

Sharing Electronic Patient Records Among Providers via the World Wide Web

by

John D. Halamka

B.A. Public Policy
Stanford University, 1984

B.S. Medical Microbiology
Stanford University, 1984

M.D.
University of California, San Francisco, 1993

MIT LIBRARIES

MAY 12 1998

SCHERING

Submitted to the Harvard University-Massachusetts Institute of Technology

Division of Health Sciences and Technology Program

in Partial Fulfillment of the Requirements for the Degree of

Master of Science in Medical Informatics

at the

Massachusetts Institute of Technology

February 1998

© 1998 John D. Halamka. All rights reserved.

The author hereby grants to MIT permission to reproduce and to distribute publicly paper and electronic copies of this thesis document in whole or in part.

Signature of Author.....
Health Sciences and Technology Division
December 29, 1997

Certified by.....
Charles Safran MD,MS
Associate Professor of Medicine
Thesis Supervisor

Accepted by.....
Martha L. Gray MD
Co-director, Health Sciences and Technology

APR 14 1998

SCHERING

LIBRARIES

Sharing Electronic Patient Records Among Providers via the World Wide Web

by

John D. Halamka

B.A. Public Policy
Stanford University, 1984

B.S. Medical Microbiology
Stanford University, 1984

M.D.
University of California, San Francisco, 1993

Submitted to the Harvard University-Massachusetts Institute of Technology

Division of Health Sciences and Technology Program

in Partial Fulfillment of the Requirements for the Degree of

Master of Science in Medical Informatics

ABSTRACT

With the reorganization of healthcare and consolidation into Integrated Healthcare Delivery Systems, medical records are increasingly distributed across multiple institutions. Timely access to these medical records is a critical need for healthcare providers. This document provides an overview of an architecture for secure retrieval of electronic health information from heterogeneous data sources using Health Level 7 (HL7), internet technologies and readily available software components. CareWeb™, a consolidation of the electronic records of Boston's Beth Israel and Deaconess Hospitals, was implemented and evaluated for this project. This report details the technical, political and organizational challenges encountered during the creation of CareWeb™ and reviews the techniques required to electronically link Integrated Health Care Delivery Systems and geographically distant information resources.

Thesis Supervisor: Charles Safran MD,MS

Title: Associate Professor of Medicine, Harvard Medical School

Contents

1. Introduction
2. Background
3. Data Sharing
 - 3.1 Devising a Common Medical Record
 - 3.2 Strategies for the Master Patient Index
 - 3.3 Vocabulary Issues
4. Architecture
 - 4.1 Overview
 - 4.2 Site Servers
 - 4.2.1 HL7 Messaging
 - 4.2.2 Legacy System Interfacing - MUMPS
 - 4.2.2.1 ODBC
 - 4.2.2.2 Visual M
 - 4.2.2.3 M/Weblink
 - 4.2.3 Legacy System Interfacing - Relational Databases
 - 4.2.4 Web technologies
 - 4.2.4.1 Common Gateway Interface
 - 4.2.4.2 ISAPI
 - 4.2.4.3 ASP
 - 4.3 Consolidator
 - 4.3.1 HTTP Messaging
 - 4.3.2 User Interface
 - 4.3.3. Browser related issues

- 5. Security and Confidentiality
 - 5.1 The National Research Council Report
 - 5.2 Enterprise-wide Authentication
 - 5.3 Access Validation
 - 5.4 Multi-organizational Audit Trails
 - 5.5 Protection of External Communications
 - 5.6 Cryptography methods
 - 5.7 Electronic authentication of records
 - 5.8 Physical security and disaster recovery
 - 5.9 Software discipline
 - 5.10 System assessment
- 6. Evaluation
 - 6.1 Technical Evaluation
 - 6.2 User Evaluation
 - 6.2.1 Information Systems Professionals
 - 6.2.2 Health Care Providers
 - 6.2.3 Patients
 - 6.3 Future evaluation
- 7. Future Work
 - 7.1 Master Patient Index Investigation
 - 7.2 Multimedia extensions
 - 7.3 Vocabulary server
 - 7.4 Bili Light
 - 7.5 Affiliated Health Networks of New England
- 8. Organizational Issues
- 9. Conclusions
- 10. References

1. Introduction

In an era of increasing competition for health care dollars, medical institutions are merging and consolidating with increasing frequency. Given that the majority of such institutions have heterogeneous hospital-based computing resources, integrating information systems across merged institutions is a difficult problem.

The CareGroup was formed in 1996 by the merger of the Beth Israel Hospital, the Deaconess Hospital, three Boston area community hospitals, and several satellite outpatient clinics, creating a billion-dollar integrated healthcare delivery system. A major post-merger issue has been the integration of existing electronic health information to deliver coordinated care to the patients who crossover between CareGroup facilities. Each site has different legacy systems, different institutional vocabularies and varying completeness of clinical information.

A standard approach to the unification of diverse information sources is to use a clinical data repository. Repositories serve as enterprise-wide data warehouses but require substantial monetary and personnel investments to deploy. The CareWeb™ project was conceived to provide a means for the virtual consolidation of the health information at these heterogeneous institutions while minimizing time, personnel and cost.

CareWeb™ is an actual implementation of the W3EMRS architecture [1]. The W3EMRS architecture provides the first approach to delivering a unified virtual medical record for both inpatient and outpatient care settings. With W3EMRS, a request for information is made via a web browser to a consolidator. The consolidator requests information from several heterogeneous hospital, clinic and office sites. A site server is implemented for each site which interprets the consolidator request and returns the requested information as an HL7 data stream. The consolidator merges data from all responding sites and returns a unified web page to the web browser. The W3EMRS project is an example of a World Wide Web-based approach that has delivered promising results in a laboratory environment.

On-going research continues to support the suitability of web technology for clinical applications, although deployed systems in a clinical environment are rare. Several authors have explored the use of the web for display of medical records for a single institution [2,3] . Others have proposed multi-institutional approaches [4] utilizing a monolithic central repository, updated at relatively infrequent fixed intervals, which limits the timeliness of data.

CareWeb™ uses the World Wide Web to successfully consolidate heterogeneous clinical data across multiple institutions. CareWeb™ implements web-exposed HL7-based [6] medical information servers at each participating institution in the healthcare delivery network. A central “Consolidator” processes requests for information from healthcare providers and queries all sites on the network. The Consolidator then delivers an integrated multi-institutional medical record to the health care provider.

Creating such a system presented many issues, both technical and political, including definition of a common medical record, creation of a master patient index strategy and implementation of a security/confidentiality policy.

2. Background

Several systems have been developed to deliver online patient data for use in outpatient settings in a text-based environment. The earliest and perhaps best known is the COSTAR system. COSTAR, which stands for Computer Stored Ambulatory Records, derives from the work of Grossman and Barnett [7] and was originally introduced at the Harvard Community Health Plan in 1968. More than 750 practices attempted installation and approximately 150 sites still use the software.

Recent descriptions of installations at the Medical College of Ohio [8] and the University of Nebraska [9] indicate substantial barriers to the wide dissemination of the online patient record. While many of these barriers are not technological but socio-political, simple problems with interfacing of laboratory systems and hospital registration systems have led to duplicative data entry and user dissatisfaction. Moreover, many of these systems including the Regenstrief Medical Record System [10], the Total Medical Record System (TMR) [11] and the STOR system [12] are paper based, limited to one specialty (medicine) and rarely integrated with other heterogeneous systems throughout the institution.

Numerous institutional on-line medical record systems have been developed, including the Beth Israel Online Medical Record (OMR) system [13], the HELP System at LDS Hospital [14], the Diogene system in Geneva [15] and the BAZIS system in the Netherlands [16]. Each of these systems has focused on the sharing of data within a single institution to facilitate the process of care. The OMR does include connectivity to geographically dispersed sites affiliated with the Beth Israel, including two community practices and 32 specialty clinics.

Today, an information system with a single institutional focus is not sufficient. Hospitals, outpatient clinics, and urgent care centers throughout the country are continuing to merge to create full-service medical service organizations. Beth Israel Deaconess Medical Center is an excellent example of this trend, having recently completed a full asset merger that allied the hospital with several satellite emergency rooms in the Boston area. For these distributed care models to be effective, clinical data must be immediately accessible at all sites within the medical service network.

Attempts to provide immediate distributed access to medical data have so far either been incomplete, or implemented only in a laboratory setting. The National Information Infrastructure Health Information Network (NIIHIN) [2] project created an architecture with a single central repository supplied on an interval basis with patient data extracted from participating hospitals. This approach suffers from a lack of timeliness as hospitals may submit data on a monthly or greater interval. Additionally, as each hospital information system evolves, it is up to the central provider to deal with continually changing hospital data.

The DEEDS Data set [5] proposed by the CDC provides an architecture to record emergency department visits for submission to a central authority for later data analysis. Although it provides a comprehensive means to exchange emergency department data, it provides no means for recording inpatient data and exchanging inpatient test results.

The W3EMRS architecture [17,18] provides the first approach to delivering a unified virtual medical record for both inpatient and outpatient care settings. With W3EMRS, a request for information is made via a Web browser to a consolidator. The consolidator issues requests for information to several heterogeneous hospitals, clinics and offices. A site server is implemented for each site which interprets the consolidator request and returns the requested information as an HL7 data stream. The consolidator merges data from all responding sites and returns a unified Web page to the requesting browser. The W3EMRS project has implemented a cross institutional medical record which includes demographics, problems, medications, allergies, notes and visits but does not include test results nor multimedia objects.

The W3EMRS project is an example of a World Wide Web based approach that has delivered promising results in a laboratory environment..

The Regenstrief Medical Record System (RMRS) is used at three hospitals on the Indiana University Medical Center Campus and more than thirty off campus clinical sites. Recently, this system has been extended to use Web Browser clients for display and entry of data. Their recent experience with Web

technology demonstrates success with the GUI interface, platform independence, security and straightforward integration. [19]

The University of Minnesota has deployed Web clients to interface to its Clinical Information System. Specifically, Clinical Laboratory Result Reporting, Microbiology Display, forms based order entry and hospital policy and procedures have been Web enabled. They conclude that the Web-based interface poses few limitations and has met a substantial number of their clinical information needs. [20]

The System for Text Archive and Retrieval (STAR) project at the University of Missouri has implemented a complex architecture providing access to text based legacy system records using Web clients. [21]

From the work done on cross-institutional records thus far, it is clearly advantageous to link existing heterogeneous sites in real-time to eliminate the problems created by infrequent data deliveries to central repositories. From the Web interfaces implemented by several institutions there is compelling evidence that the Web browser can be used successfully as an access method for both text-based medical records and multimedia objects.

