Pre-print, revised version resubmitted to Medinfo 2019 – The 17th World Congress of Medical and Health Informatics, 25-30 August 2019, Lyon, France
Published version available in proceedings (pages 1781-1782) at http://dx.doi.org/10.3233/SHTI190645

Configuration of input forms in EHR systems using spreadsheets, openEHR archetypes and templates

Erik Sundvall[a,b], Annika Terner[c], Helen Broberg[d], Carrick Gillespie[e]

[a]Department of Biomedical Engineering, Linköping University, Linköping, Sweden, [b]Region Östergötland, Linköping, Sweden
[c]Department of Public and Caring Sciences, Uppsala University, Uppsala, Sweden
[d] Department of Digitalization and IT, Region Skåne, Lund, Sweden
[e] Omniq AB, Stockholm, Sweden

Abstract

Differences in structure and semantics of data captured using screen forms in different Electronic Health Record (EHR) products and configurations is the root cause of many interoperability problems. We present a workaround enabling reuse of openEHR archetype and template semantics to configure forms in four surveyed, insufficiently standardized, EHR-products used in Sweden (Cerner Melior, Cerner Millenium, Cambio Cosmic and CGM TakeCare). Data from EHRs was then exported and queried using standardized query mechanisms.

Keywords:

EHR, semantic interoperability, openEHR, archetypes, forms

Introduction and background

EHR systems provide both hardcoded non-configurable structures and other customer configurable parts. A significant part of EHR configuration is the creation of custom input forms, sometimes called templates, which consist of headings. However, the predefined headings are only to a small extent shared between the professional groups [1] within a healthcare provider or shared between providers [2]. This variation makes it hard or impossible to share structured information in an interoperable way. If the information is too different then it cannot be safely converted/mapped to structured exchange formats. Instead unstructured versions of exchange formats or telefax-transmission of printouts are often used, combined with time consuming reinterpretation and restructuring of the information by (already too busy) healthcare staff in order to fit into the receiving organization's IT-system and routines.

Standards targeted for EHR content standardization such as openEHR, HL7 CIMI, EN ISO 13606 combined with EN ISO 13940 can, if wisely used, contribute to supportsimilar data capture at the source (data entry) making conversion processes unnecessary, simpler, or at least possible. In Sweden the implemented support for such standards in present and recently acquired EHR systems is limited. Customer-configurable forms do however exist.

The idea of using openEHR archetypes to configure input forms in a proprietary system by making limited modifications to the EHR System software has been described by Chen et.al. [3], and would likely have been useful if it had been implemented beyond research pilot. The work described here builds on similar ideas, but only uses (the more limited) mechanisms available in the studied current systems without vendor modifications of the software.

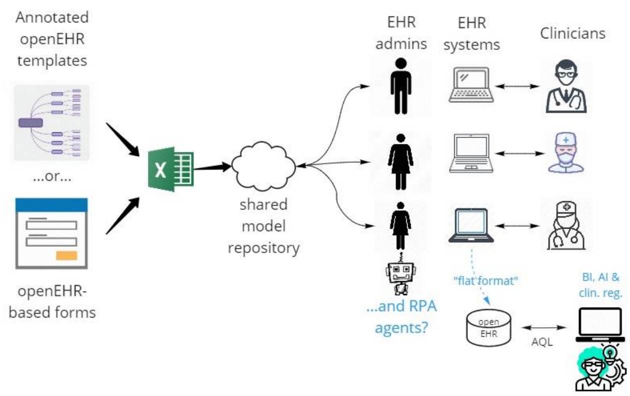

Figure 1 – Usage context of suggested workaround

Several regions have signaled interest in using shared archetypes and openEHR-templates as building blocks when creating and maintaining configurable EHR input forms also in non-openEHR systems. That is the path explored here.

Methods

We asked what methods different healthcare regions employ when formalizing requirements and configuring EHR systems. Excel spreadsheets or documents with tables were received from six of the 21 Swedish healthcare regions. A new combined spreadsheet was compiled to fit all studied vendors' main datatypes for configurable input forms, see *Table 1*. A smoking-related spreadsheet example can be found in the "ehr-form-config" directory of https://github.com/modellbibliotek/standin/

A detailed description of openEHR-based modelling is out of scope, but a brief summary is that an openEHR template for a specific use-case can be constructed by joining and configuring/constraining and combining several (possibly repeated) archetypes (and terminology subsets/lists). Any node in such a template has a unique archetype based "path" and can also carry arbitrary "annotations" (key+value-pairs). We used annotations to mark which template nodes to expose in non-openEHR systems' forms and at what "indentation" level. The annotations could also allow automation of the (now manually executed) algorithm to convert from annotated template to the suggested spreadsheet format or ideally directly to control configuration APIs or interfaces. As part of the algorithm the paths of the used nodes are also written into corresponding spreadsheet rows (to enable later storage in the EHR system's or EHR-data export toolchain's data or metadata). The ID of the openEHR-template is also written into the spreadsheet.

Table 1 – A simplified view of the main datatypes found to be used for configuration in the four studied systems. In addition to the described datatypes the spreadsheet file, where applicable, contains descriptive help text intended to be easily available upon user request in form systems. Also columns relating the rows to openEHR templates and "paths" based on openEHR-archetypes were added to support openEHR based post-processing and potential querying of data. HL7 FHIR mapping guidance can also be manually added to the spreadsheet if available. A header level "depth" column is used for systems that support headings in hierarchical levels.

