subsection you say that you need to create a new archetype, is it the lab_test_general? What is the difference with the existing lab_test? Do you have to make the existing archetypes specialisations of that one? Please, clarify the above issues within the text.

Yes, the openEHR-EHR-OBSERVATION.lab_test-general.v1 archetype is newly designed and specialized from archetype openEHR-EHR-OBSERVATION.lab_test.v1 with three additional multiple occurrence data items, namely Test item, Result, and Result unit. We add explanation within the text “One is newly designed: openEHR-EHR-OBSERVATION.lab_test-general.v1” and “Since the lab test result items all have a similar data structure, it is convenient to define a generalized archetype openEHR-EHR-OBSERVATION.lab_test-general.v1 specialized from archetype openEHR-EHR-OBSERVATION.lab_test.v1 with three additional multiple occurrence data items, namely Test item,
Result, and Result unit, after the Lab Test Data concept in IV (Figure 4), along with the specialized archetypes to represent these flexible lab test results.”.

5. At the end of the query benchmark subsection, where the performance of each database is explained it would be useful to know which tables are queried by each test query and not just how many (e.g. “In test 2, only one table in the ARM database is queried...

Thanks, it is much better to specify which tables are queried in each test query. We add further explanations to tables queried in each test query. “In test 1, both the ARM and IV database are queried with one SQL clause, using patient id as a condition. ... In test 2, table IRequestImagingExam in the ARM database is queried using one SQL clause, but in the IV database, 3 corresponding tables Imaging Exam Requester, Imaging Exam Filler, and Imaging Exam Item have to be queried with 3 SQL clauses. ... In test 3, two tables OImagingExam and OImagingExamImageDetails are queried in the ARM database and three tables Imaging Exam Filler, Imaging Exam Report, and Imaging Exam Image are queried in the IV database. ... In test 4, table IRequestLabTest in the ARM database is queried while two tables Lab Test Requester and Lab Test Filler need to be queried in the IV database. ... In test 5, both the ARM and IV databases are queried on one table with one SQL clause, using patient id as a condition. The OLabTestGeneral_patient_value column of the OLabTestGeneralStructureResult table in the ARM database is non-clustered indexed, while the Patient Identifier column of the Lab Test Data table in IV database is clustered indexed. ... In test 6, only one table is queried with different conditions, namely table OLabTestBloodGases in the ARM database, table Lab Test Data in the IV database, and table OLabTestBloodGases in the Node+Path database, so the increase of query time is trivial in all three databases. ... In test 7, the query time increases greatly for the ARM database and the Node+Path database, where more than one table is queried, namely tables OLabTestBloodGases, OLabTestFullBloodCount, OLabTestLiverFunction, OLabTestThyroid, and OLabTestUreaAndElectrolytes in both databases, but not for the IV database, where only table Lab Test Data is queried.”

6. Regarding figures and tables: Figure 1: Please add principal and foreign keys in order to understand how tables relate each other. I find Figures 3 and 4 confusing. The fact that the table has fewer rows makes it difficult to understand. Please, consider improving this representation. In Figures 5, 6 and 7, what do the asterisks mean? Please add the explanation to their caption. Please, explain also what the lines between the tables mean and why, for instance, the lines between the tables in Figure 1 are different from the ones in Figure 6, since both schemas correspond to the same IV database? What do the percentage mean in Figure 4? Please, add explanation to the table caption. The last figure, the screenshot of the application, is very small and as it is, is not very useful. Consider its change or removal.

Thanks for comments on the figures. We re-draw most of the figures and re-organize the content of each figure. In figure 1 and figure 2, we present the overview of IV database and archetypes. Then in
figure 3, 4, 5, and 6, we present the details of mappings between IV concepts, archetypes, and the arm tables. In figure 7, 8, and 9, we present the schema of the test IV database, arm database, and Node+Path database, on which the performance comparison is conducted. We think these figures are much clearer and can help the readers to understand this paper much better. In table 4, we add the explanation “the percentage values in IV and ARM columns are more time spent on each query in the slower database than the faster database” to the table caption. The screen-shot figure of the application is removed.

Minor Essential Revisions

1. Please, at the end of the background section refer to the following sections by their names and not by their numbers (Section 3, etc.), since they do not appear within the document.

Thanks, we changed the section numbers to their names. “First the ARM approach is introduced in detail, including archetype modelling, template definition, and mapping rules. Then the ARM is applied to the EHR data requirements of a tertiary hospital from China, and the performance comparison of the generated ARM database, the conventional database deployed in the hospital, and the openEHR official Node+Path database is analysed. After that challenges and issues encountered during ARM development and application are discussed, before conclusions are reached.”

2. In methods section, in the third paragraph, consider changing the definition of template “A template is a specification that defines a tree of one or more archetypes and each constraining instance of various reference model types...”. A template does not define any archetype but references or aggregates them and it further constrain them ...