3. Data Sharing

3.1 Devising Common Medical Record

To create an integrated medical record across heterogeneous institutions, the data elements to be shared must be identified. Several authors have defined common medical records for cross-institutional exchange. Examples include the EMDS[2] dataset from the NIIHIN] project and the DEEDS[3] dataset from the Centers for Disease Control.

To create a CareWeb common medical record, we explored the datasets of the healthcare information systems deployed in the CareGroup hospitals. Some institutions, such as the Beth Israel hospital, have complete online medical records with detailed notes for most visits and procedures [22]. Others, such as the Deaconess-Glover, currently keep only basic demographic and visit information on-line.

The power of the CareWeb™ concept is its ability to provide a consolidated view of the existing records of multiple institutions. To maximize the information displayed, we designed a common medical record that includes patient demographics, medications, allergies, visits and notes. For the initial deployment of CareWeb™, we elected to limit our data exchange to text and we did not include multimedia objects such as EKG's, X-rays, and heart sound recordings. This is left to future work (section 7.4)

The common medical record is presented below, with annotations indicating the details of the data to be shared.

The patient demographics common medical record contains the following fields:

CAREWEB FIELD NAME	DESCRIPTION & NOTES
Patient ID	An identifier for a person which allows the appropriate linkages of patient information. This is used exclusively as a unique identifier and there is no intelligence built into this key. The true logical identifier is a combination of attributes (SSN, DOB, name, etc...) which should be specified by the Master Patient Index.
Patient's Last Name	The current legal last name of a person. This attribute is synonymous with family name or surname.
Patient's First Name	The current legal first name of a person. It is also referred to as the given name.
Patient's Middle Name	The current legal middle name of the person.
Patient's Suffix	The suffix of the person's name. Suffix refers to hereditary order, such as Jr., Sr., III, or IV.
Patient's Prefix	The person's official title such as Dr, Mr, or Mrs.
Mother's First Name	This attribute is used for identification purposes only.
Father's First Name	This attribute is used for identification purposes only.
Date of Birth	The date and time of the person's birth. This is used as standard identifying information that helps distinguish between patients with the same name. When available, it can be used to calculate a person's age.
Sex	A classification based on reproductive function for a person. Female(F), Male(M), Other(O), Unknown (U)
Patient Alias's Last Name	The last name in a person's alias. Aliases include nicknames, previously used names resulting from marriages, and legal changes of name. Aliases are recorded to longitudinally piece together a person's health care history even when a name may change.
Patient Alias's First Name	The first name in a person's alias.
Patient Alias's Middle Initial or Name	The middle name in a person's alias.
Patient Alias's Suffix	
Patient Alias's Prefix	
Race	Patient reported racial background
Address (Street Number and Name)	The street location or other designation, such as an internal code, for the address.
Address (Other)	
Address City	The city or township for the address.
Address State	The state, territory, or district for the address.
Address Zip Code	The zip code for the address.
Address Country	The country in which the address is located.
Home Phone Number	The specific contact address for a person. The phone number may be

	prefaced with the country code and may have an extension number attached. Telephone numbers are useful for patient follow-up and for contacting providers. The contact address may also be an email address.
Business Phone Number	The specific contact address for a person. The phone number may be prefaced with the country code and may have an extension number attached. Telephone numbers are useful for patient follow-up and for contacting providers. The contact address may also be an email address.
Primary Language	Two character code
Marital Status	Single(S), Married(M), Divorced(D), Widowed(W)
Religion	Two character code
Social Security Number	The personal identification number assigned by the US Social Security Administration. Social Security Number is frequently used as an individual identifier and may be useful for linkage with some outside data systems. Not all patients will have SSNs (newborns, illegal aliens, international patients) though. The Master Patient Index will assign a unique identifier for a person, recorded in person ID, based on the values entered in name, social security number, date of birth, etc...
Ethnic Group	Two character code
Birth Place	City, State, Country

The problem list common medical record contains the following fields:

CAREWEB FIELD NAME	DESCRIPTION & NOTES
Problem ID	The numerical encoding of the problem
Problem Name	A short name for the problem on a person's problem list. The problem list contains a running list of historical, chronic medical problems that provides significant information to other health care providers. The source of the problem list could be a diagnosis or just general notes entered by a physician. On occasion, providers will enter items on a problem list that are not related to an encounter. Examples include finding out something about a patient from another family member, or from a physician that does not participate in the CareWeb system. Primary care physicians should also do a quarterly review and update of CareWeb data which may or may not be related to an encounter.
Problem Coding Standard	ICD-9
Problem Established Date	Date the problem was entered in the person's problem list. This attribute provides some indication of currency, accuracy, and validation for health care providers.
Problem Life Cycle Status	Only active problems are included in the CareWeb problem list.

The medications common medical record contains the following fields:

CAREWEB FIELD NAME	DESCRIPTION & NOTES
Medication ID	The numerical encoding of the medication
Medication Name	The trade or catalog name for a drug. The product names used in the National Drug Code Directory are generally those supplied by the labelers (firms) as required under the Food, Drug, and Cosmetic Act.
Coding Standard	National Drug Code Directory.
Dose	The quantity or amount of the medication in a single dose. The knowledge of how often a medication is given is important in deciding potential changes in therapy from a given encounter. It is also important in estimating the potential supply of medication in a household in the event of an overdose.
Quantity Dispensed	The actual quantity supplied
Dosage Units Code	An encoded indication of how the dosage is measured in units.
Dosage Units Text	An free text indication of how the dosage is measured in units.
Provider Admin Instructions	Additional instructions or comments providing further information about the drug or medication. This could include special instructions for constructing a custom preparation, or notes on when and how to reduce a dosage amount.

The allergies common medical record contains the following fields:

CAREWEB FIELD NAME	DESCRIPTION & NOTES
Allergy Code	The numerical encoding of the medication
Allergy Name	The name of the allergen or agent that can cause an allergic or adverse reaction in a person. Emergency department providers routinely collect allergy information to establish or rule out the use of certain drugs during the course of treatment.
Allergy Coding Standard	No coding standard identified for Allergies
Identification Date	The date or estimated date that any adverse reaction to a specific allergen or element was first encountered by a person.

The visits common medical record contains the following fields:

CAREWEB FIELD NAME	DESCRIPTION & NOTES
Patient Class	Emergency (E), Inpatient (I), Outpatient(O)
Admit Date/Time	The date and time that the encounter began for a patient which could be either the patient's arrival time or the registration time. This information provide a close approximation as to how long the encounter lasted.
Discharge Date/Time	The date and time that the encounter ended, either the discharge or departure time stamp.
Admit Reason Code	The numerical encoding of the admission diagnosis
Admit Reason Name	A short name or description of the reason for the encounter. In emergency departments, this is often known as the chief complaint and may be recorded using the patient's terminology. For other types of encounters, such as outpatient or home care, the encounter reason may be more generic and may not indicate a specific complaint.
Admit Reason Code Table	ICD-9.

The notes common medical record contains the following fields:

CAREWEB FIELD NAME	DESCRIPTION & NOTES
Observation Date/Time	The actual date and time of the recorded observation.
Observation Value	The classification code specified in the ICD9 scheme that will be used for diagnoses. ICD9 codes are used to facilitate data analysis, common interpretation, statistical data reporting, and claims reimbursement.
Observer ID	An identification number for a health care provider which allows for the identification of a provider but maintains confidentiality of the identity of the provider. This element should be based on the Health Care Financing Administration's National Provider Identification File.
Observer Last Name	The current legal last name of a person. This attribute is synonymous with family name or surname.
Observer First Name	The current legal first name of a person. It is also referred to as the given name.
Observer Middle Initial	The current legal middle name of the person.
Observer Suffix	The suffix of the person's name. Suffix refers to hereditary order, such as Jr., Sr., III, or IV.
Observer Prefix	The person's official title such as Dr, Mr, or Mrs.
Observer Degree	The short character designation for an educational degree, such as PhD or MD.

3.2 Strategies for the Master Patient Index

Much work has been done on the details of creating master patient indexes [23]. Examining the types of demographic data available across the CareGroup institutions, we elected to use two master patient index strategies.

For competent patients who are able to communicate identifying information, we elected to use patient last name, patient first name, date of birth, gender, mother's first name and father's first name to identify patients uniquely. Although not perfect, we believe that this strategy will suffice for more than 85% of our patients. [31]

For incompetent or critical patients, we elected to use patient last name, patient first name, date of birth and gender. Although potentially not precise enough to identify patients uniquely, we believe that this

strategy will maximize the information available to a health care provider about an otherwise non-communicating patient.

Further validation of the efficacy of this master patient index strategy will be the subject of future investigation (section 7.3)

3.3 Vocabulary Issues

Each institution in the CareGroup has a different institutional vocabulary. While the Beth Israel Hospital problem list might include Hypertension, the Deaconess Hospital problem list might include High Blood Pressure.

Much work has been done in the past on uniform vocabularies such as SNOMED, MeSH and the MetaThesarus. For the initial CareWeb deployment we elected to display the institution specific vocabularies and not attempt a uniform vocabulary conversion. An interface to the MetaThesarus has been designed and is in the process of implementation. (section 7.5)

4. Architecture

4.1 Overview

The clinical data at the Beth Israel Hospital is stored in a comprehensive, custom built MUMPS based system composed of 28,000 programs. The clinical data at the Deaconess Hospital is stored in a Sybase clinical data repository.

CareWeb™ site servers operate behind the web servers of each hospital and create a link to the underlying legacy systems at each institution. These site servers interpret incoming HL7 requests for information, translate them into specific legacy system queries and package the resulting information into an HL7 response.

