

A Flexible Approach for Electronic Medical Records Exchange

Vagelis Hristidis, Peter J. Clarke,
Nagarajan Prabakar, Yi Deng
School of Computing and Information Sciences
Florida International University
{vagelis, clarkep, prabu, deng}@cs.fiu.edu

Jeffrey A. White, Redmond P. Burke, MD
Division of Cardiovascular Surgery
Miami Children's Hospital
Jeffrey.White@mch.com,
redmond111@aol.com

ABSTRACT

Many methodologies have been proposed in the last decade for integration and exchange of medical data. However, little progress has occurred due to the following reasons. First, patients are reluctant to give full access to their historical medical data. Second, institutions are reluctant to open their systems to mediators or any type of external access, due to security, privacy (HIPAA, unique patient id) and competitive advantage-related reasons.

We propose a flexible mechanism for electronic medical record (EMR) exchange, which allows generating a customized EMR using a generic approach independent of existing healthcare applications. Furthermore, we provide an on-demand, secure, efficient, and semantics-agnostic way to exchange EMRs in a collaborative environment using our declarative communication engine, which we call Communication Virtual Machine (CVM). CVM negotiates the capabilities of the involved parties and underlying networks to guarantee Quality of Service and presentation compatibility. CVM can also be customized to enforce privacy and security requirements (e.g., HIPAA) by enabling logging, authentication, and so on. Finally, we provide a way to seamlessly integrate EMR generation and exchange between healthcare providers. A prototype of our EMR exchange approach has been implemented which integrates the i-Rounds[®] medical record system used at Miami Children's Hospital.

1. INTRODUCTION

The storage and exchange of patient medical records has been on the agenda of Federal and State organizations for decades. For instance, in November 2004 the Secretary of Health and Human Services released a report, which called for a National Health Information Network (NHIN) [OEM04]. However, there has been little progress for various reasons explained below. We identify these reasons and then propose a feasible approach.

To the eyes of a computer scientist with little or no knowledge of the sensitive issues in patient record exchange, the following model would seem ideal and feasible. Each patient has a unique patient id and all institutions' information systems are interconnected such that they all have access to the complete medical record of the patient since her birth. Today's technology clearly allows this integration model.

However, this model is not possible due to the following non-technological reasons. First, patients do not want to give to providers access to their whole medical record but only to the most recent part, since they believe that it would be unnecessary and intrusive [Wae03]. Second, a unique patient id is not possible

due to privacy-related and political reasons [Con06, Wae03]. Third, institutions are not willing to open their systems to external access due to security and competitive-advantage reasons.

Hence, we have to become less ambitious than the original vision of the integrated medical record. Waegemann [Wae03] suggests that only intra-institution sharing of medical records is feasible and we should focus on that. Then when a patient moves to another institution, she should carry her medical record or request that part of it is forwarded. Surprisingly, even the intra-institution integration of EMR systems is far from complete for most institutions today.

Along the same lines we also propose a non-intrusive approach where each institution maintains its own information system, and parts of or whole EMRs are exchanged on-demand in a semi-automated way which requires minimal human intervention. Our approach does not attempt to define the semantics of the exchanged data, but just provides a basic syntactic framework to allow exchange of complex data like a patient's medical record. The patient (or her family) is responsible to approve and initiate the EMR exchange.

A naïve approach to achieve patient-centric EMR exchange would be that when a patient leaves a hospital she is given her EMR in an electronic standardized format (e.g., on a CD or USB key), already provided by companies like CapMed [Cap06], and when she visits another hospital she will provide the desired parts of her EMR. This approach, although reasonable, is problematic because it is neither online nor on-demand and assigns to the patient more responsibility than she probably wants.

In a nutshell, our approach (presented in Section 3) works as follows. A medical mediator is deployed on each institution which, given a local patient identifier, generates a *Virtual Medical Record (VMR)*. A VMR is a description of the items in an EMR along with their actual locations. VMR and EMR are often used to denote the same entity in the rest of the text. This generated VMR does not have to follow any semantic rules but is built at the institution's discretion. Also, each institution runs a copy of the Communication Virtual Machine (CVM), which handles the on-demand, secure and efficient transfer and presentation of the VMRs. Notice that each institution builds its own authentication mechanism on their CVM for the patients to access their data remotely. The CVM seamlessly handles negotiation of presentation and device capabilities. For example, it does not make sense to transfer a Cardiac Echo to a healthcare provider whose connected device does not have the minimal resolution required for diagnostics to be performed on that Echo.

The inter- and intra-institution systems integration is outside the scope of this work. Notice that by inter-institution integration

we mean that there is a semantic mapping between the databases of the interconnected systems such that an EMR can be seamlessly imported from another system. We do not provide any semantic mappings but only the means to agnostically transfer an EMR between systems.

Our approach also addresses the following less critical yet important issues. It is desirable that the patient can request from an institution to transfer her EMR for a specific time duration. For example, we have found that patients typically prefer to only transfer to their current physician/hospital only part of their EMR, the part which they believe is relevant to their current situation. Our framework allows confining the EMR based on temporal predicates.

Another key issue is that due to the large size of EMRs, which typically include images and videos (e.g., Echo tests), it is impractical to transfer the whole EMR the moment it is requested for the following reason. The current physician may only be interested to examine a small subset of the EMR. Of course at a later time, the physician may request an additional piece from the EMR which she should be able to access without involving the patient again for authorization purposes. Hence, in our framework what is transferred is the right to access a part of the EMR until a specified expiration date (after which the patient has to re-authorize).

