

# Formal Structure for Specifying the Content and Quality of the Electronic Health Record

H. Dominic Covvey<sup>1</sup>, David Zitner<sup>2</sup>, Daniel M. Berry<sup>1</sup>, Donald D. Cowan<sup>1</sup>, Michael Shepherd<sup>2</sup>

<sup>1</sup>University of Waterloo  
Waterloo, ON, N2L 3G1  
Canada

{dcovvey,dberry,dcowan}@uwaterloo.ca

<sup>2</sup>Dalhousie University<sup>2</sup>  
Halifax, NS, B3H 1W5  
Canada

{david.zitner,michael.shepherd}@dal.ca

## Abstract

*This paper outlines a systematic approach to defining, eliciting, and specifying the structure and the information content of the Electronic Health Record.*

**Keywords:** Electronic Data Record, requirements elicitation, requirements specification, type system

## 1. Introduction and Current Situation

For any Information Technology (IT) system, among the information that must be conveyed in a requirements specification for any application is the contents of the data elements that are input, stored, and output by the application. For future extensibility, this data should capture the information inherent in the portion of the real world that lies in the application's domain. Moreover, it should be possible to add new content without having to reinvent the language of the contents each time, thus avoiding an electronic Tower of Babel [5].

This need is particularly valid in a system in which there are many applications that share the same data, with the output of one application being the input to others. In this case, the data requirements specifications transcend any application and are truly system wide.

These needs are particularly valid in the Health Informatics (HI) domain, in which large numbers of applications are designed to work with patient health records. These health records must have the property that for any given patient, all records for the patient must have consistent data, that is,

for any patient  $p$ , for any attribute  $a$ , e.g., identity number, birth date, cholesterol count on a given date and time, the value of  $a$  in all patient records for  $p$  must be the same.

Consequently, each attempt to design an HI application assumes the existence of a uniform, standardized *Electronic Health Record* (EHR) called *the EHR* [2, 8]. However, since there is no such standard, the designers of each application go ahead and make what *they* believe *the EHR* should be.

Thus, most of the work proceeding today to develop the EHR is based essentially on opinion surveys, in which individual professionals are asked:

- to identify which data elements, e.g., gender, current medications, smoking history, etc. should be, in their opinion, included in the EHR,
- to review a list prepared as a straw contents list and to rank items for inclusion based on the opinion of the professional, or
- to provide a straw version of the proposed contents based on their own preferences, which will then be combined with other submissions and a consensus set selected.

Each of these methods is flawed because:

- there are no explicit agreed upon criteria that all suggesters can use to choose from among a large number of possibilities,
- it is impossible to compare the resulting lists on other than a frequency of suggestion basis, and
- it is not possible to rationally debate the inclusions and exclusions of any individual datum since each list is derived from a separate perspective.

We believe that a formal framework for the selection of data elements for the EHR will reduce the likelihood of a failed implementation [10]. This framework would provide universally accepted criteria to be used to elicit these data elements from health-care providers.

Note that this paper represents only a proposal. The proposal has not been validated in practice. Rather, the

requirement engineers among the authors noticed a problem in HI, to which requirement engineering methods can be used to put some order in the chaos. This report has as one of its goals to convince the HI community of the need for requirements engineering practices to solve some of its problems [13, 6]. The proposal will be validated only after enough of the HI community has accepted the proposal that the proposed method gets carried out. Then and only then will it be possible to report on the success or failure of the proposed methods.

## 2. Definitions

We carefully distinguish between a *data element* and a *data item*. A data element is a field name, and a data item is a value, and they are related as follows:

A data element is the name of a field of the EHR, whose name signifies to human health-care providers the type and purpose of the data items that can be put into the field. For any data element, at any time, its field contains only one data item.

A data item is often called a *datum*.

A subrecord of the EHR, comprised of specific data elements, is called a *view* of the EHR [20]. The list of data elements comprising the EHR or a view is called *the structure* of the EHR or view. The list of data items of the EHR or a view is called *the contents* of the EHR or view.