To allow users to query multiple hospitals simultaneously, we developed a CareWeb™ “Consolidator”, which processes user requests, dispatches them to multiple hospitals’ site servers, and processes the information retrieved

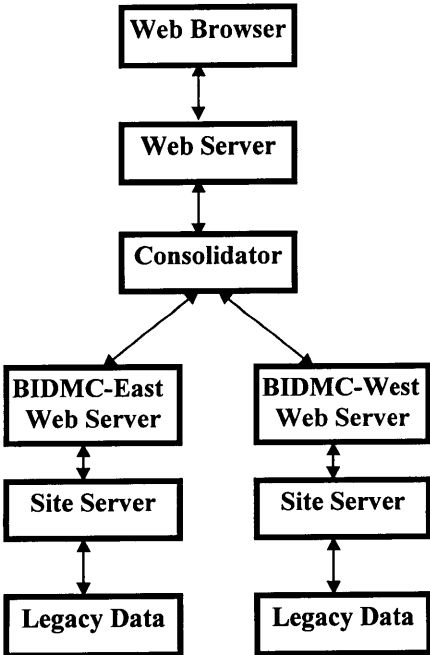


Figure 1 - The CareWeb™ Architecture

A typical session begins when a health care provider on a standard web browser creates a query for information by specifying patient identification. This information is submitted via standard HTML forms to the Consolidator. The Consolidator generates an HL7 request for information to both the Beth Israel and Deaconess site servers. The site servers return HL7 encoded demographics, problems, medications, allergies, notes and visits. The Consolidator interprets the incoming messages and creates a single unified presentation which it sends back to the health care provider as a series of web pages. Full navigational control is enabled with tool bars that allow the medical record to be scanned using a tab folder-like paradigm.

The CareWeb™ project was implemented using Visual Basic complemented by ActiveX components for HL7 messaging, HTTP communications, and legacy system interfacing. The Consolidator and site servers were created as multithreaded, interrupt-driven Internet Information Server Active Server Pages [24].

4.2 Site Servers

Site servers reside at each hospital which participates in CareWeb™. The Beth Israel and Deaconess site servers are were created as Active Server pages calling OLE Automation methods coded in Visual Basic. Each site server interprets receives an incoming HL7 message from the consolidator, parses the message and retrieves the desired information from the hospital's legacy system which is repackaged into HL7 and returned to the consolidator. The two basic components found in a site server are the HL7 messaging and legacy system interfacing.

4.2.1 HL7 Messaging

Health Level Seven provides a series of standards that can be used between collaborators to exchange healthcare information. However, the standard provides such a flexible array of options that there is no single correct way to exchange a particular piece of data. For example, a discharge diagnosis could occur in the Problem segment (PRB), the Visit segment (PV1) or as an observation (OBR). Collaborators who wish to exchange data using HL7 must agree upon a common semantic mapping for exchange. The

following is the HL7 mapping of the common medical record data which was agreed upon by the W3-EMRS collaborative and used in the implementation of CareWeb.

PID Segment

<u>Field</u>	<u>Value</u>	<u>Field</u>	<u>Component</u>
1	Patient ID	3	0
2	Patient's Last Name	5	1
3	Patient's First Name	5	2
4	Patient's Middle Initial or Name	5	3
5	Patient's Suffix (e.g ., Jr or II)	5	4
6	Patient's Prefix (e.g., Dr)	5	5
7	Mother's Maiden Name (Last Name only)	6	1
8	Date of Birth	7	0
9	Sex (F, M, O, U)	8	0
10	Patient Alias's Last Name	9	1
11	Patient Alias's First Name	9	2
12	Patient Alias's Middle Initial or Name	9	3
13	Patient Alias's Suffix	9	4
14	Patient Alias's Prefix	9	5
15	Race	10	0
16	Address (Street Number and Name)	11	1
17	Address (Other Number and Name)	11	2
18	Address (City)	11	3
19	Address (State or Province)	11	4
20	Address (Zip Code)	11	5
21	Address (Country)	11	6
22	Phone Number (Home)	13	1
23	Phone Number (Business)	14	1
24	Primary Language	15	2
25	Martial Status (A, D, M, S, W)	16	0
26	Religion	17	0
27	Social Security Number	19	0
28	Ethnic Group	22	0
29	Birth Place	23	0

PRB Segment

<u>Field</u>	<u>Value</u>	<u>Field</u>	<u>Component</u>
1	Problem ID	3	1
2	Problem Name	3	2
3	Problem Coding Standard i.e. ICD-9	3	3
4	Problem Established Date	7	0
5	Problem Life Cycle Status (e.g: "Active")	14	2

RXE Segment

<u>Field</u>	<u>Value</u>	<u>Field</u>	<u>Component</u>
1	Medication ID	2	1
2	Medication Name	2	2
3	Medication Coding Standard i.e. NDC	2	3

4	Dose	:	3	0
5	Quantity dispensed	:	4	0
6	Dosage units code i.e. mg	:	5	1
7	Dosage units text i.e. milligrams	:	5	2
8	Provider Dosing Instructions (Text)	:	7	2
9	Prescription Number	:	15	0

AL1 Segment

<u>Field</u>	<u>Value</u>	<u>Field</u>	<u>Component</u>	
1	Allergy ID	:	3	1
2	Allergy Name	:	3	2
3	Allergy Coding Standard	:	3	3
4	Allergy Date	:	6	0

PV1 & PV2 Segments

<u>Field</u>	<u>Value</u>	<u>Field</u>	<u>Component</u>	<u>Segment</u>
1	Patient Class (U,E,I,O,P,R,B)	:	2	0 PV1
2	Visit start date	:	44	0 PV1
3	Visit end date	:	45	0 PV1
4	Diagnosis ID	:	3	1 PV2
5	Diagnosis Name	:	3	2 PV2
6	Diagnosis Coding Standard i.e. ICD-9	:	3	3 PV2

OBR & OBX Segments

<u>Field</u>	<u>Value</u>	<u>Field</u>	<u>Component</u>	<u>Segment</u>
1	Observation Date/Time	:	7	0 OBR
2	Observation Text	:	5	0 OBX
3	Responsible Observer ID	:	16	1 OBX
4	Responsible Observer's Last Name	:	16	2 OBX
5	Responsible Observer's First Name	:	16	3 OBX
6	Responsible Observer's Middle Initial	:	16	4 OBX
7	Responsible Observer's Suffix (e.g: Jr.)	:	16	5 OBX
8	Responsible Observer's Prefix (e.g: Dr.)	:	16	6 OBX
9	Responsible Observer's Degree (e.g: MD)	:	16	7 OBX

Additionally, we had to agree upon the mapping for HL7 queries for information. Query types for Demographics(D) Problems (P), Medications (M), Allergies (A), Visits(V), Notes (N) were defined. Any combinations of these queries can be issued simultaneously. For example, putting the letters PMA in the Query Type segment will result in problems, medications and allergies being returned from the single query.

QRD Segment

<u>Field</u>	<u>Value</u>	<u>Field</u>	<u>Component</u>
1	Date/Time of Query	:	1 0
2	Query Format Code	:	2 0
3	Query Priority	:	3 0
4	Query ID	:	4 0
5	# of records desired	:	7 0
6	Patient Name	:	8 0
7	Subject Filter	:	9 0
8	Department	:	10 0

QRF Segment

<u>Field</u>	<u>Value</u>	<u>Field</u>	<u>Component</u>
1	Query location	:	1 0
2	Query From Date/Time	:	2 0
3	Query Thru Date/Time	:	3 0
4	Query Type (DPMAVN)	:	4 0

Using these HL7 semantic mappings of the CareWeb™ common medical record (patient demographics, medications, allergies, visits and notes) and queries we created site servers which populated these mappings via interfaces to each institution's legacy systems.

4.2.2 Legacy System Interfacing to M

The Beth Israel Deaconess Medical Center East Campus system contains over 20 gigabytes of healthcare information stored in MUMPS data structures. MUMPS (Massachusetts General Hospital Utility Multi-Programming System) provides a rich set of database tools that have made it the foundation of numerous hospital information systems. A major impediment to web-enabling MUMPS data is the difficulty of exporting the data outside of MUMPS. In the past, MUMPS required a dedicated machine and provided both operating system and programming environments. Today, with the advent of Intersystems Open M (ISM), Data Tree Systems M (DTM) and Micronetics M (MSM), which run in standard UNIX and Windows environments, connectivity options are more varied.

All connectivity options share a common structure. M is an interpreted language which creates a proprietary data structure accessible only by M. Interfacing necessitates the creation of a server in M which requires to incoming client requests for information.

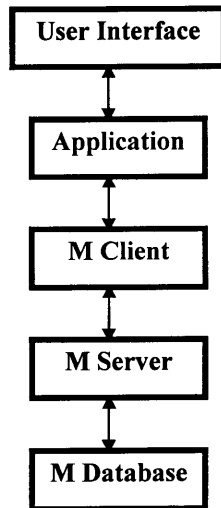


Figure 2 - M Interfacing Architecture

Three models of this client/server architecture are currently available; the Open Database Connectivity (ODBC) model, the Visual M model and the M/Weblink model. Each has advantages and disadvantages with respect to robustness, development time and transportability.

4.2.2.1 Open Database Connectivity

Microsoft created the Open Database Connectivity standard to allow its database tools including Visual Basic, Access, and Query to retrieve data from diverse non-Microsoft databases. ODBC consists of two essential parts, a client and an ODBC driver. The client is used to generate Standard Query Language (SQL) requests from the database. The driver interprets these requests, connects to the database and retrieves the desired information. In the ODBC standard, a specific driver must exist for every type of database the user wants to access. Furthermore, the database may reside on a local computer or a distant computer. The ODBC driver establishes the connection using whatever protocols are necessary.

ODBC connections are inherently slow because the driver must connect to the database, which may entail opening a TCP/IP connection, logging on to the distant system and retrieving information about available data structures.

Access and Visual Basic, for example, issue an ODBC query for every field in the database they are querying. For large databases with hundreds of fields, the amount of network traffic generated by a single ODBC request is significant.

The ODBC model for data access is attractive because it abstracts all interactions with the database into simple SQL commands. Regardless of whether the database is in Sybase, Oracle, or M the ODBC interface is the same. This abstraction makes development using ODBC both straightforward and transportable.

With regard to CareWeb™, using ODBC is especially attractive because no code in the site server requires any knowledge of the underlying database. If the M database changes structure, the ODBC driver handles all changes without having to change the site server. If the M database is replaced by an Oracle database, no site server changes are necessary because the ODBC driver is responsible for all data operations.