A prototype of our VMR exchange approach has been implemented which integrates the i-Rounds[®] medical record system used at Miami Children's Hospital. In particular, a mediator has been built to create a VMR by parsing the web pages generated by i-Rounds. The VMR is transferred using CVM which also handles its presentation through a browser.

In Section 2 we present the necessary background and related work. Section 3 provides an overview of our approach, while Sections 4 and 5 discuss the details of the CVM and the medical data mediator respectively, which are the key components of our approach. Section 6 presents our prototype and we conclude in Section 7.

2. BACKGROUND AND RELATED WORK

We present related work on the areas of medical data exchange, communicating complex data, and mediators.

2.1 Medical data exchange

The rapid adoption of the Internet by healthcare organization, once explained by Metcalfe's Law [Met98], is now fueled by national organizations like the Health Level Seven and DICOM as discussed below. By providing standards for connectivity and information transfer, the Internet is being used more and more for teleconferencing and computer-based patient record systems [LH97]. Distinctions between networked access to multimedia patient records and telemedicine are disappearing as the Internet becomes more powerful and the available bandwidth increases. The proposed EMR exchange model belongs to Level 2 of the integration taxonomy of [WPJ+05], which does not involve tight integration between the information systems.

Forslund et al. introduce the idea of a virtual medical record (VMR), where the actual data and their locations are decoupled from the view of the medical record [FPKC96]. Xiaou et al. propose an XML format for a virtual medical record [XP02]. Their architecture is based on a web server that publishes medical data from heterogeneous sources. This approach is centralized

which makes it hard to maintain. In contrast, our approach is managed locally at each institution which controls what data they want to export and how. Furthermore, we leverage the flexibility of the Communication Virtual Machine (CVM) described in Section 4, to allow exchanging VMRs in a secure and on-demand manner. MML [GAT+03] has been another XML-based approach to define the semantics of a VMR.

We next present some of the dominant medical data exchange standards, which focus on defining the semantics of the recorded data. These standards are complementary to our approach, since by combining them with our data exchange mechanism we can potentially achieve tighter integration, which however is currently out of the scope of this work. The Digital Imaging and Communications in Medicine (DICOM) [DSC06, EAR+05] is a comprehensive specification of information content, structure, encoding, and communications protocols for electronic interchange of diagnostic and therapeutic images and image-related information. JPEG Interactive Protocol (JPIP) [JPIP06] allows transmitting specific tiles of image with specific resolution, quality. JPIP is used by DICOM. The Health Level Seven (HL7) Standard [HLS06, EAR+05] specifies a message model for exchange of text-based medical data, and is currently widely used for intra-institution data exchange. The CEN/TC 251/PT3-033 (European Standardization Committee: Technical Committee for Healthcare, Project Team 22) Request and Report Messages for Diagnostic Service Departments document specifies a semantic data model and model-based compositional rules for messages, but only partial guidelines for electronic document interchange. Unfortunately, the HL7 and CEN/TC 251 specifications leave major communications issues unresolved. Implementors depend on bilateral negotiation between information system vendors to determine parameters for the unspecified details. Using a technique called "context management". CCOW [HCTC06] allows information in separate healthcare applications to be unified so that each individual application is referring to the same patient, encounter or user. CCOW works for both client-server and web-based applications. CCOW does not handle data exchange between remote users but only synchronizes data access for a single user. Finally, IHE (Integrating the Healthcare Enterprise) [IHE06] is an initiative promoting and supporting the integration of systems in the healthcare enterprise.

2.2 Communication of complex data

Current communication systems utilizing complex multimedia data continue to be designed and manufactured with little or no consideration for a common unifying architecture resulting in a fragmented and incompatible set of technologies and products. These systems are typically incapable of adjusting to the end-user's needs, the dynamics of the underlying network, and to new technologies and devices without a costly development cycle [Kre05]. As the electronic medical record evolves to include various types of information such as embedded video, voice, images and text a unifying architecture is needed to allow for the exchange of this data regardless of format and location.

We briefly discuss alternative methods to transfer an EMR and explain why our proposed method of using a mediator and a customizable communication engine is more effective. Clearly, the mediator is necessary to extract the EMR from the institution's systems.