Datatype/ System	Free text	Date & Time	Numeric value	Calculation	Selection list	Coded term	Link
Cambio Cosmic	Yes	Yes, §F	Yes, §F, 1…2	Yes, §F	Yes, §F, +N, 2	Yes, 1…*	Yes
Cerner Melior	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Cerner Millenium	Yes	Yes	Yes, +F	Yes	Yes, +F, 1S	Yes, 1...*	Yes
CGM TakeCare	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Abbreviation legend: +F = Free text field for comments can be added (configurable); §F = Free text field is always available for comment (not possible to turn off in configuration); +N = Numeric value can be added to the respective choices and can be used for calculation, e.g. for assessment instruments.; 2 = The number of choices in the selection list can be delimited to two choices by single selection, e.g. when using checkboxes for 'Yes' and 'No', respectively; 1...* = One to several code systems can be made available, e.g. both ICD-10 and ATC.; 1...2 = One or two numeric values can be specified in the same datatype. E.g. for blood pressure, two numerical values can be specified; 1S = If a special choice (in an otherwise multiple-choice type of) selection list is selected, then no more choices from the list can be made, e.g. if you choose 'No diseases' in a selection list that in addition to the choice 'No diseases' contains a selection of diseases.

Results

The combined spreadsheet format has seven different main datatypes, see Table 1. *Free text* implies narrative text without any additional structure. *Date & Time* (possibly partial) dates and times. *Numeric value* includes the value itself, unit, maximum value, minimum value and reference values. For *calculation*, a formula is given, combined with unit, maximum value, minimum value and reference values of the calculated value. For the *selection list*, the choices available to be selected are defined and whether single or multiple choice is allowed. *Coded term* carries codes that can be numeric or alphanumeric, and a meaning/display label of each code in a plain text. *Link* references other information, within or outside the EHR system

The spreadsheet method is suggested to be used as a way to "translate" annotated openEHR templates (or forms) based on archetypes to a form that resembles what current EHR system administrators are familiar with and using today. The configuration of the templates, in current EHR systems that lack an import mechanism, then has to be made manually based on the spreadsheets (as usually done today) or potentially assisted by RPA (Robotic Process Automation) tool scripts.

If/when openEHR modelling tools become more familiar in the regions and more automation is desired, then the spreadsheet step could be skipped and either annotated openEHR templates or form definitions could be used for configuration information. Templates were currently chosen instead of the form definition format already used by a couple of openEHR products, since there is not yet an official openEHR standard for that format. When standardized, it could potentially be used instead of (or as a complement to) annotated templates.

Figure 1 illustrates the workflow; templates of shared interest are collaboratively created and annotated (using normal openEHR toolchains) to fit use-cases and then algorithmically converted to the suggested spreadsheet format and uploaded to a shared repository (e.g. on GitHub) When the EHR administrators configure forms based on the spreadsheet they make sure to also carry over information regarding which openEHR template-ID the form is based on into the EHR. Information about which archetype-based path each spreadsheet row is associated with also needs to be stored in the EHR configuration or some other place that export mechanisms later can use to re-associate the EHR data with the path. Exact association mechanism may vary between systems, shorter Dewey-compressed [4] or hashed paths an also be used instead of full paths in EHR data.

The data in the (non-openEHR) EHR systems will not need to contain the full openEHR tree structures. As long as the openEHR-template-IDs and paths can be reconstructed during data export then e.g. an openEHR "flat" JSON format (https://github.com/ethercis/ethercis/blob/master/doc/flat%20json.md) can be used as import mechanism into an openEHR-compliant system and for example be queried using AQL [5]. The whole approach was tested by using an openEHR-template-derived spreadsheet to configure a CGM TakeCare EHR system. Export mappings to openEHR based "flat format" were then easily created and used to export data to an openEHR based open source EtherCIS EHR. The data (originating from the TakeCare system) could then be queried in the EtherCIS system using AQL (Archetype Query Language).

Conclusion: The described "spreadsheet" approach will of course not make existing non-standardized EHR content any easier to share in a structured way (no matter if it is user configured content or non-configurable content designed by the vendor). Organizations with an existing operational EHR installation will likely not suddenly switch all configurable EHR-entry forms from non-standardized to standardized, but can do it incrementally combined with other process and information change management and maintenance work.

References

[1] Terner, A., Lindstedt, H., & Sonnander, K. (2012). Predefined headings in a multiprofessional record system. J Am Med Inform Assoc 2012;19:1032-1038.

[2] Halilovic, A., & Terner, A. (2018). The Extent of Shared Headings in Electronic Health Records in Swedish County Councils and Regions. Journal of Medical Imaging and Health Informatics. Vol. 8, 1–5, 2018

[3] R Chen, GO Klein, E Sundvall, D Karlsson and H Åhlfeldt, Archetype-based conversion of EHR content models: pilot experience with a regional EHR system, *BMC medical informatics and decision making 9 (1), 33*

[4] E Sundvall, F Wei-Kleiner, SM Freire and P Lambrix, Querying archetype-based electronic health records using Hadoop and Dewey encoding of openEHR models, *Studies in health technology and informatics 235, 406-410*

[5] Archetype Query Language (AQL) specification https://www.openehr.org/releases/QUERY/latest/docs/AQL/AQL.html

Address for correspondence
Erik Sundvall, erik.sundvall@liu.se