The ODBC model for M interfacing appears:

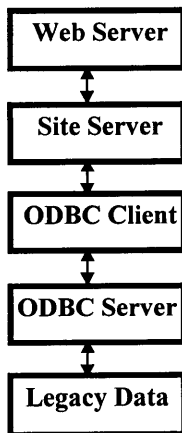


Figure 3 - ODBC Model

In the above model, the hospital web server receives an HL7 request from the CareWeb™ Consolidator. The site server parses the request and formulates an SQL query. The SQL query is passed via ODBC from

the ODBC client to the ODBC server. The legacy data is retrieved and returned to the ODBC client. The data is repackaged into HL7 and returned to the Consolidator via the hospital web server.

Two vendors have implemented ODBC systems for M. The first is Intersystems, the suppliers of Open M/SQL Relational Client/Server. The second vendor is KB Systems, which has created the KB_SQL product.

Intersystems Open M/SQL Relational Client/Server

The Intersystems product was created in 1993. Versions of this product have been evaluated by the Brigham and Women's Information Systems Applied Technologies group and by the Beth Israel Deaconess Center for Clinical Computing. Both groups concluded that the ODBC driver had unacceptably slow response times. Since this initial testing, Intersystems has released several new versions of the ODBC driver which do improve response times. However, testing of version 2.0 M/SQL ODBC client demonstrated performance that is still too slow for a mission critical application. Typical times to connect to M tables varies between ten and fifteen seconds. Typical data retrieval times are shown below with a comparison to Microsoft SQL Server typical retrieval times.

PRODUCT	SELECT 1 RECORD FROM 20000	SELECT 20000 RECORDS
Intersystems M/SQL	15 seconds	240 seconds
Microsoft SQL Server	<1 second	21 seconds

Robustness was evaluated by performing 100 successive queries. 80 queries succeeded, 10 were ignored completely and 10 caused the M server to crash. Intersystems notes that a client request which generates an M error will generally cause the server to shut down. Mission critical applications require that each user's actions should not effect other users. The M/SQL ODBC system does not provide this protection.

Both poor performance and significant instability make the Intersystems M/SQL system unusable for ODBC access. Intersystems notes that another version which corrects these issues will be available.

KB_SQL

KB Systems has produced an ODBC product called KB_SQL which has been selected by the Brigham and Women's Applied Technologies group to be used for all future custom report development. They concluded that it was well engineered, robust and well-supported. Their benchmark times indicated however, that it was not suitable for real-time data retrieval for local clients who desire instantaneous response for queries such as laboratory results. Queries requiring instantaneous response will still be written in M code.

Performance of the KB_SQL product is excellent. Typical times to connect to M tables averages 1 second.. Typical data retrieval times are shown below with a comparison to Microsoft SQL Server typical retrieval times.

PRODUCT	SELECT 1 RECORD FROM 20000	SELECT 20000 RECORDS
KB_SQL	1 seconds	24 seconds
Microsoft SQL Server	<1 second	21 seconds

Robustness was evaluated by performing 100 successive queries. All 100 queries succeeded.

Based on the performance and robustness measures of the KB_SQL product, use of this ODBC solution seems attractive. However, a serious limitation of the ODBC model is that it updates data in M data directly and does not launch "filer" programs which cascade updates to other data fields as necessary. For example, a change to the social security number may require that several other data structures be updated. Without the ability to call filer programs, ODBC solutions may be limited to read-only access of data. Currently CareWeb™ does not change nor update data, but this issue must be considered for future applications.

Thus the ODBC solution with KB_SQL provides a standard, straightforward, transportable model to access M data. It is limited in its ability to update data in that it cannot execute M programs to deal with data changes that must be cascaded to other fields. Performance and robustness are acceptable.

4.2.2.2 Visual M

Visual M was created by Intersystems to allow programmers to use the graphical user interface development tools of Visual Basic while supporting easy access to M data structures. Like other models of M access, Visual M has a client and a server piece.

The Visual M client is a Visual Basic OCX. The server is an M program which accepts client requests. The Visual Basic programming provides the OCX with a segment of M code or the name of an M program to run. The OCX then establishes communication with the server and executes the M code. The result is returned to the OCX and becomes available to the Visual Basic program. The structure appears in Figure 4.

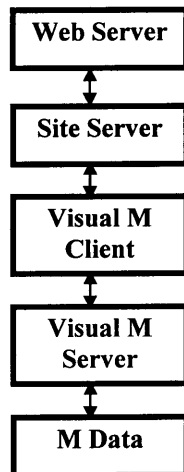


Figure 4 - Visual M Model

In the above model, the hospital web server receives an HL7 request from the Consolidator and this request is passed onto a Visual Basic Program via the Common Gateway Interface (CGI), Information Server Application Program Interface (ISAPI) or Active Server Pages (ASP). The Visual Basic application decodes the HL7 request and selects the name of the M program to run to satisfy the request. The OCX instructs the M server to run the program and the result is returned to the Visual Basic program. The patient information is packaged into an appropriate HL7 format and is returned to the Consolidator.

Performance of the Visual M product is excellent. Typical times to connect to M tables averages 1 second.. Typical data retrieval times are shown below with a comparison to Microsoft SQL Server typical retrieval times.

PRODUCT	SELECT 1 RECORD FROM 20000	SELECT 20000 RECORDS
Visual M	1 seconds	22 seconds
Microsoft SQL Server	<1 second	21 seconds

In six months of evaluation, the Visual M product has proven to be robust. However, this model necessitates programming in M and thus requires more development time than the ODBC model. Whereas the ODBC model took care of all data retrieval automatically, Visual M requires that all data access be expressly coded in M. If the structure of the hospital information system database changes, then the M program must be recoded.

However, Visual M is fast, reliable and offers flexibility beyond ODBC approaches because all aspects of the M globals , M programs and M data filers can be controlled by Visual M.

4.2.2.3 M Web/Link

Intersystems created M/Weblink to provide a robust mechanism for web-exposing M data. The product consists of two parts, an Information Server Application Program Interface (ISAPI) client and a M based server . When called, the ISAPI client launches a telnet session to M, starts up an incidence of M and then executes any program specified. The output of this M program is passed back to the web server. The model appears:

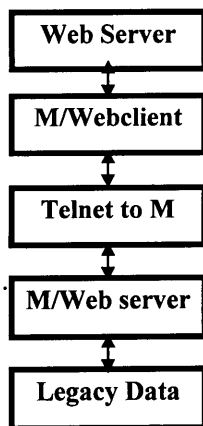


Figure 5 - the M/Weblink model

In the above model, the hospital web server receives an HL7 request from the Consolidator and this request is passed into the M/Weblink Internet Services Application Program Interface (ISAPI) client. The client launches a telnet session to the M machine, starts M and runs the M/Weblink server which runs any M program, passing along the HL7 message as a parameter. This M program is responsible for parsing the HL7, performing legacy system communications and repacking the legacy data into HL7. The patient information sent back to the Consolidator as a web page.

The M/Weblink approach maximizes development time by requiring that all components be written in M. The HL7 parser, legacy system access logic and HL7 message assembly must be done in M. None of these functions is reusable if M is replaced with another legacy system.

Robustness and performance was evaluated by performing 100 success hits to a sample web page. All hits succeeded with less than 1 second response time.

In summary, M/Weblink provides a fast and robust environment for accessing M globals via the World Wide Web. However, all programming and web page development must be done in M, precluding the use of modern web-authoring tools.

4.2.3 Legacy System Interfacing to Relational Databases

The Beth Israel Deaconess Medical Center West Campus stores clinical data in a Sybase clinical data repository (Sybase SQL Server version 4.9.2) running on a Hewlett Packard 9000 Unix System. Sybase supports complete Open Database Connectivity (ODBC) operations and thus does not require a separate product to facilitate data access outside of Sybase.

As an evaluation of Web connectivity options and performance, we created three software systems to web-expose Sybase data. All tests were performed with Microsoft Internet Information Server 3.0 running under Windows NT 4.0 with Service Pack 2, and ODBC 3.0.

The first experiment utilized Active Server Pages with Visual Basic Script and the Microsoft Active Data Objects (ADO) component to display Sybase data as web accessible tables. This implementation creates and resets the ODBC connection every time the page is requested. We created the script Sybase.ASP as follows

```

<HTML>
<HEAD><TITLE>Test ODBC</TITLE></HEAD>
<body BGCOLOR="#FFFFFF" >
<H2>The answer to your request using an ODBC query is </H2>
<HR>
<%
Set Conn = Server.CreateObject("ADODB.Connection")
Conn.Open "careweb","careweb","careweb"
sql="SELECT * from demog"
Set RS = Conn.Execute(sql) %>
<P>
<Table Border=2 width=50%>
<TR><TD WIDTH=25%> Patient ID </TD></TR>
<% Do While not RS.eof%>
    <TR>
    <TD><% = RS("pati_idin") %></TD>
    </TR>
<%
    RS.MoveNext
Loop
RS.close
conn.close
%>
</TABLE>

```

To evaluate both performance and robustness we executed the script 100 times with the following results.

PRODUCT	SELECT 1 RECORD FROM 20000	SELECT 20000 RECORDS
Sybase ODBC called from an ASP Web Page	2 seconds	14 seconds

The second experiment also utilized Active Server Pages with Visual Basic Script and the Microsoft Active Data Objects (ADO) component to display Sybase data as web accessible tables. This implementation created the ODBC connection when the page is first executed and maintained the connection throughout the session. We created the script Sybase.ASP as follows

```

<HTML>
<HEAD><TITLE>Test ODBC</TITLE></HEAD>
<body BGCOLOR="#FFFFFF" >
<H2>The answer to your request using an ODBC query is </H2>
<HR>
<OBJECT RUNAT=Server ID=Conn PROGID="ADODB.Connection"></OBJECT>
<%
Conn.Open Session("ConnectionString")
sql="SELECT * from demog"
Set RS = conn.Execute(sql)
%>
<P>
<Table Border=2 width=50%>
<TR><TD WIDTH=25%> Patient ID </TD></TR>
<% Do While not RS.eof%>
    <TR>
    <TD><% = RS("pati_idin") %></TD>
    </TR>
<%
    RS.MoveNext
Loop
RS.close
conn.close
%>
</TABLE>

Global.ASP
<SCRIPT LANGUAGE=VBScript RUNAT=Server>
Sub Session_OnStart
Session("ConnectionString") = "dsn=careweb;uid=careweb;pwd=careweb"
End Sub
</SCRIPT>

```

To evaluate both performance and robustness we executed the script 100 times with the following results.