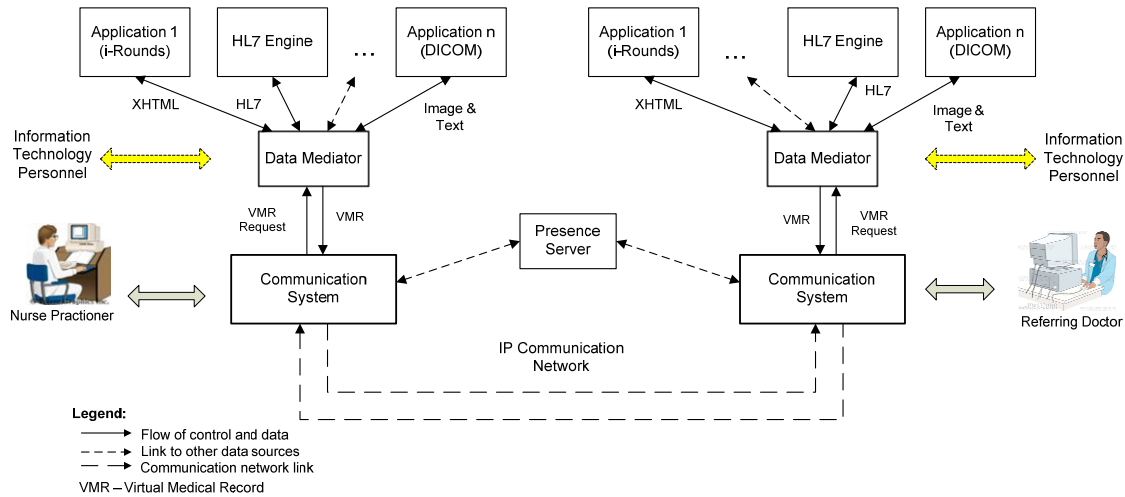


Figure 1: Architecture for on-demand sharing of EMRs

The following technologies are alternatives to the communication engine used in our model. First, the EMR could be represented by a Web page stored on a Web server and maintained by each institution. The drawbacks of this approach are the cost of maintaining a Web server at each institution, the security risk of connecting the institution's system on the internet, and the lack of support by Web servers for special features like access rights based on the date of the source data as well as the time since the authentication was granted. A second approach would be to package all pieces of an EMR in an archive (e.g., zip file) and submit through e-mail. This approach lacks on-demand partial data transfer (the whole, potentially large, EMR is transferred at once), requires a human to package and send the EMR, and provides no control on access rights.

2.3 Medical Data Mediators

A great amount of work has tackled the problem of data mediation [GPQ+97]. In the healthcare domain, due to the widespread use of HL7, a special type of mediator is used. HL7 engines listen for HL7 messages from all applications, format and retransmit them to all interested applications. However, an HL7 engine cannot handle queries on past HL7 data, as it does not store historic messages. In Section 5 we explain how we can exploit HL7 engines in building a data mediator for our purposes.

3. EMR EXCHANGE APPROACH AND ARCHITECTURE

This article discusses the use of a novel clinical content transfer strategy for delivering multimedia patient information over the Internet in a collaborative environment. The involved parties are (a) patients, who should be given control over their EMR, (b) medical practitioners, who should have instant access to all parts of the EMR specified by the patient, and (c) system integration engineers, who can focus on the intra-institution systems integration without having to worry about maintaining open connections to third-party systems.

Our approach does not attempt to define the semantics of the exchanged data, but just allows the construction and transfer of a complex data object (e.g., VMR) which is assembled, transferred and presented in a semantics-agnostic manner on the receiver's

CVM. We next present the high-level abstract architecture of our approach, which we instantiate and make concrete in Section 6.

The architectural diagram in Figure 1 consists of three layers: (1) the health applications layer, (2) the data mediator layer and (3) the communication system layer. There is also a presence server that provides the locator for the healthcare institutions. This server also supports the authentication and encryption process by managing the public and private keys. The health applications layer represents the collection of applications running in a single healthcare facility. The applications may be using different standards such as HL7 [HLS06], DICOM [DSC06] or CCOW [HCTC06], among others. We assume that the healthcare applications in a single institution currently provide interfaces that allow them to be queried for patient data stored in their respective repositories. For example, the i-Rounds system [Teg06] currently used by the Miami Children's Hospital provides a web-based interface that allows healthcare personnel to query the data store in its SQL database. Using the interfaces to the healthcare applications, the data mediator can query one or more of the heterogeneous healthcare data sources in order to generate a virtual medial record to be shared between healthcare institutions.

There are three healthcare stakeholders shown in Figure 1; these are (1) the information technology (IT) personnel – responsible for maintaining the interfaces used to query the applications in the healthcare layer; (2) the nurse practitioner – representing a healthcare professional sending a request for patient information (on the sender side) to another healthcare institution (the receiver side); and (3) a referring doctor – representing a healthcare professional processing a request and accessing data in the healthcare application layer on the receiver side to be sent to the healthcare professional on the sender side.

The duty of the mediator is to generate a VMR given a local patient identifier. This involves reading from the information systems of the institution, which clearly is not trivial as legacy and proprietary systems are abound. We discuss the challenges in building the mediator in Section 5. The communication system (described in Section 4) handles the secure, on-demand, customizable and QoS-enabled transfer of the VMR or its components, which are generated by the local mediator.

