Information Management in the Electronic Medical Record

Philippe De Smedt, Thomas B. Fisk*
Software Technology Laboratory
HPL-97-88
September, 1997

Electronic Medical Record (EMR), information management, object-oriented architecture and communication, distributed databases, health care

The development of a comprehensive Electronic Medical Record (EMR) system poses a number of unique information management challenges. The Mayo Clinic and Hewlett-Packard are currently collaborating on a pilot of an EMR system, intended to provide fast, reliable, and integrated access to patient and medical data to large numbers of simultaneous users, in a way that fits naturally with the users' normal mode of operation. We describe some key characteristics of EMR systems, with a focus on the information management issues that they raise, and offer insights on how these can be effectively addressed. While the Mayo Clinic constitutes a very large medical environment, similar issues exist in smaller settings, especially when one considers that patient and medical data will increasingly be available and gathered from a potentially large number of distributed information sources.

*Mayo Clinic, Rochester, Minnesota

Published in *Database Programming and Design*, Issue on the "Universal Database", May 1997.

Information Management in the Electronic Medical Record

Philippe De Smedt[1] *and Thomas B. Fisk*[2]

Abstract

The development of a comprehensive Electronic Medical Record (EMR) system poses a number of unique information management challenges. The Mayo Clinic and Hewlett-Packard are currently collaborating on a pilot of an EMR system, intended to provide fast, reliable, and integrated access to patient and medical data to large numbers of simultaneous users, in a way that fits naturally with the users' normal mode of operation. We describe some key characteristics of EMR systems, with a focus on the information management issues that they raise, and offer insights on how these can be effectively addressed. While the Mayo Clinic constitutes a very large medical environment, similar issues exist in smaller settings, especially when one considers that patient and medical data will increasingly be available and gathered from a potentially large number of distributed information sources.

Introduction

The goals of an EMR system are two-fold: first and foremost, the intent is to improve the quality of healthcare by supplying providers with timely and accurate clinical information about the patient and the state-of-the-art in medicine. Second, and inseparable from the first, is the goal to render healthcare more cost-efficient by helping providers choose the right treatment options for a given condition, thereby eliminating wasteful spending, and by increasing provider productivity by reducing the time spent on administrative tasks.

To satisfy both goals, information must be available to providers at the point of care - which can be an exam room, doctor's office, laboratory, emergency room, intensive care unit, and so on - and be used to bring institutional decision making in line with clinical guidelines and treatment protocols.

Clearly, the applications required to support both paradigms have different properties:

- applications at the point of care are characterized by high interactivity and focus on an individual patient's condition, whereas

- applications that aid institutional decision making often require little user intervention, run for a longer time, and yield results applicable to a large strata of the patient domain.

---

[1] Hewlett-Packard Laboratories, Palo Alto, California – desmedt@hpl.hp.com
[2] Mayo Clinic, Rochester, Minnesota – fisk@mayo.edu

Thus the key to a successful EMR system is an infrastructure that allows both sets of applications to coexist.

In what follows, we describe how these different properties translate into information management requirements, and how we are addressing them. First, we will briefly explain some architectural concepts.

Architectural Concepts

Traditional database paradigms no longer apply in a domain as complex as healthcare. Rather, a full-fledged EMR system will have a large number of diverse information-management components, each tailored to optimally process specific pieces of information and communicate with other components. This architecture gives the applications a unified view of patient and medical data.

Each of these information-management components is an object in a comprehensive object-oriented component architecture [1]. In our case, components communicate through a CORBA-compliant communication infrastructure (middleware) [2]. An information-management component can be characterized by its interfaces that define functionality as viewed by the application-level components or possibly by other information-management components, and by the type of data the component manipulates.

This encapsulation of information-management capability lets application designers focus on application functionality and the manipulation of objects relevant to the medical domain (defined in a medical information model), while hiding the complexity of data access and manipulation. In addition, the middleware infrastructure encapsulates inter-component interactions; the use of a trading service obviates the need to know the physical location of information-management components. Furthermore, operational database issues are hidden from the applications, allowing the database to change without (in most cases) affecting the applications.