PRODUCT	SELECT 1 RECORD FROM 20000	SELECT 20000 RECORDS
ActiveX in ASP first execution	2 seconds	14 seconds
ActiveX in ASP subsequent executions	1 seconds	12 seconds

The third experiment utilized Active Server Pages with an ActiveX component created as a Visual Basic OLE Automation Server to perform data access.. This implementation created the ODBC connection on the first execution and held it open during all future sessions. We created the script Sybase.ASP as follows

```
<%@ LANGUAGE = VBScript %>
The patients in the Sybase table are:
<%
html = sybase.site
%>
<% = html %>
```

```
Global.ASA
<OBJECT RUNAT=SERVER SCOPE=application ID=sybase progid="sybasesite.sybase"></OBJECT>
```

Sybase.VBP - uses Visual Basic Data Object with Read Only True, Snapshot mode

```
Public Function site()
Let site = ""
On Error GoTo site_err
Form1.data1.RecordSource = "Select * from demog"
Form1.data1.Refresh
Do While Form1.data1.Recordset.EOF = False
Let site = site & Form1.data1.Recordset("pati_idin") & vbCrLf
Form1.data1.Recordset.MoveNext
Loop
Exit Function
site_err:
site = Error
End Function
```

To evaluate both performance and robustness we executed the script 100 times with the following results.

PRODUCT	SELECT 1 RECORD FROM 20000	SELECT 20000 RECORDS
ActiveX in ASP first execution	2 seconds	14 seconds
ActiveX in ASP subsequent execution	1 seconds	12 seconds

We conclude from these experiments with Sybase via ODBC provides robust, high performance data access which is not significantly changed by using ODBC at the application level, session level or page level.

The above experiments were repeated with an Oracle version 7.3 database with the following results.

PRODUCT	SELECT 1 RECORD FROM 20000	SELECT 20000 RECORDS
Oracle ODBC called from an ASP Web Page	8 seconds	14 seconds

PRODUCT	SELECT 1 RECORD FROM 20000	SELECT 20000 RECORDS
Session level ODBC in ASP first execution	8 seconds	14 seconds
Session level ODBC in ASP subsequent execution	6 seconds	12 seconds

PRODUCT	SELECT 1 RECORD FROM 20000	SELECT 20000 RECORDS
ActiveX in ASP first execution	8 seconds	14 seconds
ActiveX in ASP subsequent execution	6 seconds	12 seconds

The Oracle experiments demonstrated significantly longer response times despite identical queries on similar machines. We attribute this performance difference to Oracle's ODBC driver implementation.

In summary, ODBC provides robust, modest performance access to non-Microsoft databases. Although ODBC methods are well supported and provide a straightforward path to web-expose data, performance can vary significantly as a consequence of the ODBC driver implementation.

4.2.4 Web technologies

To receive the incoming HL7 requests and to return the HL7 responses populated with legacy data, the site server needs a web interface. Several methods are available to web enable site server applications. The three most commonly used methods are the Common Gateway Interface(CGI), Internet Server Application Programming Interface (ISAPI), and Active Server pages.

4.2.4.1 Common Gateway Interface

The Common Gateway Interface (CGI) is a method for running external programs or gateways under an HTTP server. A CGI program acts as middleware between dynamic information, such as an SQL

database, and the web browser. A server responds to a CGI execution request from a client browser by creating a new process and then passing the data received from the browser through the environment variables and the operating system's standard input method. Results gathered by the CGI application are returned using the operating system's standard output method.

In Visual C and Visual Basic, standard kernel function calls ReadFile and Writefile are used to implement standard in and standard out. The common gateway interface methods, while reliable, have significant performance and scalability problems.

Existing HTTP servers create a separate process for each request received. The more concurrent requests there are, the more concurrent processes are created by the server. However, creating a process for every request is time-consuming and requires large amounts of server RAM. In addition, this can restrict the resources available for sharing from the server application itself, slowing down performance, and increasing wait times on the Web.

As an evaluation of site server functionality using CGI methods, a program was written to return patient demographics stored in an Access database to a web page using the standard site server HL7 messaging methods described above. Ten simultaneous requests for information were issued and then repeated 20 times. Although no instability was noted, 100% of CPU time was used for 3 seconds per request and 54 megabytes of memory were used. These experiments are detailed further in section 6.1.

4.2.4.2 Internet Server Application Program Interface

One way to avoid CGI performance problems is to turn all CGI executables into dynamic linkable libraries (.DLL). The server loads the DLL the first time a request is received and the DLL then stays in memory, ready to service other requests until the server decides it is no longer needed.

In the Microsoft Windows operating system, dynamic linking provides a way for a process to call a function that is not part of its executable code. The executable code for the function is located in a DLL,

which contains one or more functions that are compiled, linked, and stored separately from the processes which use them. There are two methods for calling a function in a DLL.

The first Load-time dynamic linking: This occurs when an application's code makes an explicit call to a DLL function. This type of linking requires that the executable module of the application be built by linking with the DLL's import library, which supplies the information needed to locate the DLL function when the application starts.

The second is Run-time dynamic linking: This occurs when a program uses the LoadLibrary and GetProcAddress functions to retrieve the starting address of a DLL function. This type of linking eliminates the need to link with an import library. When used in the context of a web application, these DLL's are termed Internet Server Application Programming Interface (ISAPI) applications, and are loaded at run time by the HTTP server.

Unlike .EXE type executable files, the ISAPI application DLLs are loaded in the same address space as the HTTP server. This means all the resources are loaded at run time by the HTTP server. There is minimal overhead associated with executing these applications because there is no additional overhead for each request. Benchmarking experiments (described below) show that loading ISAPI application DLLs in the same address space as the server create substantially faster performance than loading them into a new process. In addition, these ISAPI applications scale much better under heavy load.

Since an HTTP server knows the ISAPI application DLLs that are already in memory, it is possible for the server to unload the ISAPI application DLLs that have not been accessed in a configurable amount of time. By preloading an ISAPI application DLL, the server can speed up even the first request for that ISAPI application. In addition, unloading ISAPI application DLLs that have not been used for some time will free up system resources.

Multiple ISAPI application DLLs can reside in the same process as the HTTP server, while the conventional CGI applications run in different processes.

As an evaluation of site server functionality using ISAPI methods, a program was written to return patient demographics stored in an Access database to a web page using the standard site server HL7 messaging methods described above. Ten simultaneous requests for information were issued and then repeated 20 times. Although 100% of CPU time was used, such saturation occurred for less than 100 milliseconds and only 32 Megabytes of memory were used. Substantial increases in performance were observed and ISAPI appeared to be the technology of choice to use for site servers. This experiment is further detailed in section 6.1.

Microsoft introduced Windows NT Service Pack 2 in February of 1997 and this interval upgrade created severe instability in ISAPI programs. Rerunning the experiment above resulted in failure rate of 56% due to errors and incomplete information return. Microsoft's explanation for the new instability was the ISAPI had been superceded by a new technology, Active Server Pages.

4.2.4.3 Active Server pages

Active Server Pages (ASP), is a compile-free application environment which combines HTML pages, scripts, and OLE Automation Servers (ActiveX server components) to create dynamic web content. Active Server Pages provides the flexibility of CGI programs and scripts, without the significant performance tradeoff. Like ISAPI, Active Server Pages runs in-process with the server, is multi-threaded and optimized to handle large numbers of users.

ActiveX server components support the Microsoft Component Object Model (COM), and can provide flexible web-based methods for database and legacy system access.

The Active Database Object (ADO) provides easy access to any OLE/DB or ODBC compatible data source, including Microsoft Access, Microsoft SQL Server, Oracle, Informix, and Sybase. Further, ActiveX technology is compatible with legacy interfacing methods described above (Section 4.2.2). The legacy access business logic is encapsulated in the ActiveX server components facilitating access to virtually any kind of information accessible from the network.

As an evaluation of site server functionality using ASP methods, a program was written to return patient demographics stored in an Access database to a web page using the standard site server HL7 messaging methods described above. 10 simultaneous requests for information were issued and then repeated 20 times. Although 100% of CPU time was used, such saturation occurred for less than 100 milliseconds and only 32 Megabytes of memory were used. Unlike ISAPI, Active Server page requests had a 0% failure rate and numerous attempts to create system instability were unsuccessful.

Given its superior performance, robustness and stability, Active Server pages was chosen as the web interface technology for the CareWeb™ site servers.

4.3 Consolidator

The CareWeb™ consolidator provides the messaging middleware between the health care provider and the site servers. The Consolidator interprets the incoming health care provider request, translates the request into an HL7 query, sends the query via the HTTP post method to each site server, receives the site server response, and consolidates the collected results into a single presentation.

4.3.1 HTTP Messaging

Experience with the site servers demonstrated that Active Server Page technology provided the best foundation for web applications and the Consolidator was written in Visual Basic as an OLE Automation Server called by Active Server Pages. Consolidator HL7 messaging is performed using the same messaging methods described for the site server.

The Consolidator software implementation required an HTTP messaging component to communicate with site servers and retrieve information. Several HTTP ActiveX controls products were evaluated including those from Crescent Technologies, Mabry and Netmanage. The NetManage NEWT ActiveX Control pack provided the most full featured, high performance, robust suite of Internet messaging components. A unique advantage of the NetManage HTTP control is its multithreaded architecture which allows the

Consolidator to launch multiple HTTP requests simultaneously and then monitor their progress to insure timely return of information to the browser as soon as all requests have been answered or a maximum timeout reached..

4.3.2 User Interface

Once site server information is returned to the Consolidator, the Consolidator parses the information and creates a visual presentation layer. For ease of use and clarity of function, the presentation layer must provide an easily navigatable electronic analog to the paper medical record. The final version of the presentation layer contains seven unique screens - demographics, problems, medications, allergies, visits, summary notes and full text notes. A toolbar which is analogous to file folder tabs appears at the top of each screen and provides a consistent navigation paradigm. Each screen is illustrated below in Figures 6 through 12.