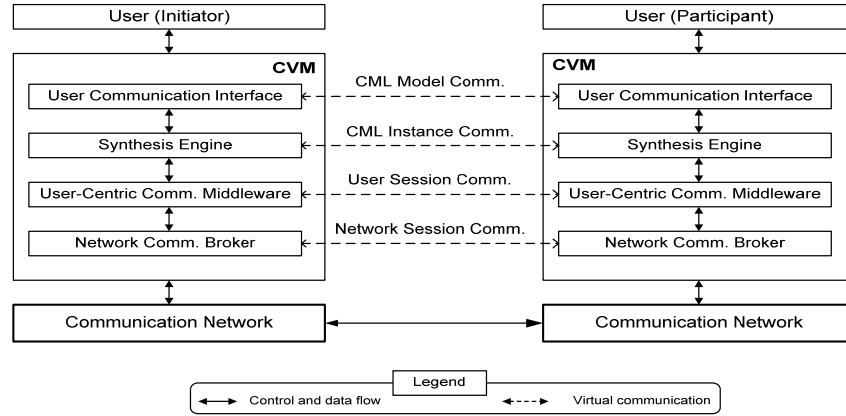


Figure 2: Layered architecture of the Communication Virtual Machine

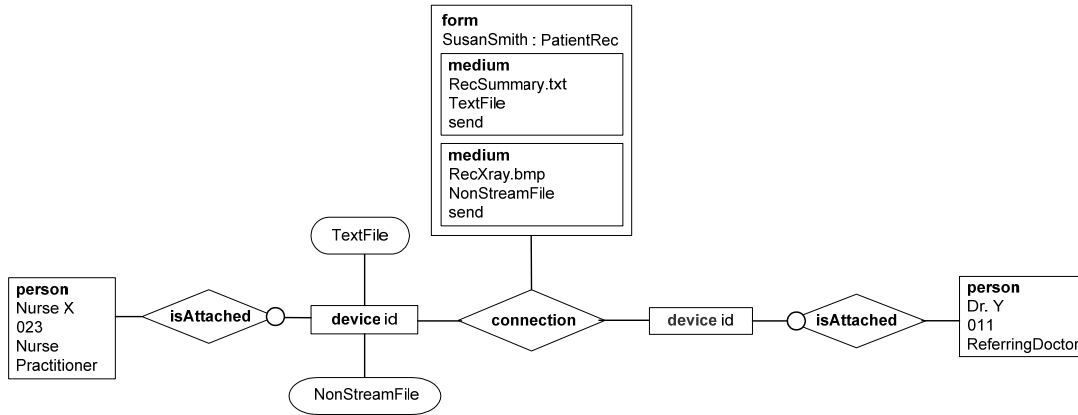


Figure 3: Representation of an EMR exchange scenario using Graphical CML

An important issue to address is that EMRs may be too large to transfer as a whole, since they include multimedia components like X-Rays, Cardiac Echos, Images, and so on. Hence, it is impractical to extract all data from the healthcare applications and copy it to the receiver's side. Furthermore, the receiver may only need a small part of the EMR, but this need may change after some time, when additional data is needed. To tackle these issues, we employ the idea of Virtual Medical Record (VMR) as discussed in Section 2. In particular, when the communication system requests from the data mediator a patient's record, the mediator returns a VMR which describes the items in the patient's record and their locations, but does not include the actual data. (VMR and EMR are often used to denote the same entity in rest of the text.) Then, the remote side can on-demand request for a specific piece of the VMR, in which case the remote communication system will notify the source communication system which will request from its mediator to extract the new piece of data.

3.1 Motivating Scenario

The scenario is based on a situation where a patient underwent surgery for a heart condition at Miami Children's Hospital (MCH) and is moved to Baptist Hospital at the family's request. The description of the scenario is as follows:

1. Susan Smith recently had surgery at MCH and is now at Baptist Hospital recuperating.
2. The nurse practitioner (Nurse X) at Baptist Hospital asks Susan's mother to provide the following information: the name of the attending physician at MCH and a copy of Susan's medical records. The medical record should include: an x-ray, a summary of heart related medical data
3. Susan's mother never got a copy Susan's EMR, but remembers the name of the attending physician at MCH (Dr. Y).

Using the proposed approach, shown in Figure 1, Susan's EMR is retrieved as follows:

1. Nurse X at Baptist Hospital (shown on the left side of Figure 1) logs on to the communication system and searches for the id/location of MCH. To obtain the id/location of MCH the communication system contacts the presence server and queries its repository.
2. Nurse X sends a request to MCH to access the medical record for Susan Smith. The communication system at MCH sends a request to the referring doctor, Dr. Y (on the right side of Figure 1) and the request is validated.
3. The communication system at MCH then sends a request to the data mediator. The data mediator formulates a query specific to the local healthcare application (i-Rounds) and

generates a virtual medical record (VMR) that is returned to the communication system.

4. The communication system at MCH applies the appropriate security and privacy policies and makes the VMR available to the nurse practitioner via the communication system at Baptist.

3.2 Comparison to Alternative Approaches

There are several possibilities of retrieving Susan's medical record, including: (1) Dr. Y compiles an electronic package and sends it via email, (2) MCH uses i-Rounds to store the Susan's EMR therefore Baptist can log onto i-Rounds and retrieve the EMR, and (3) Susan's mother goes to MCH and gets her medical records on a CD or USB key.

We stated the problems associated with such alternatives in Sections 1 and 3.2. The first alternative lacks on-demand partial data transfer, and requires Dr. Y to manually package the EMR. The second alternative requires that Baptist have a compatible healthcare application to the one at MCH. Currently Baptist does not have a system that can interface to the healthcare information systems at MCH. The third alternative would result in Susan's mother making a trip to MCH and waiting until the EMR can be compiled. If Susan's mother obtained her EMR before she left MCH this would be a viable alternative.

4. COMMUNICATION SYSTEM

In this section we first describe the duties and requirements of a communication system capable of supporting our EMR exchange architecture and then give an overview of the specific system we have used in our implementation, which we refer to as Communication Virtual Machine (CVM).

4.1 Communication System Requirements

The functional requirements of the Communication Systems are:

1. Provide an interface to request and receive parts or the whole VMR from the Data Mediator.
2. Provide a mechanism to authenticate the users of the system.
3. Provide a user-friendly interface that allows the healthcare personnel (nurse practitioner) to interact with the system.
4. Hide the complexities of the underlying communication networks.
5. Provide a way to identify/locate other healthcare institution.
6. Provide a method of enforcing the security/privacy concerns required by the various standards [HHS06]. As an example, the system should allow logging of the transferred data.

4.2 Communication Virtual Machine (CVM)

Before going into the details of the CVM system [DSC+06, CHW+06], it is important to clarify that CVM is a general purpose communication system and can support complex communication scenarios defined in an intuitive declarative manner using a *communication modeling language (CML)*. For instance, the user of CVM can on-the-fly define constraints on the number of participants, encryption level, and types of exchanged media, as well as construct complex media types (called *CVM Forms*) using CML. An EMR is merely viewed as an instance of a CVM Form from the point of view of the CVM.