We agree that “A template is a specification that defines a tree of one or more archetypes and each constraining instance of various reference model types...” is not exact, and we revise it into “A template is a specification that structures a tree of one or more archetypes and each constraining instance of various reference model types...”.

3. References: Reference 5: Architecture is mentioned twice in the title. Please, add the last accessed date to the web references

We revised the references section, and add last accessed date to reference 11 and 17.

Discretionary Revisions
1. In the subsection template definition, the addition of an example of such template with the ARM constraints would add more clarity to the paper. As it is now it is hard to understand how these constraints look like, especially the ones mentioned in 3).

Thanks for this comment. In the template definition subsection, we give an abstractive description of mapping between generalized archetypes and specified archetypes. We think it is better to give an example in the ARM mapping subsection. We add an example using specialized archetype openEHR-EHR-OBSERVATION.lab_test-full_blood_count.v1 and generalized archetype openEHR-EHR-OBSERVATION.lab_test-general.v1 in the ARM mapping subsection.

Reviewer 2:

The paper Archetype Relational Mapping – An openEHR Persistence Solution discusses an important subject: development of a solution that will map openEHR archetypes to a relational database. This solution can facilitate the development of archetype based systems and will bring the adoption of the openEHR specification onto a new level. The paper provides very detailed description of the solution, however, I think that the structure of the paper is suboptimal. Especially the methods and results sections. I would advise reorganizing them, as the result section? Especially the very beginning of the section, contains much that should be in the methods section.

Thanks, the paragraph at the beginning of the results section is more closely belonged to the methods section. We moved this paragraph to the methods section under the sub section title “Database comparison”.

In the methods section, we would like to put the ARM approach in a more general way, so we can focus on the introduction of the ARM mapping rules and be brief on archetype modelling which is well investigated in many works. Since the ARM approach contains multi steps from archetype modelling to database mapping, so we think it is better to put all the outputs from each step in the results section, where readers can bear in mind the whole ARM approach after going through the methods section.

The other point of improvement for this paper would be a comparison of the provided solution with a state of the art mapping solutions if there are any. And also the benefits of the solution should be better emphasized.

We give an introduction on similar data persistence methods in the background section. “There are also similar works on data persistence of other two-level modelling based approaches such as
EN13606 or HL7 Version 3. In a proof-of-concept work of EN13606, the data storage is developed by applying the Object Relational Mapping (ORM) approach to the reference model of EN13606. We investigated this approach in an earlier work. The deep inheritance and complicated relationships of the EN13606 reference model causes a lot of JOIN operation during data query. And classes near the top hierarchy are heavily over loaded with data, for example, DATA_VALUE is the basic class of all data type classes and contains the common attributes of all the instances of all data type classes. It cannot operate in a real-time setting. In IBM Clinical Genomics medical research product, a hybrid data model is developed based on HL7 Version 3 Reference Information Model (RIM). The hybrid data model combines elements of ORM and EAV approaches which suits medical research data warehouse well where the data is sparse and flexible, but it also has similar problems as other ORM and EAV approach and does not improve the performance enough to support clinical transactions.” These published data persistence methods are not tested for clinical workflow and needs further optimization. This is also the main reason for us to develop our ARM method. We hope the ARM method can facilitate the adoption of openEHR approach.

Also please improve the language. There some phrases that are hard to read.

Thank for this comment. We try to use short sentences to be more accurate and use the same words to be more coherent.

Major Compulsory Revisions

Line 37: Healthcare data is too complicated, flexible, and changeable to capture a universal, comprehensive and stable schema of information, which is the foundation of the whole EHR architecture.

You say that the openEHR approach provides means to avoid this. Could you please describe in more details how your solution tackles this problem?

Thanks for this review comment. From our experience, adapt to the change of information requirements is one of the most important feature for healthcare information systems to increase flexibility and lower costs. openEHR provides a solution to build future-proof healthcare information system using archetypes to model the changeable healthcare information and make the information system driven by archetypes. The openEHR approach covers many aspects of a healthcare information system, such as data model, data persistence, data access, data exchange, and user interface. Our solution is focused on the data persistence and tries to provide a high performance data persistence
method for openEHR approach. When information requirements change, clinicians adjust or define new archetypes to adapt to these changes. Then the data storage can be generated automatically using the mapping rules from archetypes and templates. The openEHR approach brings dynamic, rapidly evolving development style to healthcare information systems. It is not necessary to wait until all the information requirements become stable, the requirement change management is built into the healthcare information system development process. Our method follows this fundamental principal of openEHR and facilitates the healthcare information systems to adapt to the change of information requirements.