Demographics

Site	BIDMC-EAST	BIDMC-WEST
site status	Ok	Ok
first name	FRIEDA	FRIEDA
family name	ALLEN	ALLEN
date of birth	19340501	19340501
gender	F	F
middle name	M	M
previous given name	MARTHA	MARTHA
previous family name	JONES	SMITH
previous middle name	D	D
race	A	A
address	12 MAPLE STREET	101 ELM STREET
	APT 21	APT 14
	NEWTON	BROOKLINE
	MA	MA
	02215	02215
home number	6172222345	6172221432
work number	6176671234	6176671224
marital status	D	D
religion	RC	RC
birth place	DES MOINES	DES MOINES
SSN	512341234	512341234

Figure 6 - Demographics Screen

Problems

Site	Problem Date	Problem Name	Problem Code
BIDMC-EAST	05/01/1980	URTICARIA	708
	10/22/1996	OSTEOPOROSIS	733
	11/01/1996	INSOMNIA	780.52
BIDMC-WEST	05/01/1980	FX DISTAL RADIUS	813.42
	10/22/1996	FX PROXIMAL FEMUR	821

Figure 7 - Problem Screen

Medications

Site	Medication Date	Medication Name	Medication Dose
BIDMC-EAST	01/02/1996	TYLENOL	1 PO Q 4 PRN PAIN
	01/25/1996	IBUPROFEN	1 PO Q 6 PRN PAIN
BIDMC-WEST	10/02/1994	NITRO SL	1 PO Q 5 M x 3
	04/12/1993	PEPCID	1 PO BID

Figure 8 - Medications Screen

Allergies

Site	Allergy Date	Allergy Substance
BIDMC-EAST	05/23/1973	PENICILLIN
	05/23/1996	PEANUTS
BIDMC-WEST	02/25/1983	IODINE CONTRAST
	05/14/1992	TETRACYCLINE

Figure 9 - Allergies Screen

Visits

Site	Visit Date	Visit Reason	Visit DX Code
BIDMC-EAST	01/05/1996	FEMUR FX	821
	05/01/1995	CHEST PAIN	414
BIDMC-WEST	12/05/1996	FX DISTAL RADIUS	813.42
	03/11/1994	DYSPEPSIA	701

Figure 10 - Visits Screen

Notes

Site	Note Date	Note First Line	Note Author
BIDMC-EAST	11/02/1996	Thank you for refering this very interesting patient.	DR JAMES HAYES II MD
BIDMC-WEST	10/14/1995	After a thorough evaluation, we cannot isolate any	DR SUSAN SMITH JR MD

Figure 11 - Notes screen

Note Text

Had telephone conversation with patient regarding continuing care. Patient desires to transfer care to MGH where his cardiologist, Dr. Stone will continue to follow him. Patient has no specific complaints at this time and notes he has been compliant with his coumadin and PT checks. Informed Drs. Roth and Epstein that patient will transfer all care to the MGH

[Go Back to Notes](#)

Figure 12 - Full text notes screen

4.3.3 Browser related issues

Much thought was given to the technologies used in the Visual Presentation layer. Possible browser side visual elements include HTML, Java, and browser side ActiveX components.

The existing infrastructure of the Beth Israel Deaconess Medical Center includes 5000 display devices of which 2000 are personal computers. Of these personal computers, over 50% are 16 bit platforms running windows 3.1 and Mosaic. Requiring a Java or ActiveX compatible browser would require a major infrastructure upgrade. Thus, the presentation layer was implemented entirely in HTML without browser side components, achieving complete browser independence and a low common denominator for browser technology. The Active Server Page technology on the server side uses ActiveX components, but these components return plain HTML to the browser and not browser side software components.

Future releases of CareWeb are likely to allow submission of medical information from the browser into the permanent medical record. These future implementations will likely require browser side components.

5. Security and Confidentiality

In his 1997 state of the Union address, President Clinton noted that “we should connect every hospital to the Internet, so that doctors can instantly share data about their patients with the best specialists in the field.”[25]. The security and confidentiality implications of web-connecting the nation’s clinical data are a major impediment to realizing this noble goal.

In 1995, the National Research Council of the National Academy of Sciences was charged with evaluating the practical measures that can be used to reduce the risk of improper disclosure of confidential health information while providing justified access to those interested in improving the quality and reducing the cost of care. Their March 1997 report, “For the Record: Protecting Electronic Health Information” presents the findings of two years of collaborative investigations and site visits.[26]

The section reviews the public policy context, the internal and external threats to organizations which possess health information and outlines technical and organizational approaches to protecting health information.

We implemented all of the technical recommendations of the report in CareWeb™. Implementing this security model presented many issues, both technical and political.

The NRC recommended eight technical practices for immediate implementation and five for future implementation. We found that these thirteen recommendations provided the foundation on which to design a robust security architecture and chose to implement and evaluate them using CareWeb™.

5.1 The National Research Council Report

The NRC recommendations are separated into two categories, practices for immediate implementation and practices for future implementation. This discrimination is made with the assumption that the immediate practices will suffice as a minimum for the current state of technology. However, as more health information is available in electronic form and as more security technology becomes generally available, the need for more complete security implementations becomes necessary.

Practices for Immediate Implementation

Individual Authentication of Users. The NRC site visits discovered that many health care organizations have generalized login usernames/passwords such as MD for physicians and RN for nurses. To properly authentic individuals on any computer system containing health care data, every individual should have a unique username/password for access. Such a policy allows individuals to be held accountable for all actions taken while logged on.

Access controls. Many healthcare computing systems allow all users to view all information. There is, however, no good reason for a laboratory technician to read the confidential full text data contained in a patient psychiatric profile. Healthcare providers should be allowed to view clinical information on a need to know basis. The most obvious implementation of such controls would be to assign access to different healthcare computing functions based on job role.

Audit trails. Although newspaper articles highlight the threat of computer break-ins by unauthorized “hackers” from outside healthcare organizations, inappropriate healthcare data access from inside the organization is far more common. Normal human curiosity leads individuals not involved in a patient’s care to lookup the records of VIPs, celebrities and fellow employees. If authenticated users are to be held accountable for actions taken while using the health care computing system, retrievable audit trails that log

all accesses to information should be kept. These logs should include time, date, information accessed, and user ID. Audit trails should be available for patient review on demand.

Physical Security and Disaster Recovery. Computer terminals should be positioned where they cannot be accessed by unauthorized users. Unauthorized personnel must be denied access to paper printouts and electronic storage. Backup tapes should be made frequently and tapes should be housed off-site in the case of a physical disaster.

Protection of remote access points. Firewalls should be implemented to provide strong, centralized security and to allow outside access to only those systems critical to outside users. All remote accesses should be protected by single session or encrypted passwords.

Protection of External Electronic Communications. All patient-identifiable data transmitted over public networks should be encrypted.

Software Discipline. Virus checking programs should be installed on all servers and downloads from the internet to servers should be limited.

System Assessment. Audits should be performed on a monthly basis to examine vulnerability to password cracking programs and to verify procedures implemented to detect system vulnerabilities.

Practices for Future Implementation

Strong Authentication. Healthcare providers occasionally share usernames/passwords or write them down near a computer terminal. Such practices defeat the authentication, access controls and audit trails offered by unique usernames/passwords. Authentication is substantially strengthened by requiring that logon be paired with physical possession of “hardware tokens” such as smartcards, magnetic strip IDs or devices with constantly changing passwords..

Enterprise-wide Authentication. Healthcare environments typically have many heterogeneous computing systems. If users are forced to have different logon information for each computer system, remembering such information is an inconvenience and users will have a tendency to write down login information. To minimize such behavior, users should authenticate once and then have access to all relevant systems.

Access Validation. In the simplest form of access control, different system functions are available based on job role. A more sophisticated implementation would tailor content within functions by job role. For example, a discharge summary could be viewed by both a physician and a billing coder, but details of the patient's psychiatric evaluation would not appear for the coder.

Expanded Audit Trails. Simple audit trails capture information at a single organization. An expanded auditing system would provide inter-organizational audit trails that trace information as it passes through the health care complex .

Electronic Authentication of Records. Electronic signature should be used to "sign" submitted medical records and cryptographic digital signature should be used when retrieving records to insure that records are not modified during the transmission process.

Security Architecture

To implement the NRC recommendations, we reduced the present and future recommendations to eight action areas - Strong enterprise-wide authentication, access validation, expanded audit trails, encryption of public network transmissions, electronic authentication of records, physical security, disaster recovery, software discipline, and system assessment.

5.2 Enterprise-wide Authentication

We guarantee the authenticity of each user with Security Dynamics SecurID hardware tokens. These tokens are small, handheld devices containing microprocessors that calculate and display unpredictable codes. These codes change at a specified interval, typically 60 seconds. Our implementation requires that

each user accessing CareWeb™ begin a session by entering a username, a memorized personal identification number (PIN) and the currently displayed password from the SecurID device. This information is transmitted to a security server which authenticates the user and verifies that the correct password was entered. The security server compares the user-entered password with its knowledge of what password should have been entered for that 60 second period. If the password does not match, it also checks the password from the previous 60 second period to account for delays in typing and transmission. Once a password is verified, the user is authenticated for the entire enterprise for the duration of the web session or 15 minutes, whichever is less. An encrypted security “cookie” is sent back to the user’s browser and this cookie is automatically used for all future security dialogs. Using Visual Basic Script and Microsoft’s Active Server Pages, we dynamically decrypt the cookie within the web server and invisibly re-verify authentication before responding to additional requests for healthcare data. The code to perform this on the fly decryption appears:

```

<%@ LANGUAGE = VBScript %>
<% function getuser()
getuser = ""
dim temp(4)
dim base2()
cookies = request.servervariables("HTTP_COOKIE") & ";"
webidbegin = instr(cookies,"webid=")
webidoffset = 6
if webidbegin = 0 then
webidbegin = instr(cookies,"webid_dom=")
webidoffset = 10
end if
if webidbegin=0 then
getuser="Username not found"
exit function
end if
webidbegin = webidbegin + webidoffset
webidend = instr(webidbegin,cookies,";")
if webidend=0 then
getuser="Username not found"
exit function
end if
base64 = urldecode(mid(cookies,webidbegin,webidend-webidbegin))
base2len = int(len(base64)/4)*3 - 1
reDim base2(base2len)
For i = 0 To int(Len(base64) / 4) - 1
For j = 0 To 3
temp(j) = Asc(Mid(base64, i * 4 + j + 1, 1)) - 32
Next
base2(i * 3) = shiftleft(temp(0), 2) Or (shiftright(temp(1), 4) And &H3)