The essence of CVM is that it facilitates a user-centric communication defined using a declarative communication modeling language (CML). As shown in Figure 2, CVM lies between the communication network and the user (or application). The user defines a communication scenario using CML, and CVM handles the execution of the communication.

The design of CVM handles the common tasks performed by communication applications. These tasks include: presentation of the data/information being communicated, interfacing with user devices that are sources/sinks for the data/information, composition of the communication services that the application provides (to end-users), and interfacing with underlying communication network(s) that deliver the data/information. The *first* design principle of CVM is the *separation of concerns*. The flexibility of CVM comes from separating and encapsulating major aspects of communication concerns into individual compartments, which can be handled separately. At a broader level, the CVM encapsulates the specification and processing of user communication logic into a horizontal layer cross cutting different communication applications and services, and by doing so, it allows us to uniformly and systematically address these concerns across these applications. In addition, it provides the conceptual basis to develop uniform interfaces to communication devices and networks and hence makes it possible to manage user communication transparent to the heterogeneity of device types, network protocols and configurations.

As mentioned earlier, the CVM divides communication concerns into four major levels of abstraction cross-cutting different communication applications, which contribute toward implementing the communication service. The four levels of abstraction represent the key artifacts of the CVM and include the: (1) *user communication interface*, which allows users to declaratively specify their communication needs and requirements (in CML), (2) *synthesis engine*, which provides the process and techniques to automatically transform and synthesize a user communication instance (a specification of communication using CML) to an executable form called *communication control script*, (3) *user-centric communication middleware*, which executes the communication control script to manage and coordinate the delivery of communication services to users independent of the underlying network configuration, and (4) *network communication broker (NCB)*, which provides a network-independent API to the user-centric communication middleware and works with the underlying networks to implement the communication services.

Scenario - CVM: Going back to the scenario of Section 3.1, CVM facilitates communication as follows. The nurse practitioner at Baptist hospital interacts with the CVM using a screen similar to the one shown in the upper right corner of Figure 4. The actions of the nurse result in the creation of an underlying communication instance that can be represented in graphical CML, as shown in Figure 3. Susan's patient record (defined as a CVM Form, attached to the connection relation) is the data requested by the nurse practitioner at Baptist Hospital. We also see that the communication instance requires that the receiving device at Baptist Hospital has the capabilities to display text and non-streaming binary data. The constraint on the resolution required to effectively display the x-ray image is also specified in CML (although not displayed in Figure 3 for clarity). For more details on the CML constructs we refer the reader to [CHW+06].

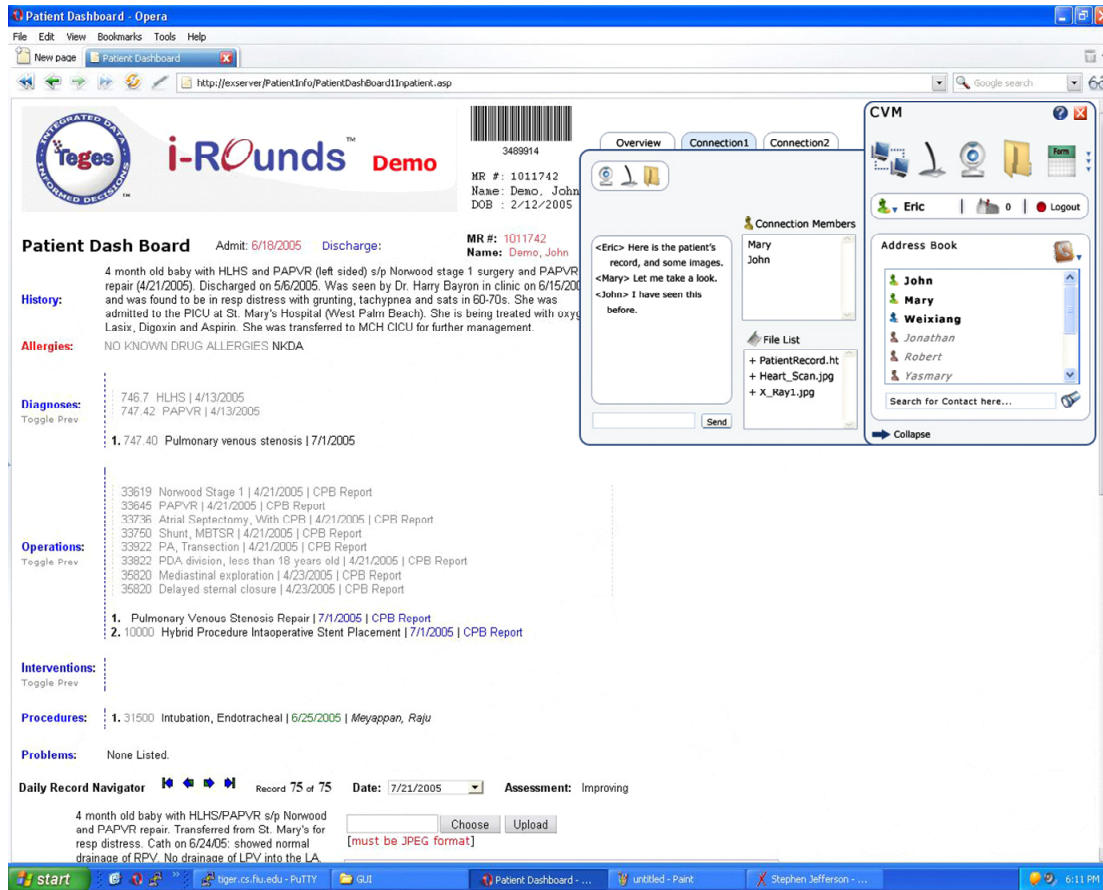


Figure 4: Integration of CVM with the i-Rounds system

5. MEDICAL DATA MEDIATOR

Similarly to Section 4, we first state the requirements of the data mediator for the purpose of our system and then discuss our implementation along with other possible alternatives.