A view [17] of a patient's EHR is built to support a particular treatment of the patient and consists the data elements whose data items are needed by the treatment, either prior to, during, or after the treatment. Consequently, we call a view a *usage-supporting view*.

Within this general framework, there are two complementary approaches suggested for eliciting (1) the structure of the EHR and of its views and (2) the views themselves.

1. Determine a data element that belongs to the EHR and then assign it to all existing views that need it.
2. Determine a view, determine its data elements, and then add to the EHR those of the new elements that are not already in the EHR.

Observe that the description of these approaches suggests that data-element and view elicitation is a continual, unending process. As new treatments are discovered, new views and new data elements will be needed to provide the data items to carry out the new treatment.

The implications of this framework and of the implied continual requirements elicitation are:

1. The software using the EHR should be structured in way that continual structural change to the EHR and

its views and continual addition of views should not necessitate rewriting of any existing software. That is, adapting to the changes should be part of the functionality of the software.

2. The initial encounter with a patient should include filling an instance of the EHR for the patient and the selection of views that are useful for the patient. Perhaps, a form filling application needs to be developed to assist a patient in filling his or her EHR.
3. Each subsequent encounter with a patient should include updating the patient's EHR with new data items for previously filled data elements and data items for data elements introduced since the last encounter. The updating should include also development of new views for new treatments needed by the patient. Perhaps, a form updating application needs to be developed to assist a patient and his or her health-care providers in updating his or her EHR.

## 3. The Way Things Are Done Now

In many settings, the EHR is simply a collection of all information captured electronically and available in a provider-accessible form. The types of data in a collection like this are those data that happen to be captured and stored at any point in the health system. These data are stored for future retrieval either in a central location or in distributed locations. We would call this the *Default EHR*, a form that has several negative characteristics:

1. The Default EHR is not designed, i.e., its contents are determined by happenstance or by implementation considerations. Thus, the resulting EHR is merely an aggregation of data.
2. There is no guarantee that a datum needed for some purpose, such as an Emergency encounter, will even be in the Default EHR
3. Even if the needed datum were present in the Default EHR, there is no guarantee that it will be of suitable quality.
4. Even if the needed datum were present in the Default EHR, finding it may be a challenge; for example, it may be on any arbitrary page, many keystrokes or clicks away.
5. Finally, other information equally important for a particular purpose may not be associated with the first datum, but may be located elsewhere in the database.

Based on these observations, it is clear that a useful and quality-assurable EHR must be a structured collection of data in which usage-supporting views of the data actually facilitate multiple-purpose use [5]. These usage-supporting views are presentations of data based on defined

organizing principles, e.g., the set of data needed by a particular user for a particular use to achieve a particular effect, presented in a way that facilitates this use.

In summary, either we must define the EHR from scratch using concepts of data importance and relevance to determine its contents and views of these contents, or we must organize the Default EHR and address the missing data problem.

#### 4. Required Characteristics of Method

The method used to determine the EHR structure and the EHR views and their structure must have the following characteristics:

- It must define an EHR structure that is transportable, i.e., for widespread consensus and adoption, the application of the method must be insensitive to geographical location.
- It must be able to be applied without change to many different areas of health care, i.e., to allow a gradual buildup of content and the creation of an inclusive EHR, it must be specialty invariant.
- It must be able to be reapplied at any time as new areas of health care need to define new elements for the EHR, thereby ensuring that the EHR be a vital, growing entity.
- It must be understandable by the average health-care provider so as to be suitable for different levels of expertise and interest, in order to ensure the participation of all types of providers in all types of settings.
- It must provide insights into the quality level required for each data element.
- It must serve the purposes of a health record: the support of clinical care, health services administration, research, and patient and student teaching.
- It must support the requirements for one-time recording for multiple purposes.

#### 5. Proposed Structure Definition Method

We propose the application of two models for the definition of EHR content:

1. The User-type–Use-type (U–U) Model: This model recognizes that the EHR is intended to provide practical support to many types of care providers, to health services administrators, and to researchers. In this model, we first determine the evolving set of User-types, e.g., *Emergency Physician*, *Family Physician*. Then, for each User-type, we define the set of Use-types to which an instance of the User-type (i.e., a user of the User-type) will put the record [7].

