An Object Oriented Computer-Based Patient Record Reference Model

Laurent Doré, Marion Lavril, François-Christophe Jean and Patrice Degoulet
Medical Informatics Department, Broussais University Hospital, Paris, France.

In the context of health care information systems based on client/server architecture, we address the problem of a common Computer-based Patient Record (CPR). We define it as a collection of faithful observations about patients care, with respect to the free expression of physicians. This CPR model supports several views of the medical data, in order to provide applications with a comprehensive and standardized access to distributed patient data. Finally, we validated our CPR approach as a primary data model server for an application for hypertensive patient management.

INTRODUCTION

As reported by the Institute of Medicine (IOM)[1.], the current trends in Hospital Information System (HIS), or more generally in any kind of health care information systems, are toward open and client-server system approaches[2.3.4.]. The IOM committee also identifies the computer-based patient record (CPR) as the core of such systems, regardless of the centralized or distributed nature of the record, so that all patient data, collected from departmental services (e.g., clinical or administrative services) may be shared among all the care providers. This need for a common patient data repository is particularly crucial for patients requiring shared care, where adverse effects may arise from inadequate coordination of activities and communication[5.]. Besides direct improvements in health care delivery, CPR systems may reduce the costs of health care (e.g., by eliminating redundant tests), improve practitioners productivity (e.g., by reducing the time needed to access relevant information) and enhance the outcomes research programs[1.].

The CPR is the reference model for the information concerning a single patient collected through applications during interactions between care providers and the patient[1.]. Along with Rector et al. we think that the CPR should be faithful, permanent and structured[6.]. The role of the CPR in the care process (i.e., in the applications) requires that all data relating to the patient be available to authorized persons when needed (e.g., at the time and place of patient care). This implies that any such reference model of patient data must be able to receive and transmit understandable data with internal or external systems involved in patient care, such as clinical, administrative, or research systems. It requires at least electronic data interchange using standardized message formats allowing the transfer of structured data[7.1.], and a common medical vocabulary for mutual understanding[1.].

In this study, we stress the medical (as opposed to the administrative) content of the CPR. We start from the nature of the CPR content (i.e., the medical data) to determine both the common aspects of the medical data structures that collect the data elements in the departmental applications, and the structure of the CPR model itself. We then present the functionalities of any such CPR system within a distributed HIS modelling approach based on the HELIOS client/server architecture[18.].

THE CPR MODEL REQUIREMENTS

In our opinion, the CPR model must be as general as possible and not dedicated to any specific needs such as those implicitly managed by Weed's problem orientation. We believe its organization should be based primarily on the common characteristics of the medical data it contains, whatever the original application.

Shared characteristics of the medical data

The medical data referenced in the CPR are specific occurrences of medical concepts; for instance, the systolic blood pressure of a patient is the 'association' of the medical concept «Systolic Blood Pressure» with the value 180 mmHg at a given time. By definition, a medical data element is characterized first by a *medical concept*, referring to a knowledge network as defined by the CEN[10.], that gives the data element its medical meaning. The second inherent characteristic of the medical data are being strongly time-oriented. The medical data's temporal tag is twofold[11.]. First, the *occurrence time* is the time when the data happened (e.g., the date/time of a myocardial infarction), and second, its *duration*. On the other hand, the recur-

0195–4210/95/$5.00 © 1995 AMIA, Inc.

rence of an event or the chronic status of a disease is specific to the medical knowledge and may be used later on to distinguish between the identity of several data. An unfortunate patient may break his left leg twice but will only once have a coronary insufficiency that we may observe several times in his life.