5.1 Medical Data Mediator Requirements

The functional requirements of the Data Mediator are:

1. Given a local patient identifier, it accesses the institution's healthcare applications to construct a locally defined VMR, compatible with the Communication System.
2. Provide a mechanism to enforce access rights and privacy concerns on the extracted data. In particular, when a VMR is requested, the requestor must be authenticated on each of the accessed systems.
3. Provide tools for the IT Personnel to update the mapping templates used to query the healthcare applications.
4. Handle the on-demand transfer of a VMR, that is, only query the pieces of the VMR requested by the Communication System.

5.2 Challenges and alternative approaches in building a medical mediator

Developing and deploying the Data Mediator module is challenging since it needs to access the often proprietary and heterogeneous information systems of each institution, extract all patient-related data and export it into a (syntactic and not semantic) format appropriate for the Communication Engine. We discuss two alternative ways to build this mediator. Notice that building and maintaining the Data Mediator of Figure 1 is orders of magnitude simpler than integrating the information systems of different institutions, since it does not attempt to match the semantic elements across institutions.

The first way, which we have adopted in our prototype (CVM Data Mediator), is to build an interface for each information system of interest (i-Rounds in our prototype) in the institution. In particular, given a local patient identifier, the mediator must extract the relevant data from the information system and package it in a format (CVM Form in the prototype) appropriate for the Communication Engine (CVM in the prototype). The drawback of this approach is the potentially high cost of this mediation if many information systems are used in an institution.

The second way is based on the widespread use of HL7 engines in distributing patient data across an institution. Currently, in many institutions an HL7 engine is intercepting all

HL7 messages from all subsystems and formats them in a suitable way to be fed into the rest of the subsystems. Unfortunately, current implementations of this engine are memory-less, that is, the HL7 messages are not stored in any central repository. However, this is expected to change in the near future when these HL7 messages will be logged in an HL7 database¹. In that case, deploying the mediator will involve building scripts to extract the relevant HL7 messages given a patient id, and formatting them appropriately. The benefit of this approach is that the same mediator will be used in all institutions since HL7 is a standard, which will dramatically reduce the cost of the deployment.

5.3 Architecture of the CVM Data Mediator

The architecture of the CVM Data Mediator, shown in Figure 5, consists of four major components: (1) the *Data Handler* – accepts the VMR request from the CVM and coordinates the packaging of the VMR to be returned to the CVM; (2) the *Domain Application Interface* – formulates the query to be sent to the domain application, in the healthcare applications; (3) the *Presentation Handler* – processes the data returned from the Domain Application Interface and applies any security/privacy policies on the data content, the formatting of the data, and generating the retrieval mapping data; (4) the *local repository* – stores the templates, tags and mapping information used in the CVM data mediator.

The enforcement of security and privacy policies require two steps: (1) the Information Technology Personnel of the healthcare institution provide the CVM Data mediator with the metadata for the data fields used in each of the healthcare applications prior to the VMR request, and (2) during the processing of the request the Presentation Handler performs the required tasks to secure the data before packaging it into a VMR.

Scenario – CVM Data Mediator: Going back to the scenario of Section 3.1, the CVM at MCH receives a request for Susan’s VMR from the CVM at Baptist Hospital. After validation of the request by the CVM at MCH, a request for the VMR is sent to the CVM Data mediator. The Data Handler analyzes the request and sends it to the Domain Application Interface. Domain Application Interface generates a URL and invokes the XHTML pages by making a call to i-Rounds using a URL. The data returned from i-Rounds is then packaged in the Presentation Handler using the CVM Form structure. The security and privacy policies are applied in the Presentation Handler. The Form is then sent back to the CVM at MCH. Note that only the data types specified in the communication instance in Figure 3 are sent to Baptist Hospital.

6. SYSTEM PROTOTYPE

To demonstrate the applicability of our EMR exchange, we built a prototype on MCH’s clinical documentation system called i-Rounds [BW04] and the Communication Virtual Machine (CVM) [DSC+06]. The resulting combination of technology allows multiple users to connect and share medical information through a web browser interface in real-time using any preferred form of communication including: video, audio, images and textual data. The feasibility of using this technology for communicating patient electronic medical records in a secure and stable teleconference environment over the Internet was tested.

¹ It is relatively simple to add logging to current HL7 engines.

We have completed the first version of the CVM prototype and have integrated it with the i-Rounds system [Teg06]. We have deployed the prototype at FIU and MCH. Figure 4 shows a screenshot of the CVM embedded in the i-Rounds system. The CVM prototype has been implemented using web-based technologies. The user interface of the CVM, shown in the upper right hand section of the browser in Figure 4, has been deployed with the Opera 8.5, a voice enabled browser [Ope06]. The technologies used to develop the CVM user interface and the Opera browser include: HTML, Javascript and XHTML+Voice. Javascript provides a means to implement the logic for the CVM user interface.