```

```

base2(i * 3 + 1) = shiftright(temp(1), 4) Or (shiftright(temp(2), 2) And &HF)
base2(i * 3 + 2) = shiftright(temp(2), 6) Or temp(3)
Next
For i = 16 To base2len
if base2(i)=0 then exit for
  getuser = getuser + Chr(base2(i))
Next
end function
Function UrlDecode(sEncoded)
pointer = 1
Do
  pos = InStr(pointer, sEncoded, "%")
  If pos = 0 Then Exit Do
  sEncoded=Left(sEncoded, pos-1 ) &Chr("&H" & (Mid(sEncoded, pos + 1, 2)))&Mid(sEncoded, pos+3)
  pointer = pos + 1
Loop
UrlDecode = sEncoded
end function
Function shiftright(inbyte , shift )
shiftright = Int(inbyte / (2 ^ shift))
End Function
Function shiftright(inbyte , shift )
shiftright = inbyte * (2 ^ shift) Mod 256
End Function %>

```

If the SecurID token is lost or stolen, it can be immediately deactivated for the entire enterprise by disabling it at the security server.

5.3 Access Validation

In addition to storing encrypted username and password information, the security cookie contains the job role of the user. Displays of healthcare information are generated dynamically by Active Server page scripts which assemble the multi-institutional medical record. The scripts can tailor delivered health care information based on the job role indicated by the cookie. In our proof-of-concept implementation, we have restricted this tailoring of access to specific areas of the medical record such as discharge summaries. We have not created a facility to scan for and restrict specific content within an area, such as removing a psychiatric evaluation from a discharge summary.

5.4 Multi-organizational Audit Trails

It has been the security policy of the Beth Israel hospital to provide auditing at the level of the specific patient queried and the individual menu selections used [27]. CareWeb™ implements a complete multi-organizational audit trail.

In any multi-institutional architecture there are two places to capture the audit - either at the institutional level where the information is stored (the sites) or at the point where the information is delivered (the CareWeb™ “Consolidator”). We elected to capture the information at the site level. Although only a single CareWeb™ “Consolidator” exists today, CareWeb™ could be expanded such that other regional or national “Consolidators” might query information from the CareGroup institutions. If the audit was captured at the “Consolidator” level, each institution would have to rely on the security practices of the “Consolidator” operators. By storing audit trails at each site, each hospital can control and audit the information that leaves its site, regardless of where it is delivered. Each hospital site server captures patient identification information, requester, the requester’s IP address, date, time, and information requested.

Although information is stored at the site level we have implemented a multi-institutional auditing system that provides patients with the details of the movement of their medical information throughout the healthcare enterprise. The auditing query system has the same hardware token authentication and access controls as are required for any CareWeb™ healthcare data request. Once authenticated, an auditor enters patient identification information and submits the information to an “Auditing Consolidator”. This “Auditing Consolidator” uses secure, password protected Open Database Connectivity (ODBC) connections to query the audit trails of the individual hospitals. It produces a consolidated report showing all flows of information about the patient for all institutions.

5.5 Protection of External Communications

The existing legacy systems at the Beth Israel and Deaconess hospitals employ a complex series of hardware controls which limit internet transactions from outside the institution. Using routers and

firewalls, network administrators limit legacy system access to hardware devices physically located within the campus.

To create security between a browser running on a user's desktop and the Consolidator web server, we implemented the Netscape standard Secure Sockets Layer [28]. The SecurID username and passcode are only exchanged after an encrypted connection has been established by the Secure Sockets Layer.

5.6 Encryption of public network transmission

For communications between the Consolidator and site servers, we implemented RSA public key encryption for key exchange, session key cryptography for data exchange, and digital signature for authentication of the Consolidator and site servers [29]. This is shown in Figure 13.

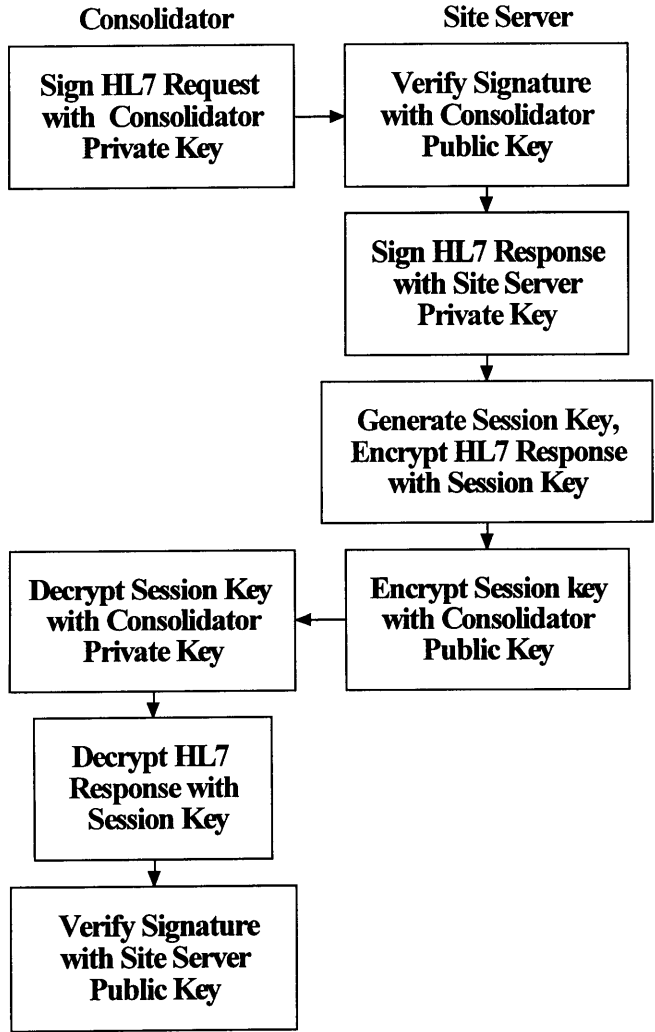


Figure 13 - Encryption of Network Transmissions

Each Consolidator HL7 request is signed with the Consolidator's RSA private key. The request is sent to the site server which uses the Consolidator's public key to validate the digital signature through standard hashing and signature verification methods. The site server retrieves the information requested and signs the HL7 response with its private key. The site server then generates a session key which it uses to encrypt the HL7 response. The session key is then encrypted using the Consolidator's public key. The encrypted session key and encrypted data are sent back to the Consolidator. The session key is decrypted using the Consolidator's private key. The encrypted HL7 response is decrypted using the decrypted session key. Finally, the HL7 response is validated using the site server's public key. All decrypted site server messages are consolidated into a single web page and returned to the original requesting browser over the Secure Sockets Layer.

5.7 Electronic authentication of records

The use of hardware tokens for system access also facilitates electronic signature. Since possession of the hardware device authenticates the user, the SecurID token is used as the official electronic signature for "signing" all CareWeb™ documents and audit trails.

As noted above, digital signature cryptography methods are used for all network transmissions, ensuring the integrity of all health data delivered. The NRC recommends an implementation of hashing and digital signature to insure that medical records are not changed on the individual systems where they are stored. In the CareWeb™ architecture we have no control of the integrity of the data stored at each institution. We have created a secure mechanism to transport each institution's data and can guarantee that the data was not changed during the retrieval process. The reputability of the data is dictated by security policies of each institution providing the data.

5.8 Physical security and disaster recovery

The notion of a multi-institutional architecture provides significant physical protection for health data. Instead of physically locating all patient records in a central data source which is vulnerable to physical disasters, the CareWeb™ architecture depends upon the consolidator which stores no health care information. All that is needed to restore a physically destroyed “Consolidator” system is to connect another computer containing the “Consolidator” software and its required cryptographic keys to the hospital network. Currently, all site servers are geographically dispersed and are locked in secure computer rooms accessed by electronic keycode. In the CareWeb™ architecture we have no control of the physical security and disaster recovery practices of the individual sites which provide data. However, if any sites sustain a disaster and cease to provide data, the Consolidator notes that a site is currently unavailable and provides a virtual medical record comprised of all functioning sites.

5.9 Software discipline

No browser software is installed on either the site servers or the Consolidator machines, precluding inappropriate downloads. Virus checking programs are in place on all CareWeb™ systems and are executed daily by a system daemon.

On the end-user workstation, we have been careful not to cache pages returned by the Consolidator. In our laboratory environment we have verified that neither Netscape nor Internet Explorer cache pages that have been returned via a secure socket connection such as that used by CareWeb™. We cannot protect against an authenticated user who installs a new type of browser that does cache secure pages. However, all pages returned by the Consolidator have an HTML header which indicates that they expire on delivery. Even if a new browser was installed which cached information, this expiration forces the browser to replace each cached page as new requests for information are made, minimizing the amount of information that is stored on the end-user workstation.

5.10 System assessment

Daily assessment is performed on both the Consolidator and site server systems. On the Consolidator, a security log lists all SecurID tokens used, all failed login attempts, and all changes made to the token database. Web server log analysis (WebTrends) shows all attempts to contact the Consolidator web server showing IP address, time, date and page accessed. System assessments are also performed on a daily basis at each institutional site, per their own institutional guidelines.

6. Evaluation

The evaluation of CareWeb™ demonstrates that its implementation is robust, scalable and provides rapid response times.

6.1 Technical Evaluation

Performance and scalability are significant issues in any client/server architecture. To investigate performance and scalability of the CareWeb™ architecture, we ran several experiments in the Windows NT 4.0/Service Pack 2 environment on an HP Vectra XU Pentium 150 with 32 Megabytes of RAM and a 3 gigabyte SCSI hard disk.