Information-management components dynamically bind themselves to other components requiring their services by registering interface instances - or "advertising" themselves - with a trading service. The trading service manages requests from client components and dynamically refers them to the most appropriate information-management component. The level of appropriateness is determined based on expected response time, expected throughput, topological proximity, and so on - essentially balancing the load across the information-management components that offer the requested service.

Figure 1 shows how components can be freely distributed - and replicated, if necessary - across the architecture based on these performance and availability considerations. It also shows the interactions among the application components, the trading service, and information-management components.

An information-management component can be very complex. In its simplest incarnation it can consist of a thin shell over a file system; more complicated implementations can involve large-volume relational databases that are distributed or replicated to enhance performance and availability. In either case, the application should be unaware of the underlying data management complexity. Regardless of complexity, however, these components should support the applications efficiently and reliably.

Now that we have defined the notion of information management components, we can focus on the requirements that they are to satisfy.

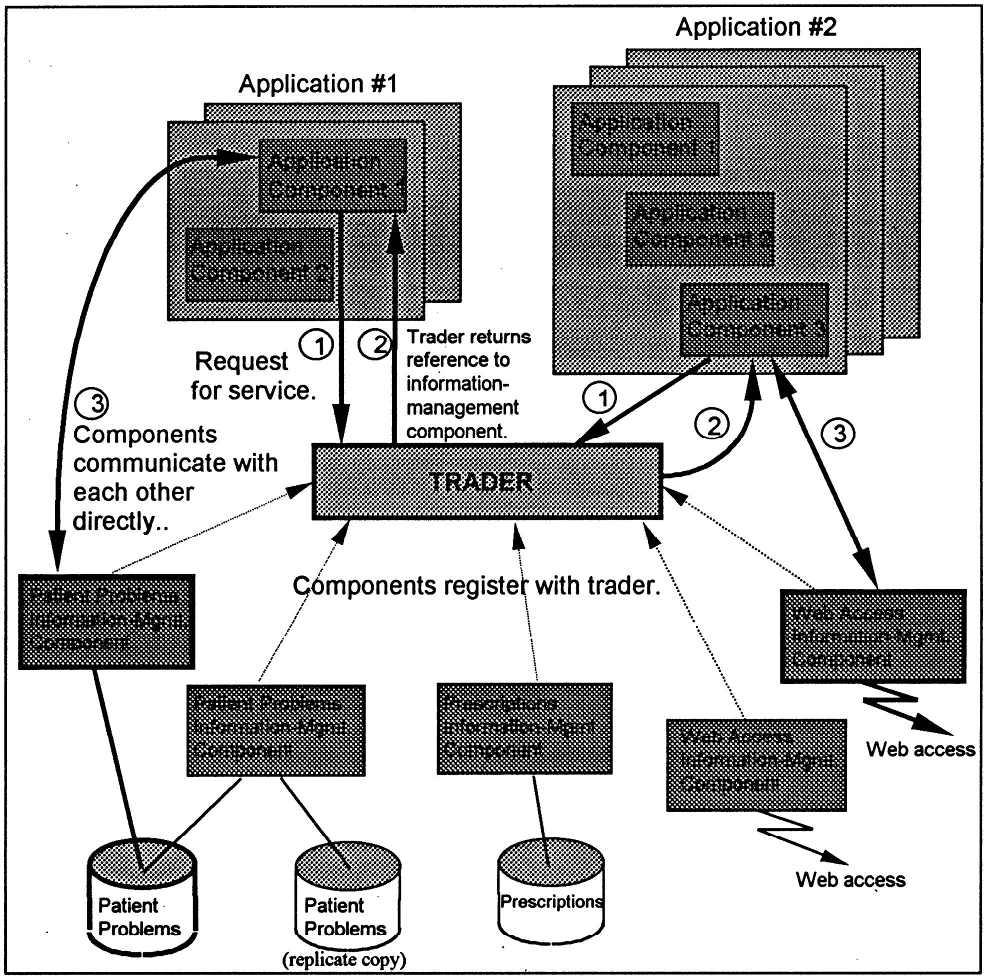

Figure 1. Example of interplay among application components, trading service, and information-management components.