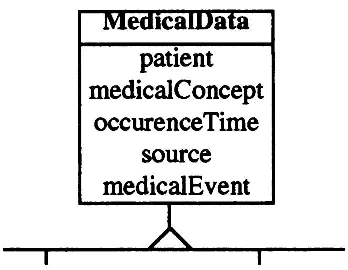
Furthermore, a data element is always the result of a particular *observation*. An observation is made by a source (i.e., a human observer or a machine), during a medical event (i.e, at a particular time in a particular place). This contextual information, as defined in the Pragmatic Data Model, is the required signature that defines the identity of the medical data for a better understanding of the record[12]. It facilitates data (re)interpretation and can deal with multiple or contradictory data elements. The *observation time* is the only time stamp that is always available. Since the occurrence time of a data precedes (or is sometimes equal to) the observation time, we can always at least infer its position when it is missing.

DESIGN OUTCOMES

We used these general characteristics of medical data elements for the design and implementation of the medical data structures and for those of the HELIOS CPR model.

Medical data structures design

Thanks to the Object paradigm and the inheritance property, the common features of the medical data can be placed in an abstract class (i.e., *MedicalData*). The medical data structures that collect data in each departmental application are its subclasses and inherit these common features.



The *MedicalData* subclasses design is specific to what the clinicians want to say and record in the target application. Indeed, there is no need for these classes to be general or relevant for other applications since their common features required for our CPR model are supported by the superclass MedicalData.

The CPR reference model

We agree to say that the CPR should be as independent as possible from the data manipulation (e.g., aggregation, comparison, inference) performed in dedicated applications for patient care, ward organization, epidemiology, or any other purpose. The CPR is a «primary product» defined as a collection of the data structures defined above[13]. This collection has been organized in a specific structure convenient for primary data manipulation: storage and retrieval. We have previously identified features shared by all data that we considered as the main axes of the CPR organization. We provide a *time-oriented* view using the occurrence time, an *encounter-oriented* view using the observation time, a *source-oriented* view according to the data origin, and a *concept-oriented* view using the medical concept. The last view is an extension of Weed's problem-oriented view[9] since we believe all concepts merit an index entry.

For this purpose we built two different subclasses of the class Dictionary[1]: a TimeDictionary (*TD*) and a LabelDictionary (*LD*). *TDs* use dates as keys. We developed a temporal knowledge representation to take into account duration, interval and point in time (i.e., date). A date can be known with a certain accuracy like in the LIED[14] or TOMR systems[11], and expressed with different granularity levels. In practice, we use as temporal keys dates expressed in <year>, <year, month> or <year, month, day>. The *LDs* use as keys the label of the medical concept that characterizes the data.

In our CPR model, medical data are referenced in several data dictionaries (DD) that combine the previous ones.

- One structure orders medical data according to the observation time. First, a *TD* orders the encounters of the patient. Then, each encounter references a *LD* to organize the patient medical data of this event by concept.
- Another structure provides a chronological view of the data. A *LD* organizes the medical data around medical concept labels (e.g., pulse, hypertension). Then, for a given concept, the data are organized within a *TD* indexed on the occurrence time.

When new patient medical data is added to the system, it can be referenced by existing keys in the dictionaries, or induce the creation of new ones.

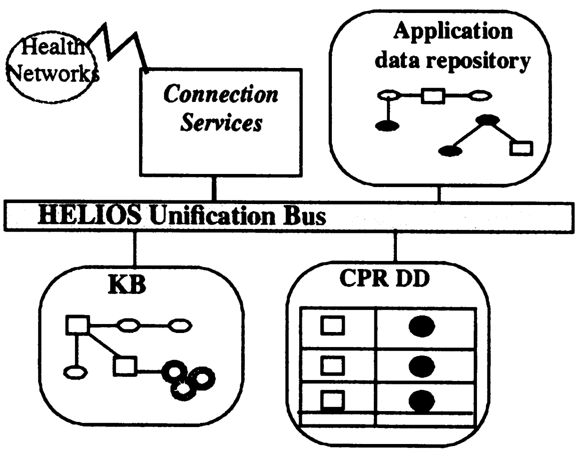
The values referenced in these different dictionaries are not directly objects (for example, encounter, medical data) but rather pointers to these objects. The use of pointers in the Object paradigm provides us with

1. Dictionary is a kernel-ordered collection of Object-oriented language that associates a key entry to a value.

several ways to access the same object for example, by its label, by a date.

