Structured Data Entry in ORCA: the Strengths of two Models Combined

Astrid M. van Ginneken, M.D. Ph.D.
Department of medical Informatics
Erasmus University, Rotterdam, The Netherlands

The capture of patient data in a structured format receives increasing attention. Data can be extracted from free text using natural language processing techniques, but it can also be collected in a structured fashion at the time of data entry. The latter has the advantage that completeness and unambiguity can be promoted by offering predefined terms and options for description of findings. The paper discusses two models for supporting structured data entry. In the direct model, there is an immediate relationship between the terms and options for data entry and the structure of the underlying database. In the indirect model, terms and options for data entry are based on a controlled vocabulary and not directly related to the structure in which actual data is represented. Both models have been utilized by ORCA (Open Record for CAre). We discuss the pros and cons of these two models in relation to the type of patient data and the task involved. It is concluded that a strategic combination of both models has more strengths and less weaknesses than the use of each model only.

INTRODUCTION

The growing body of medical knowledge and complexity of patient care has made sharing of data among care providers more important and has increased the need for decision-support and access to reference knowledge. Other demands on patient data involve clinical research, epidemiologic studies, quality assessment, and health care management [1-4]. Free text may be the easiest way to collect and store, but its usability is very limited [5]. As soon as data need to be available for consultation and interpretation by a variety of people or systems, accessibility, standardization, completeness, and reduction of ambiguity become important. The collection and representation of patient data in a structured, coded or codable format, has been an important research focus for more than two decades. The efforts have produced a variety of strategies involving natural language processing (NLP) and structured data entry (SDE). NLP is used to extract coded data afterwards from free text [6,7], whereas SDE involves the use of a predefined structure and vocabulary at the time of data entry [8-14]. In parallel, there are efforts to arrive at standardization, which are reflected in the development of a variety of coding systems [15-17], controlled vocabularies, and representation schemes for data and knowledge exchange [19-22].

In this paper, we will focus on two models underlying SDE: the direct model and the indirect model. Both have strengths and weaknesses, but they are seldom found in combination and we will explain how a combination of the two can produce a very powerful mechanism for the collection and retrieval of structured, codable data.

THE PROS AND CONS OF TWO MODELS

The direct model

With the term 'direct' model we refer to applications, involving a direct mapping between items on the data entry screen and the attributes in the tables of a relational database. A direct model has several pros and cons. We will start with the positive aspects. Implementation can be done with most commercially available relational database systems [23] and once the data to be collected are known, modelling is straightforward. The collected data are transparent in the sense that tables and their contents can be browsed in a meaningful way. Furthermore, data can be extracted from the database in a variety of views using standard (SQL) queries. Hence, data exchange with other relational databases poses few technical problems.

Disadvantages involve maintenance and also include redundancy and lack of semantics. When both the domain of application and the number of users are limited, modelling and implementation do not require much effort. However, every change or expansion of the data entry set requires changes in the database, the interface, and the corresponding queries. In medicine, with its variety of specialties and physician preferences, a huge number of screens and tables would be necessary to tailor applications to their users. The effort of maintenance would exponentially grow with expansion of the domains to be covered. The problem of redundancy is a consequence of the fact that each database attribute has to be uniquely addressed. Because of the differences in focus, the internist will probably work with a different table for

0195—4210/96/$5.00

findings at physical examination than the neurologist. Some concepts, however will be present in both. A concept, such as blood pressure, may be present in several tables, possibly with different attribute names. A query involving all blood pressure measurements would require knowledge of all attributes in the database, representing that concept. Such semantics are not inherent to the relational model, but will have to be explicitly added and maintained by system experts. Another approach would be to define a variety of views on a set of tables, where each attribute is present only once. However, most databases only support queries on views and do not allow inserts and updates on views. Even if that functionality is present, the database structure will be far from optimal for individual applications and many tables would be very sparsely filled.