The CVM prototype adheres to the architecture in Figure 1, where the medical application used is i-Rounds. The i-Rounds system is a Web-based voice-activated software solution that can extract data from existing hospital database sources, such as patient information, lab results, and bedside monitors [Teg06a]. The CVM prototype also uses voice recognition. The i-Rounds system provides an easy to navigate interface that is based on the “patient dash board” concept. The screenshot in Figure 4 show the dash board for patient “John Demo”. The patient dash board allows a user of the i-Rounds system to access any component of an EMR and allows the users to easily return to the dash board from any of the of the components of a patient’s EMR. When a request is made to the i-Rounds system a query is formulated dynamically and sent to the hospital data source to retrieve the patient data. This data is then displayed in the i-Rounds browser. Figure 5 shows the architecture of the data mediator used in our prototype. More details on the workflow of the control signals and data in the prototype system can be found in the appendix.

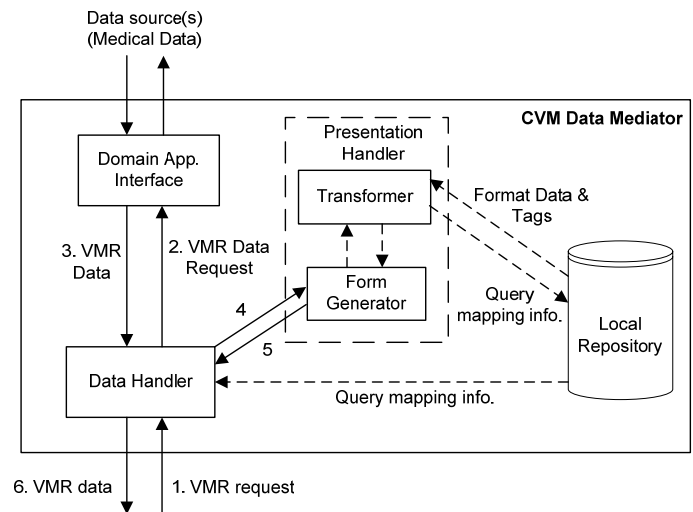


Figure 5: Architecture of the CVM Data Mediator

7. CONCLUSIONS AND FUTURE WORK

We studied the impediments in building a medical data exchange mechanism between healthcare institutions. These impediments are mainly non-technological, but social, political and privacy-related. We proposed an approach that offers an acceptable degree of integration between institutions without violating the

sociopolitical limitations or requiring expensive deployment effort. In particular, in our approach the patient (or her family) is responsible to authenticate and initiate an EMR transfer. The cost of deploying this approach is greatly decreased by employing a smart communication virtual machine (CVM) to handle security, Quality of Service, capabilities negotiation and network abstraction. The EMR is transferred using the CVM in a semantics-agnostic manner, independent of the current semantics standardization efforts.

In the future we plan to investigate ways to increase the level of integration by exploiting widespread standards like HL7. In particular, we will study the feasibility of importing EMRs to an institution's information system, instead of just viewing them as our current approach does. Furthermore, we will study the requirements of a central system to manage the access rights and authenticate users across the nation's institutions. We will compare that approach to the peer-to-peer approach we currently employ.

8. REFERENCES

- [BW04] Burke RP, White JA. Internet rounds: A congenital heart surgeon's web log. *Semin Thorac Cardiovasc Surg.* Fall 2004;16(3):283-292.
- [Cap06] CapMed. <http://www.capmed.com>, 2006
- [Con06] Joseph Conn. Identity crisis? Renewed debate over national patient ID In *Modern Healthcare* magazine <http://www.modernhealthcare.com/article.cms?articleId=39954>, 2006
- [CHW+06] Clarke PJ, Hristidis V, Wang Y, Prabakar N and Deng Y. "A Declarative Approach for Specifying User-Centric Communication." *International Symposium on Collaborative Technologies and Systems (CTS 2006)*.
- [DSC06] DICOM Standards Committee. Digital Imaging and Communications in Medicine (DICOM). <http://medical.nema.org/> April 2006.
- [DSC+06] Deng Y, Sadjadi SM, Clarke PJ, Zhang C, Hristidis V, Rangaswami R, and Prabakar N. A Communication Virtual Machine. 30th Annual International Computer Software and Applications Conference (COMPSAC 06).
- [EAR+05] Eichelberg M, Aden T, Riesmeier J, Dogac A, and Laleci GB. A survey and analysis of Electronic Healthcare Record standards. *ACM Comput. Surv.* 2005; 37(4): 277-315.
- [FPKC96] Forslund, D. W. et al. Experiences with a distributed virtual patient record system. In *Proceedings of the 1996 American Medical Informatics Association Annual Fall Symposium* (Washington, D.C., Oct. 26-30). J. J. Cimino, Ed. American Medical Informatics Assoc., Hanley & Belfus, 1996, pp. 483-487.
- [GAT+03] Guo J, Araki K, Tanaka K, Sato J, Suzuki M, Takada A, Suzuki T, Nakashima Y, Yoshihara H. The latest MML (Medical Markup Language) version 2.3--XML-based standard for medical data exchange/storage. *J Med Syst.* 2003 Aug;27(4):357-66.
- [GPQ+97] Hector Garcia-Molina, Yannis Papakonstantinou, Dallan Quass, Anand Rajaraman, Yehoshua Sagiv, Jeffrey Ullman, Vasilis Vassalos and Jennifer Widom. The TSIMMIS Approach to Mediation: Data Models and Languages. *Journal of Intelligent Information Systems*, 1997
- [HCTC06] HL7 CCOW Technical Committee. Clinical Context Object Workgroup (CCOW). http://www.hl7.org/special/Committees/ccow_sigvi.htm, April 2006.
- [HHS06] United States Department of Health and Human Services: Office of Civil Rights-HIPAA. <http://www.hhs.gov/ocr/hipaa/> (May 2006).
- [HLS06] Health Level Seven Inc. Health Level Seven (HL7). <http://www.hl7.org/>, April 2006.
- [IHE06] Integrating the Healthcare Enterprise, <http://www.ihe.net/>, 2006
- [JPIP06] JPEG Interactive Protocol (*JPIP*), <http://www.jpeg.org/jpeg2000/j2kpart9.html>, 2006
- [Kre05] David Krebs. "The Mobile Software Stack for Voice, Data, and Converged Handheld Devices", Mobile and Wireless Practice Venture Development Corporation, April 2005.
- [LH97] D Lindberg and B Humphreys, "Medical Informatics", *JAMA*, 227 (18 Jun 1997) 23, pp. 1,870-1,871.
- [Met98] R Metcalfe, "From the Ether Metcalfe's Law: A Network Becomes More Valuable as it Reaches More Users", *InfoWorld*, 1995.
- [Ope06] Opera 8.5 <http://www.opera.com/>
- [OEM04] J. Marc Overhage, Lori Evans, Janet Marchibroda. Communities' Readiness for Health Information Exchange: The National Landscape in 2004. *J Am Med Inform Assoc.* 2005;12:107-112
- [Teg06] TegesTM Corporation. i-Rounds. <http://www.teges.com/index.asp>, May 2006.
- [Teg06a] TegesTM Corporation. i-Rounds – Integrated Clinical Information System. <http://www.teges.com/PDF/iRoundsProductSheet.pdf>, May 2006.
- [Wae03] C. Peter Waegemann. EHR vs CCR: What is the difference between the electronic health record and the continuity of care record? *Medical Records Institute*, <http://www.medrecinst.com/pages/libArticle.asp?id=42>, 2003
- [WPJ+05] Jan Walker, Eric Pan, Douglas Johnston, Julia Adler-Milstein, David W. Bates, Blackford Middleton. The Value Of Health Care Information Exchange And Interoperability. *Health Affairs*, January 2005
- [XP02] Xiaou Z, Keng PH. XML-Based Virtual Patient Records System for Healthcare Enterprises. 4th International Conference on Enterprise Information Systems. Ciudad-Real, Spain, April 3-6 2002.