EXPERIMENT & RESULTS

The HELIOS medical software engineering environment (SEE) is dedicated to the development of distributed multimedia medical applications, and based on the concepts of distribution and openness[18.]. We assumed that it was a relevant framework to test our object oriented CPR model as the core of a distributed health care information system like a HIS, since it allows the execution of application to be distributed across several computers. This section summarizes the required pieces of such a HIS modelling approach, that includes, beside the CPR itself, the data transfer medium and a common medical knowledge base. We then describe the access methods and the main functionalities of the CPR system we have already implemented. The clinical application ARTEMIS-2 currently under development plays the role of a data client/server. We specify how this application initially benefited from the CPR facilities.



Distributed CPR model

Data exchange

In a distributed approach, the CPR has to exchange information with the specific applications that run in each departmental service. It must collect the medical data acquired by these applications or provide them with previously acquired data, for example to check for a particular drug intolerance before validating a drug prescription.

Within HELIOS, the exchange of data between components hosted on heterogeneous machines linked by a network (e.g., the CPR and a remote application) is ensured by the Helios Unification Bus (HUB). The applications may be plugged directly on the bus or may communicate through the Connection Service (CS) component[15.] that is responsible for the easy integration of applications, developed within the HELIOS framework in broader clinical information systems. Thanks to the HUB toolbox[18.], a Component Communication Interface (CCI) is attached to each HELIOS component that allows it to be connected on the HUB and to code/decode data in the ASN.1 ISO presentation protocol, which is the internal HELIOS coding scheme. Furthermore, the CCI of the CS automaton already deals with ASTM, HL7, EDIFACT and ACR-Nema[15.], all standards that cover simple data exchanges such as ADT messages or lab results, and complex data, for example, images[16.]. In both cases, the distribution aspects are hidden to the users. The CPR and the applications communicate simply by sending/receiving messages through their own CCI.

The Knowledge Base

The Knowledge Base (KB) is implemented as a hierarchy of objects that allows all medical concepts to inherit from common structures. *Medical concepts* are described with their *labels* (those used in the *LabelDictionaries* of the CPR) and synonyms, sometimes in different languages, together with canned sentences useful to display and report concepts. To compensate for the ambiguity and contextual specificity of medical terms, we associate with each concept *definitions* in different languages and sources (Harrison, Dorland,...). Medical concepts are also described by their *codes* in several nomenclatures (CIM10, MESH, SNOMED, and others). To sum up, this KB supports a metathesaurus that links related terms and concepts from a variety of existing vocabularies and classifications. Furthermore, we provide declarative facilities to classify and organize concepts, using *is-a/subtype* and *contains/part-of* links. For instance, the blood pressure measurement is composed of the systolic blood pressure and the diastolic blood pressure. Those concepts and links, organized in a semantic network, represent the medical knowledge. We believe that this kind of highly flexible architecture, where physicians can update the library by adding synonyms, definitions, etc., or modifying links without major side effects in the system, can lead to a controlled vocabulary fitting smoothly into clinical practice. As reported by Gouveia-Oliveira, it can contribute to the acceptance of the system by physicians[17.].

The CPR system functionalities

The CPR system implemented relies on an Object Oriented Database Management System (OODBMS) that supports functions such as reliability, security, and, for example, only allows authorized persons to access the CPR contents. In our experiment, the data is exchanged between the CPR structure and the ARTEMIS-2 application through the HUB using standardized messages. Data is referenced in the CPR according to required information the application may provide; at least a medical concept label and an observation time. Access methods/messages allow the application to access the entire record in a flexible manner. Currently, these methods combine the following search criteria: encounter (i.e., a particular event) / source (i.e. a physician) / referenced concept's label / observation time / occurrence time. The temporal search can be performed on a time interval or relatively to a specific date (i.e., before or after it). Search by concept's label can also use the synonymy links of the concepts to enlarge the search process.