Examples of Use-types include: for *Emergency Physician: Unconscious Patient Triage*; and for *Family Physician: First Encounter With New Patient*. Finally, given a specific Use-type for a particular User-type, define the data that are required to serve an instance of the User-type carrying out an instance of the Use-type. It is possible in the R-E model, defined below, to define levels of importance of the data by considering some Use-types to be common, others to be less frequent, and still others to be rare.

2. The Relevance to Effects (R-E) Model: The R-E model explores the effects for which a care process aims and the relevance of each data element in the EHR for being able to determine if the effects are needed or desired, to determine if the effects have been achieved, and to measure the magnitude of the effects. Each effect is defined as a set of impacts, outcomes, benefits, etc. In this case, data elements are incorporated into the EHR to allow the provider to determine if his or her intervention is needed or desired, if the intervention is likely to succeed, and eventually if the intervention has had an effect, what the effect is, and how successful the effect was. For example, for *Emergency Physician*, the relevant data are data that allow the determination of an unconscious patient's medication status, so that the physician can administer agents that are not contraindicated. The R-E model potentially allows for a statistical correlation between the existence of data elements and the measurability of the effects. It is possible also to rank the effects based on their order of importance to the patient.

We see the greatest value coming from the application of the combined models because different benefits derive from each, and together they yield a record definition better than that obtained by applying only one model.

We call the combined model the User-Uses-Effects (U-U-E) model, and propose its application as follows:

1. Define User-types: develop a list of User-types, ranked by priority, e.g., User-types from areas of health care that would benefit the most, or that are the most time-and-information critical in terms of patient outcomes. The list will evolve over time.
2. Define Use-types: for each User-type, define a list of Use-types, ranked by priority, e.g., common Use-types or important Use-types, in terms of their effects on patient outcomes. Each list of Use-types will evolve over time.
3. Define Target-Effect-types: for each Use-type of each User-type, define a targeted or desired effect, i.e., a set of impacts, outcomes, benefits, etc. A set of effects is an Effect-type, just as any other set of values can be considered a type.

4. Define Data Elements: for each Use-type of a particular User-type, identify those data elements, ranked by priority, e.g., absolute minimal required information, followed by nice-to-have information. It may be wise to specify the contribution of each data element to the particular Use-type.
5. Determine the Relevance of each Data Element to each targeted Effect. Analyze the contribution that each data element defined above makes to the existence, measurability, and magnitude of each expected effect of clinical intervention, thus allowing a final ranking of the elements.
6. Use the above ranking to define data quality effort levels. Knowing the importance of each item should allow tuning the data quality assurance process.

A number of steps call for ranking by priority. Some ordering is essential, because ultimately, it is necessary to have some order of proceeding through an encounter with a patient. In any circumstance, the context will determine the ordering appropriate for the circumstance.

Normally, it is difficult to get a person to be able to rank a list of items. It is even harder to get a group of people to be able to rank a list consistently. A useful tool is FocalPoint [14, 15]. Each person or group is asked to do pairwise ranking, and the tool interactively helps to build a consistent total ranking for all the elements of the list.

Instead of insisting on a total ranking, it might be useful to obtain a ranked list of equivalence classes, each item in a class being of the same priority as others in the same class. However, placing the items in a list of equivalence classes avoids the important question of determining the importance of items for a particular purpose. Moreover, having multiple views, each with its own ranking avoids forcing a single ranking on all situations.

## 6. Framework for Viewing Our Approach

It is advantageous to view the scheme we have proposed as follows:

- Consider the U-U-E Model as a 3-dimensional model having the User-types, the Use-types, and the Effect-types as 3 orthogonal axes.
- Each point in the space defined by the U-U-E Model is a view of the EHR defined by one User-type, with one Use-type, and one Effect-type. See Figure 1. A point in the space is defined as a triple:

$$(User\text{-}type_i, Use\text{-}type_j, Effect\text{-}type_k).$$

The data required are those data that allow a user of type  $User\text{-}type_i$ , carrying out a use of type  $Use\text{-}type_j$ , to achieve an effect of type  $Effect\text{-}type_k$ . The figure shows a typical point representing a family

physician's first encounter with a potential patient, leading to the potential patient's acceptance as one of the physician's patients. In this case,  $User\text{-}type_i$  is  $FamPhys$ ,  $Use\text{-}type_j$  is  $1stEnc$ , and  $Effect\text{-}type_k$  is  $PatAccept$ . The view for that point lists all data needed for accepting a patient by a family physician after the first encounter.