The indirect model

Indirect models are often used in applications with knowledge-driven data entry [24-28], where screen options are dynamically created on the basis of a controlled vocabulary and user input. Such a dynamic approach requires that database structure and content are not related. This can be achieved by storing findings as instantiations of concepts. A finding entered as '... - heart - murmur - phase - systolic' can be represented as a tree by a table with parent-child pairs: (obj12,obj13), (obj13,obj14), (obj14,obj15), where obj12 = 'heart', obj13 = 'murmur', etc. Any finding can be represented in this format.

The term 'indirect' denotes that the attributes in this table cannot be immediately interpreted, but require a mapping to their corresponding concepts to become meaningful. The indirect model also has pros and cons. Compared to the direct model, the pros and cons are essentially reversed.

Knowledge-driven data entry and the instantiation of concepts constitute a flexible strategy. The indirection allows for a powerful modular approach in which declarative and procedural knowledge are separated. Declarative knowledge applies to the contents of the knowledge base. Procedural knowledge applies to the algorithms that support data entry and retrieval using the contents of the knowledge base and user input. Likewise, an editor for knowledge base maintenance only involves procedural knowledge. Changes or expansion of the data entry set do not require changes to the database or interface architecture, but only require updating of the knowledge base. The model can be compared to a VCR where the recorder represents the procedural knowledge and the tape the declarative knowledge. Changing the tape requires no changes to the recorder. Hence, the indirect model is pre-eminently suited to tailor data entry to the needs of a specific specialty, department, or even user. Because of instantiation of concepts, there is no need for redundant concepts in the knowledge base. A non-redundant knowledge base explicitly represents the semantics that are not present in the direct model. A query involving all blood pressure measurements in a patient boils down to retrieving all objects that are instantiations of the concept blood pressure.

Retrieval in the indirect model is not as straightforward as in the direct model and consequently, patient data cannot be directly browsed in a meaningful way: all data have to mapped to their corresponding concepts. Explicit retrieval of complex findings may involve mapping of smaller or larger trees of objects. In a direct model, each query requires a specific query-script. In an indirect model, the database attributes are not directly accessible. A general query algorithm is required to retrieve the patient data. Yet, such retrieval software is procedural knowledge. Once the algorithm is available, it is independent of the contents of the knowledge base and the amount of data stored.

DISCUSSION

Even with insight in the overall strengths and weaknesses of the two models, the pros and cons do not weigh equally for every part of the patient record. The trade-off also depends on the task for which patient data are needed. The following paragraphs discuss aspects that need to be considered in more detail to take advantage of the best of both worlds.

Which model for what data?

As explained above, the most essential characteristic of the indirect model is its flexibility. Hence, the model should only be used when the advantage of flexibility outweighs the disadvantage with respect to retrieval. There are several categories of patient data. They can be divided in categories that vary highly per specialty and those that are very similar among most specialties. Depending on specialty and personal preferences, notes on history and physical examination may cover different topics in varying detail. A cardiologist may omit to check the reflexes, whereas a neurologist examines them all in detail. On the other hand, it is likely that a cardiac murmur is only mentioned by a neurologist and described with all its aspects by the cardiologist. Similar examples may be given for certain test results, such as ultrasound, MRI, and X-rays. In other words, the attributes needed to describe the findings are variable. Data entry for these categories can best be supported with the indirect

model. Other categories, such as laboratory test results, medication, and diagnosis, do not show much variation in the sense that the attributes needed to describe them are not domain-dependent. A drug prescription involves the name of the drug, the dosage, route of administration, and frequency of intake, irrespective of the kind of drug. In the same way, laboratory test results involve the name of the test, the value, the normal range, and possibly a unit of measure. Data entry of medication, laboratory test results, and diagnoses can best be supported by the direct model, taking advantage of the transparency of the model and the straightforward retrieval.

Efficiency of data entry.