Line 131. An underpinning principle of openEHR is the use of archetypes and templates. Please provide more information about templates and their role in the openEHR specification. You talk about this later. But I think you should talk more about them in the introduction

Thanks for this review comment. In the background section, we focus on archetypes and the introduction of templates is not sufficient, it is a bit sudden when we come to templates in the method section. We moved “A template is a specification that structures a tree of one or more archetypes and each constraining instance of various reference model types, such as Composition, Section, Entry subtypes, and so on. Templates usually correspond closely to screen forms, printed reports, and in general, complete application-level lumps of information to be captured or sent; they may therefore be used to define message content” from the method section to the background section, and add “Archetypes are deployed at runtime via templates that specify particular groups of archetypes to use for a particular purpose, often corresponding to a screen form” to the third paragraph to give an introduction to all the three artifacts of the openEHR specification, which are reference model, archetype, and template.

Line 179 Define mappings between generalized archetypes and specialized archetypes to facilitate the data...How do you define the mapping between archetypes? I think your paper is missing a figure where you can show all the archetypes and their relationships. You have a DB scheme. I think a figure that will show the archetype structure and corresponding DB data fields will improve the paper.

Thanks for this review comment. We re-draw most of the figures and re-organize the content of each figure. In figure 1 and figure 2, we present the overview of IV database and archetypes. Then in figure 3, 4, 5, and 6, we present the details of mappings between IV concepts, archetypes, and the arm tables. In figure 7, 8, and 9, we present the schema of the test IV database, arm database, and Node+Path database, on which the performance comparison is conducted. We think these figures are much clearer and can help the readers to understand this paper much better.
For EHR systems, it is impossible to define so many concepts once, and hard to maintain if new concepts are coming out continuously. Can you quantify the problem? How many archetypes require a revision a year for example in your clinic? You also state in discussion that your method faces a versioning issues. Could you describe a possible solution?

Thanks, we encountered a lot of changes of requirements during the development of many healthcare information systems. But it is quite difficult to quantify the revision of archetypes. We can try our best to describe the situation and context drive us to seek solutions like openEHR. For example, during the development of an EMR system, at first the information requirements are focused on the order centered work flow, we design archetypes to represent the information structure of patient, medication order, symptoms, operations, examinations, lab tests, and so on. But as the EMR system has been deployed and used by clinicians from different specialties, new information requirements and structured information requirements emerges gradually. For example, during the screening of Alzheimer’s Disease, clinicians use a lot of scales. Only a single total score of the scales leaves a lot of information of the patient out of the information system and clinicians want to record all the answers. As new scales are developed continuously, there are already six scales in the EMR system for AD screening. Then some experts think that there are too many scales for an AD patient who is always aged and try to extract a subset of questions which can be easy to use and even improve the accuracy of the screening. Then new scales are developed, deployed and evaluated in the EMR system. We encounter this kind of requirements from many different specialties. A lot of clinicians need to record as much structured information as possible and raise lots of changes to the information requirements. We seek approaches like openEHR to adjust to changes with archetypes.

In discussion section, we discussed the situation of change data type of data item of new version in detail. Change data type of columns is one of the incompatible data table operations, which is also incompatible in archetypes. In the ARM approach, we provide two solutions for this situation. One is to map the new version of archetype to a new data table, and import the data from the old data table to the new data table, using the data covert algorithm defined in the new version archetype to covert the incompatible data item. This can be carried out automatically with all the information defined in the archetype. The other is to map each version of archetype to a data table. Then there is no more versioning issue, but the number of data tables will increase. As old version archetypes become outdated, the corresponding data tables can be moved to back up database.

Then the data stored in the generalized archetype can be queried with specialized archetypes, and vice versa, using the mappings. I think you don’t store data in archetypes. It is more about archetype instances. Please reformulate this.
Thanks, the archetype is like the data structure, and the data is stored as archetype instances. We revised it as “Then the data stored as generalized archetype instances can be queried with specialized archetypes, and vice versa, using the mappings.”

**Line 239. To explore the performance of the ARM approach... I think that you’d rather placed this in the methods section**

Thanks, it is more closely belonged to the methods section. We moved this paragraph to the methods section under the sub section title “Database comparison”.

**Minor essential revisions**

**Line 151. Archetype preparation. I think you should use another term, like modeling, or definition.**

Thanks, archetype modeling is better. We changed all “archetype preparation” to “archetype modeling”.

**Line 175. Collection data types such as CLUSTER, ITEM_TREE, ITEM_LIST, and archetype slots. I would not call these elements or reference model “data types”. Data structures is a better term**

Yes, data structure is better than data type here. We changed all “collection data types” to “collection data structures”.

We are very grateful for your warm work earnestly. In all, we found the comments are quite helpful. They point the deficiencies about our manuscript, also the aspects that we have not done enough, and helps us for further improvement.

Thank you and best regards.

Yours sincerely,

Li Wang

**Corresponding author:**

Name: Xudong Lu

E-mail: lvxd@zju.edu.cn