These searching primitives have been used for coding the two general functions (i.e., not application oriented) among those defined by the IOM[1.] that are currently supported by our CPR system: a patient health state description at a given date, and a chronological clinical evolution (e.g., those of the patient's hypertension). Other functions may be added later.

The CPR access methods have also been used to fulfill the specific needs of our ARTEMIS-2 test application, and the requirements of the medical knowledge consistency rules contained in our KB. These rules are triggered to validate the data entry for a patient. Indeed, they are specific behavior of the medical concepts referenced by the data, but they also need to access the patient medical context by querying the CPR to know about other data. For instance, during a drug prescription, the CPR is queried concerning the existence of an intolerance to this drug.

The ARTEMIS-2 application
Based on the ARTEMIS-1 experience[18.], ARTEMIS-2 is a specific multimedia prototype application that manages information about hypertensive patients (e.g., clinical and biological data, digital angiography images, medical reports) and that relies on the GemStone® OODBMS. In the context of a distributed HIS, applications are medical information subsystems, where medical data is acquired and manipulated according to the specific department needs. These subsystems may be located on another computer and have their own physical data storage. Thanks to the HELIOS architecture and its plugging facilities, it could be considered as another software component plugged on the HUB as illustrated by the figure «Distributed CPR model».

Its medical data structures have been made compliant with the design requirements presented above (see the section "Medical data structures design"). These classes express specific links between data. For example, the hypotensive drug is *prescribed for* the patient's hypertension and *causes* a particular after effect. Furthermore, negative or uncertain statements commonly found in the medical domain (e.g., the assumption of hypertension) are registered as particular values of the existence variable of the data. Our system includes a dedicated multi-valued logic to manage data uncertainty. A subset of the 60 000 patient records ARTEMIS-1 database that contains several hundred clinical data items per record, has been loaded using these new data structures.

Dedicated functions, such as reports or discharge summaries that need to query the CPR contents, have been rewritten with SmaltalkDB, the data language of GemStone®. Patient record manipulations are only performed through the public access methods of the CPR.

DISCUSSION

As reported by the IOM committee and despite the lack of consensus for the CPR content, no current system is capable of supporting the complete CPR, but those that most closely approximate the ideal CPR system share several common traits: comprehensive and flexible data dictionary, time tagged data, flexibility in reporting patient data and research tools for use with the CPR.[1.] The CPR model we have developed implements the first three features and provides us with a faithful, permanent and structured record.

- Faithfulness is enhanced since the CPR refers to both raw data or direct observations and to interpretations: all data elements collected in the different applications are effectively referenced. The arbitrary levels of detail and abstractions used by physicians (for example, arterial stenosis *vs.* stenosis of the right renal artery, a symptom *vs.* a diagnosis), the recording of uncertain or negative data, as well as that of conflicting data and multiple measurements (for example, several blood pressures measurements during a visit), are allowed.
- Permanence is achieved thanks to the temporal orientation of the CPR organization that makes the CPR a patient lifetime record. Since everything may be of importance[13,6.] once data is stored changes can be made only by authorized persons, those that have made the observation for example.
- Flexibility is provided by the CPR structure and allows fast and easy access to all data elements. The access facilities are intuitive and general (i.e., not application-oriented).

Research tools may be added easily either as particular applications or as specific functions of the CPR thanks to the access methods of the CPR and the fact that we collect raw data rather than summary interpretation.

Finally we have validated the CPR definition as an independent system for patient descriptions and confined the application as a set of functions and data manipulation using the CPR system. The access methods provided by the data organization in the CPR will facilitate the addition of intelligent capabilities such as decision support systems to manage medical alarms, for example. Nevertheless, a major enhancement is planned because of the imprecision of the occurrence time of most data and the resulting difficulties in managing them in the *TDs*. We will try to use the IxTeT algorithm of Ghallab[19] to order the medical data according to this time stamp. Finally, in the near future, the administrative data may come from the remote hospital administrative patient identification server, using the Connection Service facilities, providing us with a more complete CPR model.