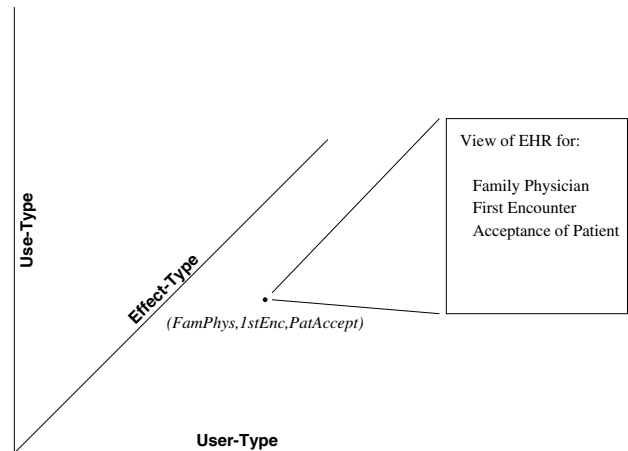


Figure 1: Views as Points in a 3-D Space

- In principle, every point or small region in this space can be occupied by a view. Moreover, methods and systems working with this space are or must be prepared to work with any such view.
- For example, object orientation and inheritance [3, 19] can be used to build 3 different extensible types of types, User-type, Use-type, and Effect-type, to allow defining new views in the space as innovations, e.g., new clinical interventions, give rise to the need for new views.\*
- While in principle, every point in the space could be occupied, in practice only a few are occupied.
- Experience has shown that the best way to identify useful and relevant points in the space is to proceed down the dimensions sequentially. That is, choose a particular User-type, then choose a particular Use-type for that User-type, and next choose the desired Effect-type. Finally, identify the dataset required for that point in space.
- One advantage of the 3-D view is that, if software is prepared to deal with any point in the space, it will be

\* Indeed, after this paper was accepted for publication, we became aware of a very recent publication by Portoni and Combi that proposed an object-oriented health-care database system providing user-oriented views [17].

impossible to define a situation in which the data are needed for which there is no point in the space.

- Also, while our method chooses dimensions in a particular order, it is not hard to conceive of problems in which choosing dimensions in another order makes sense. One example of another order would be to start with a known Use-type, then determine the User-type that executes this Use-type, and then determine the Effect-type and the data elements required.

## 7. Proposed Process for Defining Content

Assuming the method above, the following multiple-step Content Definition Process would be applied:

1. Establish a Content Development Oversight Group (CDOG). The purpose of this group would be to guide the overall process and ensure its progress. The composition of the group would cover the major areas of health care at a high level, and would include: a family physician, a specialist physician, e.g., a cardiologist, an orthopedist, or another specialist, a surgeon, a psycho-social care professional, a nurse practitioner, an HI expert, a health information management professional, etc. The objective is to have a reasonable cross-section of User-types, with a group size limited to about 15 [9, 18].
2. Engage the CDOG in a process to define the set of potential User-types of the EHR. It might be suggested to list these User-types in some order, e.g., in order of decreasing frequency, in order of increasing information dependence, etc. Record the criteria used to order the list. From time to time, the User-type list may be revisited to add additional User-types. In parallel with this, implement a database application to capture and manage the information collected in the process.
3. Define and convene, physically or virtually, a group of interested and motivated providers of each User-type, called a User-type group, for the first group of User-types to proceed with the charge. Between 5 and 10 representatives will be required in each User-type group. Prepare each User-type group for its role and the process each group must use.
4. Have each User-type group define the Use-types that its User-type will make of the record. Again, it may be advantageous to list these Use-types e.g., in order of decreasing frequency, in order of increasing information dependence, etc. Proceed systematically though the User-types as far as desired. Record the criteria used to order the lists.
5. Once each User-type group has identified a satisfactory number of Use-types, freeze the list of Use-types,

and proceed to identify the list of Required Data Elements (RDEs) need to satisfy each Use-type.