It is often expressed that flexibility is achieved at the expense of efficiency, because flexibility requires generality. Efficiency of data entry is highly determined by the way a certain model is used in an actual implementation. When the direct model would be used to build a large number of screens, covering all possible findings for all specialties in detail, then data entry and retrieval would involve navigation through many of these screens and become cumbersome. On the other hand, data entry via a well designed screen, tailored to a specific task, will be much more efficient than navigation through a large menu tree, based on an indirect model.

Hence, the question should be reformulated as: Which model has the largest potential for efficient data entry? Although the direct model permits the development of highly dedicated interfaces for data entry, such interfaces place very high demands on maintenance. It is questionable whether these demands can be met when scaling up such an application. It is certainly true that navigation through a large number of menus is by far not optimal for data entry. Yet, using an indirect model does not preclude the use of screens, tailored to a specific task. Such screens can be based on a view of the knowledge base, where each item on such a screen provides immediate access to the proper position in the knowledge base as if the author had been using a menu tree.

However, there are additional advantages of using an indirect model for data entry. First, the physician has freedom to choose the degree of detail in which he wishes to express his findings. This is important because it allows physicians to gradually migrate from using free text to a more structured style of reporting. Furthermore, the degree of detail needed may vary according to the situation: description of a major or minor complaint, daily routine or a clinical study. The second advantage is that a physician can enter findings beyond the scope of his own dedicated screens when he is confronted with observations outside his domain of expertise. Third, data entry can be made more efficient by using physician-specific descriptions of statements which' meaning is not trivial. A typical example are statements of the type 'X=normal'. The meaning of such statements may vary among physicians. However, the meaning per individual physician is fairly constant and related to his personal style of history taking and physical examination. The physician-specific descriptions can be acquired and updated interactively and re-used for substitution whenever applicable. Since a routine history and physical examination often produce many normal findings, a considerable gain can be achieved, both in time and in quality of information.

Efficiency of retrieval.

Speed of retrieval is closely related to the size and structure of the implemented database, and the complexity of the query. This applies to both models. The main concern is whether retrieval in the indirect model is acceptable, since it is less straightforward than in the direct model. In this respect, it is important to distinguish two types of queries, each of which has different consequences for speed in the indirect model. One type of query involves the data belonging to one patient, the other type involves data from groups of patients. Patient care mostly involves retrieval of data within one patient. This is fairly straightforward as only one object tree or needs to be traversed and displayed. This type of retrieval is fast, which fits with the time-pressure that is often present in patient care.

When a query involves a number of criteria for a group of patients, matching may become more complex and time-consuming than in the direct model. However, such queries are primarily done in a research setting, where time is not as critical as during office hours. The fact that a particular query in a direct model may be faster than its equivalent in an indirect model, does not mean that 'direct' retrieval is simply preferable over 'indirect retrieval'. Direct retrieval requires knowledge of SQL and thorough insight into the database to be queried. When technical people need to offer assistance, retrieval time may be in the order of hours to days. The advantage of an indirect model is that the knowledge base can be used to support the end-user in defining his queries interactively with immediate feedback.

Maintenance and consensus.

In the context of research, it is essential that all participants agree on the data set to be collected. However, daily routine in patient care requires room

for departmental and personal preferences. With the direct model, user satisfaction requires as many implementations as there are different needs. For practical reasons, it is desirable that such needs are formulated at the level of a specialty or department. Hence, someone in that department will be responsible to achieve consensus on data items and screen layouts. This is a difficult and continuous task. Apart from maintenance effort, the result will be a compromise for most of the users involved.
With the indirect model, there is still an expert needed to define the contents of the knowledge base. However, there need not be as much concern for achieving consensus. The contents of the knowledge base can cover all data items that a variety of physicians want. This permits physicians to enter extra detail in a structured fashion. Views can be defined to enforce adherence to minimal departmental requirements. Data entry for routine and scientific purposes can easily be combined.