Acknowledgments

This work received financial support from the Commission of European Union (ESSI project no 10271 and AIM contracts A1004 and A2015).

References

1. Institute of Medicine, Committee on Improving the Patient Record. The Computer-Based Patient Record: An Essential Technology for Health Care. Dick RS, Steen EB (eds). Washington, DC: National Academy Press; 1991.
2. Van Mulligen EM, Timmers T, Van Bemmel JH. A new architecture for integration of heterogeneous software components. *Methods Inf. Medicine* 1993;32:292-301.
3. Chueh HC, Barnett GO. Client-server, distributed database strategies in a health-care record system for a homeless population. *JAMIA* 1994;1:186-98.
4. Huff SM, Haug PJ, Stevens LE, Dupont RC, Pryor TA. HELP the next generation: a new client-server architecture. *18th Annual SCAMC.* 1994; 271-75.
5. Fletcher RH, O'Malley MS, Fletcher SW, Earp JAL, Alexander JP. Measuring the continuity and coordination of medical care in a system involving multiple providers. *Medical Care* 1984; 22;403-1.
6. Rector AL,Nowlan WA,Kay S. Foundations for an Electronic Medical Record. *Methods Inf Med* 1991; 30:179-86.
7. Branger PJ, van't Hooft A, Duisterhout JS, van der Lei J. A standardized message for supporting shared care. *Proc. 18th SCAMC.* 1994; 473–77.
8. Engelmann U, Jean FC, Degoulet P (Eds). The HELIOS Software Engineering Environment. *Comp. Methods Programs Biomed.* 45, Suppl. 1994: S1-S152.
9. Weed LL. The Problem-Oriented Record as a basic tool in medical education, patient care and clinical research. *Ann Clin Res* 1971; 3: 131-4.
10. CEN TC251/PT003. Medical Informatics - Structure Of Concept Systems - Vocabulary (Project Team Mose). Brussels: CEN. 1993.
11. Pinciroli F, Combi C, Pozzi G. Object-orientated DBMS techniques for time-orientated medical record. *Med Inform* 1992; 17(4): 231-41.
12. Degoulet P, Jean FC. Pragmatic Database Modelling. *Proc. MEDINFO 1989*; Amsterdam: North-Holland 1989; pp. 1127-31.
13. Nygren E, Henriksson P. Reading the medical record. I. Analysis of physicians' ways of reading the medical record. *Methods Prog. Biomed.* 1992; 39: 13-25.
14. Degoulet P, Devriès C, Rioux P, et al. LIED: A Temporal Data Base Management System. In. *Proc. MEDINFO 86.* Amsterdam: North-Holland 1986; pp. 532-6.
15. Jean FC, Mascart JJ, Codaccioni A, Lavril M, Sauquet D, Degoulet P. Using a meta-model to build a Connection Service in a object-oriented medical application development environment. *Proc. 18th SCAMC.* 1994; 483–87.
16. Hammond WE. The role of standards in creating a health information infrastructure. *Int J Biomed Comp.* 1994;34(1-4):29-44.
17. Gouveia-Oliveira A,Salgado NC, Azevedo AP, Lopes L, Raposo VD, Almeida I, Galvão de Melo F. A Unified Approach to the Design of Clinical Reporting Systems. *Methods Inf Med* 1994; 33:479-87.
18. Degoulet P, Chatellier G, Devriès C, Lavril M, Ménard J. Computer assisted techniques for evaluation and treatment of hypertensive patients. *Am. J. Hypertension* 1990; 3: 156-63.
19. Ghallab M, Mounir Alaoui A. Managing efficiently temporal relations through indexed spanning Trees. *Proc. IJCAI 89.* 1989;1297-1303.