Multiple Information Management Dimensions in EMR Systems

EMR systems have several dimensions, each with unique characteristics - yet only together do they provide a clear picture of the magnitude and difficulty of the challenge. The following list is not exhaustive; depending on the requirements of the application domain, some dimensions may be added or deleted. It is also not intended to be entirely orthogonal. The National Institute of Standards and Technology, through its Advanced Technology Program, has also identified its own information-management dimensions in healthcare [3].

Scale

The scale of the problem at hand is among the most important characteristics of an EMR system. This characteristic is defined by the large number of simultaneous users (likely to be in the thousands in a fully operational setting), the large number of accesses to data (both as look-ups and as new or modified entries into the EMR), and the sheer volume of patient medical records. It is estimated that the typical ASCII-based record totals a few hundred kilobytes to a few megabytes; if you include images, sound, waveforms, and so on, the record size increases by several orders of magnitude. For an organization the size of the Mayo Clinic, placing all existing records online will yield a data volume of many terabytes that should increase by 10 to 20GB per month, as more clinical functions are brought online. (The Mayo Clinic expects to deploy its EMR system on 10,000 to 12,000 workstations.) In addition, the emergence of Web-based medicine will further expand the scope of accessible information beyond what is currently available, facilitating information exchange among institutions.

Information-management components can help address the issue of scale. Such components can be distributed across the architecture to where they're needed, scoped to a specific granularity, and tailored to the particular tasks for the data they control - thus providing the flexibility to adapt to changes in scale.

Heterogeneity

Medical and patient data exists in many forms and is represented by various data models. For example, much of the Mayo Clinic's clinical information is stored in relational databases. As more clinical functions involving nontraditional (abstract) data are brought online and integrated into a comprehensive EMR system, other data models - structured, semi-structured, and unstructured - will apply as well. Each will have its own representation and methods of manipulation. Examples include imaging systems, clinical waveforms, text-based systems (clinical notes or medical literature), file systems, and object-oriented databases.

Information-management components hide heterogeneity from the applications by presenting an object-oriented view of the underlying data source at the component's interface. The role of the component in this capacity is to map between the application-managed object-oriented model and the data model supported by the data source. In many cases, such mapping - between relations and objects, for example - is nontrivial and requires custom-crafted conversion methods.

Legacy

As with most complex integration efforts, the existence of legacy applications and data sources poses several unique challenges.

Legacy systems, by definition, have been in operation for some time, and resistance to modifying them for integration into a new system can be strong. As a result, the integrating system must either rely on existing APIs or find a way to encapsulate the legacy system [4]. This encapsulation is one of the many tasks an information-management component may need to perform, and depending on the available interfaces, it can be very complex.

If the legacy is an application, the data is not directly accessible; if it's data, direct access is possible. It is almost always preferable, though not necessarily easier, to access data through the application, which may have certain semantic behavior that is not present in the data itself.

For example, consider the look-up of a patient's medical record. A flag may be set in the record that controls access based on certain privileges, and the flag is interpreted in the application. In this case, direct access to the data could be unauthorized. In hybrid situations, where some of the data logic resides with the data - as stored procedures in a relational database or methods in an object-oriented database, for example - accessing the data through those procedures or methods is imperative.

Performance

The acceptance and use of an EMR system are predicated on its ability to enable access to information without impeding normal user activities, and, most important, to do so quickly.

The Mayo Clinic has categorized response-time requirements based on several types of care, ranging from the emergency room to a scheduled visit. Generally, while under a provider's care, access to patient data should take less than a second within the same screen, and no more than two seconds for a transition between screens. Each care category imposes different requirements on the system. In the case of a scheduled visit, portions of the patient's record may be staged into a departmental server - allowing faster access and increased availability - as opposed to retrieving the data from the primary data sources at the time of the visit. During this staging process, which could occur at night, portions of patient records are loaded into the relevant departmental servers.

Similar policies can be based on institutional practice or other available information. The policies can be implemented through replication mechanisms that ensure the consistency between the primary record and the replicate copies. By localizing data access as much as possible, performance and availability improve and network traffic is reduced because only updates require primary server access. Another approach is to partition data sets across several database servers based on characteristics such as patient identifier and relevance of data to a particular hospital location.