6. Once the list of Use-types and the content required for each Use-type are defined for the first set of User-types, apply the Relevance-to-Effects process. This involves first identifying the Effect-types expected from each Use-type. For each Effect-type it assesses the relevance of each RDE to determining if the desired Effect-type actually resulted, the measurability of the Effect-type, or the magnitude of the desired Effect-type. This assessment will provide
  - a. an ordering framework for the RDEs,
  - b. a means of determining the content of the first versus the successive pages or the prominence given to a given RDE within a view, and
  - c. a means of determining the level of quality assurance that will need to be applied to each RDE.
7. Subject each User-type, Use-type, Effect-type, and content set to consultation by other providers on a national basis.
8. Proceed to define additional User-types and Use-types and repeat the relevant steps of the process.

## 8. Technological Support for the Process

There are many technical and logistical approaches to carrying out the above steps. In order to control costs and minimize participant effort, the following approach is appropriate:

- Meetings of the CDOG may be convened using a virtual meeting support system, although the first meeting may be best accomplished in a physical meeting or via videoconferencing.
- The development of the Use-types list, and the RDEs list may be accomplished through a web-based Delphi process. The first virtual meeting of each User-type group will require the support of a document conferencing virtual meeting system.
- The Relevance-to-Effects portion of the process would best be performed off-line by analysts using techniques similar to those applied by InteRAI. (InteRAI is a non-profit organization "seeking to improve care of the elderly world-wide, by encouraging adoption of standardized assessment methods". See <http://nt8380.hrca.harvard.edu/Default.htm>.)
- Presentation of the products produced in the process and obtaining feedback from the broader provider community may be accomplished via e-mailed questionnaires.

## 9. Possible Drawbacks

A possible drawback to the approach is that the three-layer hierarchical approach of defining first User-types, then Use-types, and finally Effect-types may lead to a large number of similar Use-types and Effect-types, which will not easily be identified and merged. A solution to this problem is to regard the 3-dimensional U-U-E model as a 3-D data space in which each point in the space represents a view on the EHR. One can then apply OLAP techniques to take a slice along one dimension to form a subspace [12]. As an example, consider a slice formed by taking all points for which *User-type* is *Fam-Phys*. The slice results in a subspace that has only two dimensions, Use-type and Effect-type. The points in this subspace can be examined and those which are essentially the same can be identified. The (*Use-type*, *Effect-type*) pairs whose points have similar elements can be reviewed to determine if they can be merged on the grounds that they are really for the same User-type and result in the same view of the EHR. Similarly, the 3-D data space can be sliced along other dimensions, such as the Effect-type, to determine (*User-type*, *Use-type*) pairs whose points have the same Effect-type and thus result in the same view of the EHR.

Another possible drawback is that reordering and hiding data elements on such a fine-grained basis, for any point of the 3-D type space, can confuse a user who may log in to different points of the space and get slightly different views about the same patient. However, this potentially confusing situation must be compared to the default of showing everything every time and the user's having to wade through the entire list. Nevertheless, the effect of this potential confusion for the user needs thorough evaluation as EHR systems are being developed.

## 10. Conclusions

This paper has presented a scheme for specifying a uniform, but extensible EHR and for eliciting both the structure and contents of this EHR. The next step will be to carry out the proposed process with health-care providers to begin to put some order into health records and to allow specifications of families [16, 1, 4, 11] of interacting HI applications.

## Acknowledgments

The authors thank the anonymous reviewers of a previous version of this paper for their helpful comments.

D.M. Berry's work was supported in part by NSERC grant NSERC-RGPIN227055-00.