Appendix - Technical details of transfer of VMR using the CVM

The steps to process a VMR data request by the CVM are as follows: (1) the request is sent to the Data Handler to be processed, this involves determining if any mapping information from the repository is required (this is required to generate the query for the data sources); (2) the data request is passed to the Application Communication Interface where a query is generated based on the data source(s) to be queried; (3) the results from the data source(s) are then sent through the Application Communication Interface to the Data Handler; (4) the data from the data source(s) is then passed to the Presentation Handler to be packaged into a VMR using the Form Generator and the Transformer; (5) the Presentation Handler then passes the VMR to the Data Handler; (6) the Data Handler sends the VMR to the CVM for transmission to the CVM at the institution that made the request.

The Presentation Handler (shown in the center Figure 5) creates a CVM Form that is used to package the VMR and deliver it to the CVM at the receiver institution. The Presentation Handler consists of the Form Generator and the Transformer. The Form Generator is where the data of the VMR is packaged using a CVM Form construct. There are two classes of VMR data that use a CVM Form: (1) VMR data with presentation information, and (2) VMR data without presentation information. In our prototype the VMR generated from i-Rounds contain presentation information. The Transformer is the component that processes the source containing the presentation information (e.g. HTML or XML) and packages the source as a text file, which becomes part of the CVM Form. To maintain the presentation information for a VMR we use the Transformer to convert the presentation source (HTML or XML) to a text file by applying metadata for the data source as specified by the information technology personnel. We provide additional details on the transformation in the next section.

The main entity used to transfer a VMR across a communication connection is the CVM Form. The CVM Form is a data structure that groups structured or unstructured related data for the purpose of data communication. The CVM Form also provides a default presentation structure that allows access to the data contained in the CVM Form. The structure used to present the CVM Form is a nested two dimensional grid structure. If an application uses specific source code (e.g., HTML or XML) to display the contents of the VMR then the source containing the presentation information is preprocessed and stored as a text file in the CVM Form. A CVM Form may consist of basic CVM data types or other CVM Form types i.e., it may be recursively defined. The basic CVM data types include: text files, audio files, video files, audio/video files, non-streaming files, live audio streams, live video streams and live audio/video streams [CHW+06].

The CVM Form is defined using the communication modeling language (CML) [CHW+06]. The CML definition of the CVM Form contains basic data types, other CVM Form types and attributes. The CVM Form attributes specify: (1) the suggested application that should be used to display the Form,

(2) the action to be taken at the sender's side and receiver's side of the communication for the Form (send, doNotSend, start, request), (3) the list of persons having access to the Form, (4) the encryption level of the Form (low, medium, high), and (5) the expiration of the access to the contents of the Form. We plan to extend the CVM Form to have constraints on the media transmitted across a communication connection.

The CML provides a means to separate the attributes in the Form type into data provided during communication and metadata provide by the healthcare technology personnel prior to communicating the VMR. The metadata for the CVM Form include values for any combination of the attributes mentioned in the previous paragraph. For example, given the Form type "MCH-patient record" the metadata may include the following: (1) suggested application – "Opera browser", (2) action – "send", (3) person role – "attending physician, surgeon", and (4) encryption level – "high". The suggested application requires the Opera browser due to the voice activated commands, the action "send" specifies that the Form should be sent to the receiver's side, the person role attribute states that only a person with role attending physician or surgeon have access to the Form, and the encryption level is "high" meaning that all data in the Form is encrypted using at least a 128 bit long key.