Adding semantics to the patient data

In his foundations for the electronic patient record, Rector proposed two levels of information [29]. Level one includes all information that stems from what physicians observed, thought, and did. Descriptions of findings and medical interventions belong to this first level. Level two provides the links between the components at the first level to make explicit how they fit into the line of reasoning and decision-making. In the direct model, the finest possible granularity for the links is at the level of a set of findings, corresponding to a row in a database table. This is sufficient to express for example that drug A is given because of lab result B. However, when many items pertaining to the physical examination are represented in one table, it is only possible to make explicit that "a chest X-ray was requested because of the findings at physical examination". It is not possible to state that the X-ray was requested because of the complaints "cough" and "fever". In the indirect model, the finest granularity is not a fixed set of items, but any sequence of instantiations, representing one or more findings. Although there may not be many occasions in which such granularity is needed or relevant, the option may be useful for understanding treatment in complex cases, for research, and automated decision-support.

CONCLUSION

We have explained pros and cons of two models for SDE: the direct model and the indirect model. The two models are complementary in many aspects: their pros and cons are virtually reversed, as can be seen in Table 1. The main trade-off between the two involves flexibility and efficiency. The balance between flexibility and efficiency varies with the type of patient data involved. When flexibility is important, SDE can best be supported with the indirect model. This applies to patient data that vary greatly per specialty, such as history and physical examination. Other categories, such as laboratory test results, medication, and diagnoses, can best be reported using a direct model. The patient record, now called ORCA (Open Record for CAre), combines the strengths of both models into a powerful strategy with much gain in flexibility, efficiency, and maintenance [30]. Even more, the remaining disadvantages regarding retrieval of the combined model are less than those of each model when applied alone. We are rapidly approaching an era in which the capture of structured data, directly by the physician, will no longer remain an unattainable goal.

Table 1. Summary of pros and cons of both models

	Direct model	Indirect model

Pros	- Direct browsing of data tables
	- Direct retrieval
	- Data exchange
		- No redundancy
		- Semantics
		- Flexible in:
		  - data entry
		  - data retrieval
		  - expansion
		  - scaling up

Cons	- Redundancy
	- No semantics
	- Less flexible in:
	  - data entry
	  - data retrieval
	  - expansion
	  - maintenance
	- Effort scaling up
		- No direct browsing of data
		- Special retrieval algorithm required
		- Data exchange not straightforward

References

1. Dick RS, Steen EB. The computer-based patient record: an essential technology for health care. Committee on Improving the Patient record Division of Health Care Services. Institute of Medicine. National Academy press 1991.
2. McDonald CJ, Tierney WM. Computer-stored medical records: Their future role in Medical practice. JAMA 1988;259:3433-40
3. Shortliffe EH, Tang PC. Patient records and computers. Ann Intern Med 1991;115:979-81
4. Reiser J. The Clinical Record in Medicine. Part 2: Reforming Content and Purpose. Ann Intern Med 1991;114:980-85.
5. Cimino JJ. Data storage and knowledge representation for clinical workstations. Int J Biomed Comput 1994;34:185-94.