The information-management components encapsulate the mechanisms for improving performance; the interface between the information-management and application components defines functionality. The location of data that needs to be manipulated to execute the function, and the mechanisms that access the data, should be entirely transparent to the application.

Availability

Medical information systems must be highly available - any disruption of access to patient data is intolerable.

As with performance, the mechanisms to ensure high data availability are encapsulated by the information-management components. The staging or replication mechanisms we discussed earlier can enhance availability by providing access to a local copy of the data. If the primary site were to become unavailable, the provider could continue with the locally available information.

Other traditional database mechanisms such as mirroring and hot stand-by can be applied as well. One of the tasks of the information-management components is to hide the intricacies of these mechanisms from the applications.

Application Characteristics

EMR-based applications can be divided into three categories: those operating on an individual patient record, those operating across patient records, and those providing access to general medical information. All are equally important in a successful EMR system implementation.

Because information-management components are specific for each information type, they must be constructed to support these different interaction paradigms. Their functionality, as specified in their interfaces, is to retrieve information for individual patients - a list of prescriptions, for example - and to integrate it with information for the same patient retrieved by other components. Furthermore, they must be able to retrieve information for a large set of patients and correlate it with other information for the same set of patients.

In the latter case, the more data mining functionality that can be integrated into the databases, the more efficient the effort will be. The more interesting data mining results, however, will most likely be associated with data residing in multiple data sources. In that case, additional integrative functionality may be required of the information-management components.

Access to general medical information may range from retrieving the hospital's list of authorized medications to accessing the medical literature or databases at other institutions. In these cases, information-management components that provide vocabulary or information-retrieval services or filtering can be applied.

Integration

One of the goals of the EMR system is to provide full integration of all available information sources at the visual user-interface level, as well as at the data level.

Although achieving visual integration is relatively straightforward through the intelligent sharing of screen real estate, data-level integration requires close interaction among the relevant information-management components. In some cases, these components may be controlled by multiple applications. This level of integration is necessary in a number of clinical applications; an example is an application for ordering medical tests. This module may need to interact with the patient problem application to establish relevance of the proposed test for a particular patient. For data mining applications, this kind of integration is crucial to enable, for example, the extraction of co-relational, statistical, or trend data over multiple sources of data.

Information-management components communicate with each other through their interfaces and are therefore well equipped to provide basic data-integration capabilities. Some more involved forms of information integration can only be performed at the application level; still others may be pushed down to the database level, obviously the most efficient method.

Events

Relevant activity on a patient's record must be propagated to interested parties as soon as possible. In many cases, a record may be modified by the provider while being accessed by a different provider accessing the same patient's record simultaneously, or through some other change to the underlying data, such as an incoming test result. To provide a consistent view of the record to all users, a mechanism that recognizes relevant events and propagates changes to all pertinent parties or applications is necessary.

The information-management components' role in this area is to generate events corresponding to lifecycle changes (modifications) to the patient record, deliver them to the event service (an integral part of the architecture), and respond to requests for service triggered by an event. An example of such a request would be to deliver an updated list of a patient's prescriptions to a provider who is not the initiator of the new prescription.

Transactions

Like other applications that access disparate information sources, EMR systems must do so in a transactional manner.

Information-management components rely on the transactional mechanisms provided by the information sources under their control. Although a large fraction of the transactions in an EMR system involve only one information-management component at a time, transactions may cross multiple information managers, thereby requiring transactional mechanisms that span these components. Therefore, an information-management component can participate in such a distributed transaction and signal that it's ready to commit if instructed by an overarching transaction mechanism. Any of a number of available distributed transaction monitors can provide such a mechanism.

Mobility and Disconnectedness

Access to medical data through mobile channels may become increasingly important. One possible manifestation of mobility is disconnected operation.

In some cases, mobility will imply disconnectedness. Providers may download relevant patient information to a hand-held device for review later in the day and be unconnected to the main data sources for some time.