## References

- [1] Bass, L., Clements, P., Cohen, S., Northrop, L., and Withey, J., "Product Line Practice Workshop Report", CMU/SEI-97-TR-003, Software Engineering Institute, Carnegie Mellon University (1997).
- [2] Bates, D.W., Ebell, M., Gotlieb, E., Zapp, J., and Mullins, H.C., "A Proposal for Electronic Medical Records in U.S. Primary Care", *Journal of American Medical Informatics Association* **10**(1), pp. 1–10 (Jan/Feb 2003).
- [3] Booch, G., *Object Oriented Design*, Benjamin/Cummings, Redwood City, CA (1991).
- [4] Chastek, G., Donohoe, P., Kang, K.C., and Thiel, S., "Product Line Analysis: A Practical Introduction", CMU/SEI-2001-TR-001, Software Engineering Institute, Carnegie Mellon University (June 2001).
- [5] Chute, C.G., Cohn, S.P., and Campbell, J.R., "A Framework for Comprehensive Health Terminology Systems in the United States: Development Guidelines, Criteria for Selection, and Public Policy Implications", *Journal of the American Medical Informatics Association* **5**, pp. 503–510 (1998).
- [6] Cysneiros, L.M., "Requirements Engineering in the Health Care Domain", pp. 350–356 in *Proceedings of the IEEE Joint International Requirements Engineering Conference*, Essen, Germany (2002).
- [7] Description, CPRI Work Group on CPR, "Computer-based Patient Record Description of Content", Computer-based Patient Record Institute (May 1996), <http://www.cpri-host.org/resource/docs/content.html>.
- [8] Dolin, R.H., Altschuler, L., Beebe, C., Biron, P.V., Boyer, S.L., Essin, D., Kimber, E., Lincoln, T., and Mattison, J.E., "The HL7 Clinical Document Architecture", *Journal of American Medical Informatics Association* **8**(6), pp. 552–569 (Nov/Dec 2001).
- [9] Gause, D.C. and Weinberg, G.M., *Exploring Requirements: Quality Before Design*, Dorset House, New York, NY (1989).
- [10] Goddard, B.L., "Termination of a Contract to Implement an Enterprise Electronic Medical Record System", *Journal of the American Medical Informatics Association* **7**, pp. 564–568 (2000).
- [11] Gupta, N., Jagadeesan, L., Koutsofios, E., and Weiss, D., "Auditdraw: Generating Audits the FAST Way", pp. 188–197 in *Proceedings of the Third IEEE International Symposium on Requirement Engineering*, Annapolis, MD (1997).
- [12] Han, J. and Kamber, M., *Data Mining: Concepts and Techniques*, Morgan Kaufmann, San Francisco, CA (2001).
- [13] Higgins, S.A., de Laat, M., Gieles, P.M.C., and Geurts, E.M., "Managing Product Requirements for Medical IT Products", pp. 341–349 in *Proceedings of the IEEE Joint International Requirements Engineering Conference*, Essen, Germany (2002).

- [14] Karlsson, J., Olsson, S., and Ryan, K., "Improved Practical Support for Large-Scale Requirements Prioritizing", *Requirements Engineering* **2**(1), pp. 51–60 (1997).
- [15] Karlsson, J. and Ryan, K., "Prioritizing Requirements Using a Cost-Value Approach", *IEEE Software* **14**(5), pp. 67–74 (September/October 1997).
- [16] Parnas, D.L., "Designing Software for Ease of Extension and Contraction", *IEEE Transactions on Software Engineering* **SE-5**(2), pp. 128–138 (March 1979).
- [17] Portoni, L. and Combi, C., "User-Oriented Views in Health Care Information Systems", *IEEE Transactions on Biomedical Engineering* **49**(12), pp. 1387–1398 (December 2002).
- [18] Robertson, S. and Robertson, J., *Mastering the Requirements Process*, Addison-Wesley, Harlow, England (1999).
- [19] Schach, S.R., *Object-Oriented and Classical Software Engineering*, Fifth Edition, McGraw-Hill, New York, NY (2002).
- [20] Zeng, Q., Cimino, J.J., and Zhou, K.H., "Providing Concept-oriented Views for Clinical Data Using an Knowledge-based System", *Journal of American Medical Informatics Association* **9**(3), pp. 294–305 (May/Jun 2002).