6. Baud RH, Rassinoux AM, Scherrer JR. Natural Language Processing and Semantical Representation of Medical Texts. Meth Inform Med 1992;31:117-26.
7. Satomura Y, Do Amaral MB. Automated diagnostic indexing by natural language processing. Med Inf 1992;17:149-63.
8. Trace D, Naeymi-Rad F, Haines D, et al. Intelligent Medical Record-entry (IMR-E). J Med Syst 1993;17:139-51.
9. Lussier YA, Maksud M, Desruisseaux B, Yale PP, St-Arneault R. PureMD: a Computerized Patient Record software for direct data entry by physicians using a keyboard-free pen-based portable computer. Proc Annu Symp Comput Appl Med Care 1992:261-4.
10. Gouveia-Oliveira A, Salgado NC. A unified approach to the design of clinical reporting systems. Meth Inform Med 1994;33:479-87.
11. Bernauer J. Conceptual graphs as an operational model for descriptive findings. In: Clayton PD, ed. *Proceedings of the 15th SCAMC*. New York: McGraw-Hill 1991:214-8.
12. Poon AD, Fagan LM. The design and evaluation of a pen-based computer system for structured data entry. In: Ozbolt JG, ed. *Proceedings of the 18th SCAMC*. Special issue of JAMIA 1994:447-51.
13. Moorman PW, van Ginneken AM, Siersema PD, et al. Evaluation of Reporting Based on Descriptional Knowledge. JAMIA 1995;2:365-373.
14. Cimino JJ, Clayton PD, Hripsack G, Johnson SB. Knowledge-based approaches to the maintenance of a large controlled medical terminology. JAMIA 1994;1:35-50.
15. United States Center for Health Statistics. Int. Classification of Diseases, ninth revision, with clinical modifications. Washington. D.C. 1980.
16. Côte AR. Architecture of SNOMED. In: Orthner HF, Blum BI, eds. *Proceedings of the 10th SCAMC*. Springer-Verlag 1986:167-79.
17. NHS Centre for Coding and Classification. Read Code File Structure Version 3: Overview and Technical Description. Woodgate, Leicestershire, U.K. 1993.
18. Campbell JR, Payne TH. A comparison of four schemes for codification of problem lists. In: Ozbolt JG, ed. *Proceedings of the 18th SCAMC*. Special issue of JAMIA 1994:201-5.
19. Cimino JJ. Data storage and knowledge representation for clinical workstations. Int J Biomed Comput 1994;34:185-94.
20. Nowlan WA, Rector AL, Rush TW, Solomon WD. From terminology to terminology servers. In: Ozbolt JG, ed. *Proceedings of the 18th SCAMC*. Special issue of JAMIA 1994:150-4.
21. Rector AL, Nowlan WA, Glowinski A. Goals for concept representation in the GALEN project. In: Safran C, ed. *Proceedings of the 17th SCAMC*. New York: McGraw-Hill 1993:414-8.
22. Masarie FE, Miller RA, Bouhaddu O, et al. An interlingua for electronic interchange of medical information: using frames to map between clinical vocabularies. Comput Biomed Res 1991;24:379-400.
23. Elmasri R,Navathe SB. Fundamentals of Database Systems. 1994. The Benjamin/ Cummings Publishing Company Inc, Redwood City, CA.
24. Moorman PW, van Ginneken AM, van der Lei J, van Bemmel JH. A model for structured data entry based on explicit descriptional knowledge. Meth Inform Med 1994;33:454-63.
25. Rector AL, Kay S. Descriptive models for medical records and data interchange. In: Barber B, Cao D, Qin D, Wagner G, eds. *Proceedings of MEDINFO 89*. Amsterdam: North-Holland 1989:230-4.
26. Dolin RH. Modeling in relational complexities of symptoms. Meth Inform Med 1994;33:448-53.
27. Campbell KE, Musen MA. Creation of a systematic domain for medical care: the need for a comprehensive patient-description vocabulary. In: Lun KC, et al, eds. Proceedings of MEDINFO '92. 1992:1437-42.
28. Bell DS, Greenes RA, Doubilet P. Form-based clinical input from a structured vocabulary: Initial application in ultrasound reporting. In: Frisse ME, ed. Proceedings of the 15th SCAMC. McGraw-Hill, 1992:789-90.
29. Rector AL, Nowlan WA, Kay S. Foundations for an electronic medical record. Meth Inform Med 1991;30:179-86.
30. van Ginneken AM, Stam H, Moorman PW. A multi-strategy approach for medical records of specialists. In: Greenes RA, Peterson HE, Protti DJ. Proceedings of MEDINFO '95, Vancouver B.C., IMIA 1995.

Address of correspondence:
Dept. of medical Informatics, P.O. Box 1738
3000 DR Rotterdam, The Netherlands
E-mail: vanginneken@mi.fgg.eur.nl