Although the interaction paradigm gives the user great flexibility, it poses many challenges in the information-management domain. Disconnectedness implies that updates by the user to the patient record will only be local and will not be propagated to the primary data source or as events to other interested applications or users. It also implies that users may not have access to the most recent information about the patient's condition. Furthermore, a patient record may have been downloaded by multiple providers , and parts of the record may have been manipulated by more than one person, resulting in potentially conflicting modifications to the record.

The information-management components responsible for these pieces of information must be able to reconcile such disparities when the dowloaded records are reconciled with the master record, possibly necessitating human intervention. Disconnected operation is thus a mode of operation that must be carefully evaluated against these drawbacks [5]. However, mobility does not always imply disconnectedness: Continuous communication may be accomplished, for example, over secure radio channels.

In terms of information-management support inside the hand-held devices, lightweight memory-resident databases are not out of the question. As we discussed earlier, applications running in these devices generally do not need to know whether the data is local. It may be useful to inform users that the data that they are accessing is a local copy that may not contain the most up-to-date data.

Security

Confidentiality of information is critical in the healthcare domain - whether within an individual institution or across many institutions.

Because EMR systems have many different components, each component's security mechanism can be combined to ensure systemwide security and confidentiality. Databases, for example, have their own access-control mechanisms, the object-oriented middleware infrastructure may supply its own security mechanisms via encryption, and common-login facilities may be used to control access to applications. Although judicious use of these mechanisms goes a long way toward satisfying security requirements, a fully integrated security architecture - an area of continuing research - would be the ideal situation.

Organizational Requirements

Each organization or application domain has a set of information-management requirements that must be satisfied. Each requirement translates into additional functionality that the information-management components need to provide, or the information architecture that is put in place will be restricted. For example:

- *Data retention requirements.* Patient data may need to be retained for extended periods of time, requiring large amounts of storage and mechanisms for archiving data when it's no longer required online.
- *Access logging.* In addition to the requirement to prevent unauthorized access, all accesses to patient records may need to be logged.
- *Primary data location.* Although multiple copies of a patient's record - or a fraction of it - may exist, only one location is considered to contain the "true" record. Extreme care must be taken to ensure that this location contains the most up-to-date information as quickly as possible after it becomes available.
- *Data consistency requirements.* Mechanisms must be in place to ensure consistency among multiple copies of a record; as in any practical implementation of a distributed system, temporary inconsistencies may be unavoidable.
- *History of change.* Modifications to a patient's record must be traceable. Thus data records should be appended, not overwritten.

Conclusion

Developing a comprehensive Electronic Medical Record that satisfies the strict performance, availability, and decision-making requirements of the medical domain is a challenging task. With ever-increasing volumes of patient and medical information being generated and brought on-line, the possibility exists to develop systems that will truly have a positive and measurable impact on how health care is conducted in the next century. In this article, we've attempted to summarize some of the challenges that must be addressed in the information-management domain. Solutions for some of these challenges exist and are being implemented. Others, however, are still in the exploratory phase and require breakthrough thinking. A component-based architecture in which information-management components play a key role provides the power and flexibility required by EMR systems, which must adapt to the ever-changing needs of the medical domain.

Acknowledgements

We wish to thank our colleagues at Hewlett-Packard Laboratories and the Mayo Clinic for their continuing collaboration. The work presented here would also not be possible without the involvement and dedication of our colleagues in Hewlett-Packard's Medical Products Group.

References

[1] Seliger, R.A. "An Approach to Architecting Enterprise Solutions." *Hewlett-Packard Journal*, 48(1): February 1997.

[2] Moore, K.E. and E.R. Kirshenbaum. "Building Evolvable Systems: The ORBlite Project." *Hewlett-Packard Journal*, 48(1): February 1997.

[3] Advanced Technology Program. "Information Infrastructure for Healthcare." National Institute of Standards and Technology, 1997.

[4] De Smedt, P., J. Annevelink, T. Pham, and P. Strong. "A Physician's Workstation as an Application of Object-Oriented Technology in Healthcare." *Proceedings of the First International Conference on Applications of Databases*, ADB-94, Vadstena, Sweden. *Lecture Notes in Computer Science*, #819, Springer-Verlag, 1994.

[5] Forman, G. and J. Zahorjan. "The Challenges of Mobile Computing." *IEEE Computer*, 24(10): April 1994.