As discussed in Section 4.2.4, numerous web technologies are available to create interactive web content. The most commonly used, the standard Common Gateway Interface or CGI is a robust, proven technology. However, each time a CGI is called, a new process is started which requires its own memory space and CPU time. This can be a problem if starting a process requires significant overhead. Since the CareWeb™ software is complex and contains several ActiveX components, 3-4 Megabytes of memory are required per process and startup requires significant CPU resources. As a test of the resources required by CGI's, we built a web page consisting of 10 frames, each of which simultaneously launches a consolidator which requests information from one site server. Charting CPU usage and memory requirements, we

obtained the following: following:

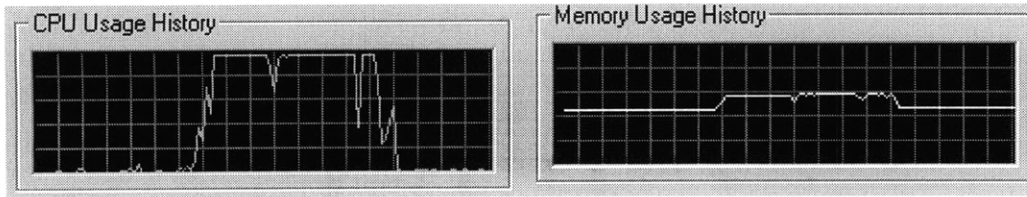


Figure 14 - CGI Performance (x range from 0 to 95 seconds, y range from 0% to 100% usage)

The above demonstrates that calling multiple CGI's simultaneously pegs the CPU at 100% usage for approximately 30 seconds. Further, additional memory is required by each CGI request. The end result is that as more CGI processes are called, performance becomes increasingly poor and with 10 simultaneous consolidator requests launching 20 site server requests all on the same machine, response was so poor that every consolidator query timed out after 15 seconds and returned no information. We conclude from the above experiments that a CGI based CareWeb™ architecture is not scalable.

The basic problem with CGI's is that they are out of process servers launched each time a request is made. Microsoft has implemented Active Server Pages technologies that create In Process servers residing in the same memory space as the web server itself. These servers are ActiveX components which process requests by executing a new thread per request. The result is that very little additional memory and CPU time is needed per request and program load time is eliminated.

As a test of the resources required by Active Server Pages, we built a web page consisting of 10 frames each of which simultaneously launches a consolidator which requests information from one site server.

Charting CPU usage and memory requirements, we obtained the following:

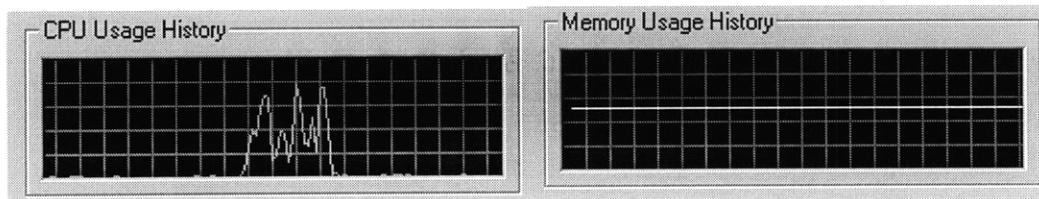


Figure 15 - Active Server Page evaluation (x range from 0 to 95 seconds, y range from 0% to 100%)

The above demonstrates that calling multiple ASP's simultaneously transiently increases CPU use to 80% of capacity for a very short time. Memory requirements are not increased as the application is permanently

integrated into the web server. Response time for 10 simultaneous consolidator requests via ASP is under 20 seconds. We conclude from the above experiments that Active Server Pages are both robust and scalable.

6.2 User Evaluation

As an early evaluation of CareWeb™, we sought and received Institutional Review Board (IRB) approval to web-expose selected medical records from actual patients who have records at more than one CareGroup institution. Patient approval was obtained and patients were allowed to view the CareWeb™ versions of their medical record before making them generally available. Furthermore actual patient names and addresses were pseudonymized, but medical information was not altered.

The system was evaluated by 25 healthcare providers, chosen at random from both institutions, who assessed CareWeb’s ease of use, response times and utility in patient care. Further evaluation was performed by 25 information systems staff members who evaluated CareWeb’s robustness, security and potential for deployment in the live environment. The cross-institutional medical record was composed of patient demographics, problems, medications, visits, allergies, and notes for a selection of sample, fictional patients. During the evaluation period, the CareWeb™ system processed 3000 accesses for patient information. To document clinician, information systems staff and patients opinions about CareWeb™ technologies, we performed several surveys.

6.2.1 Healthcare Providers

Twenty-five members of the clinician staff of the Beth Israel and Deaconess hospitals were selected

CareWeb Evaluation Data Form					
Type: Healthcare provider					
Date:					
Department: Medicine Emergency Medicine Surgery					
CareWeb information is useful	Strongly Agree	Agree	Indifferent	Disagree	Strongly Disagree
CareWeb is easy to use	Strongly Agree	Agree	Indifferent	Disagree	Strongly Disagree
CaseWeb is fast	Strongly Agree	Agree	Indifferent	Disagree	Strongly Disagree
CareWeb protects confidentiality	Strongly Agree	Agree	Indifferent	Disagree	Strongly Disagree

23/25 healthcare providers strongly agreed that CareWeb information was useful. 25/25 strongly agreed that CareWeb was easy to use and 25/25 strongly agreed that CareWeb was fast. 23/25 strongly agreed that CareWeb protects confidentiality.

6.2.2 Information Systems professionals

Twenty-five members of the Information Systems staff of the Beth Israel and Deaconess hospitals were selected randomly and surveyed as to their perception of the robustness of the security methods implemented. The survey appears as follows:

CareWeb Evaluation Data Form					
Type:	IS				
Date:					
CareWeb is easy to use	Strongly Agree	Agree	Indifferent	Disagree	Strongly Disagree
CareWeb should be deployed	Strongly Agree	Agree	Indifferent	Disagree	Strongly Disagree
CareWeb is robust	Strongly Agree	Agree	Indifferent	Disagree	Strongly Disagree
CareWeb is secure	Strongly Agree	Agree	Indifferent	Disagree	Strongly Care

25/25 information systems staff members strongly agreed that CareWeb was easy to use. 24/25 strongly agreed that CareWeb should be deployed. 23/25 strongly agreed that CareWeb was robust. 25/25 strongly agreed that CareWeb is secure.

6.3 Future evaluation

To date, the evaluation of CareWeb™ has been limited to the proof-of-concept system. Over the next six months, CareWeb™ will be deployed in the live environment of six CareGroup Emergency Departments. An expanded evaluation of the deployed version will include detailed feedback from the daily users of the system. To accomplish this evaluation, Emergency Department attending physicians will be given SecurID hardware tokens to access this system. The evaluation period will be 90 days.

This future evaluation will examine and evaluate the implementation of CareWeb in the East and West campus emergency departments. The study will focus on how clinicians integrate the electronic medical records into their daily work practices and collaboration with colleagues. In particular, we will investigate how the systems mediate the interactions and relations across the occupational groups, work teams, and geographically dispersed communities found in each emergency room and between the two hospitals. The study has both practical and theoretical implications. Practically, the data gathered can facilitate the further development and implementation of CareWeb. Theoretically the study enhances medical record research and computer supported collaborative work research which have primarily centered around the use of information technology within one but rarely across two occupational groups

The study will be a longitudinal ethnographic study which involves data gathering before, during, and after the implementation of CareWeb. The study will focus on both the groups directly interacting with the medical record systems as well as communities in the two hospitals indirectly effected by the transition from one system to another. Through the use of participant observation in the two emergency rooms, interviews, and review of textual data sources we will create a detailed account of the ways in which existing medical records (such as OMR in Beth Israel) and the new system (CareWeb) support the work practices of and relations among the different occupational groups and teams. We will highlight the similarities and differences between the two settings. The resulting data will consist of field notes, audio tapes of interactions, interview transcripts, and excerpts from various electronic and paper based documents such as medical records, emails, organizational charts or meeting protocols. More formal work flow analysis will also be considered if appropriate.

7 Future Work and new applications

Several projects are currently under which utilize the architectures and software tools developed in the CareWeb™ project.

7.1 Master Patient Index Problem

The CareWeb™ project implemented a master patient index strategy described in Section 3.2 which utilized the patient name, gender, date of birth and optionally the mother's/father's first name. Although this scheme was face validated and informally evaluated during the testing of CareWeb™, no formal investigation of this strategy has been performed. We are currently engaged in a collaboration to formally evaluate master patient index strategies between the databases of the Beth Israel Deaconess Medical Center and Brigham and Women's Hospital.

In this study, we will formally test the hypothesis that a combination of patient name, date of birth, gender and other relevant data elements will identify patients uniquely enough to enable sharing of electronic patient records between the emergency departments of the Beth Israel Deaconess Medical Center and the Brigham and Women's Hospital.

This project will produce statistics reflecting the success of correctly matching patients in the databases of two hospitals and will not exchange actual patient data. By limiting the initial study in this manner, we minimize the security/confidentiality, organizational and political issues that would be created by sharing clinical information between different hospitals. To perform the study we will:

1. Obtain IRB approval for this study from both the Beth Israel Deaconess Medical Center and the Brigham and Women's Hospital .
2. Design software to be installed at both the Beth Israel Deaconess Medical Center and the Brigham and Women's Hospital that when given a patient name, date of birth, gender, and other relevant data

elements will indicate that a single match is found, that multiple matches are found or that no matches are found.

3. Choose 1000 patients at random from the emergency department logs of the Beth Israel Deaconess Medical Center and use the software to analyze them for matches against the entire Brigham and Women's database.
4. Choose 1000 patients at random from the emergency department logs of the Brigham and Women's Hospital and use the software to analyze them for matches against the entire Beth Israel database.
5. Obtain a count of true positives and false positives by verifying all automated matches against manual lookups done in the native legacy systems.
6. Obtain a count of true negatives and false negatives by verifying all automated non-matches against manual lookups done in the native legacy systems.
7. Compute the average number of patient crossovers between the two institutions to verify that sharing emergency clinical data between the two institutions would benefit a significant number of patients.
8. Compute the sensitivity and specificity of our automated methods to uniquely match patients between the two institutions.

7.2 Multimedia extensions

The initial implementation of CareWeb exchanges text data from two legacy systems. However, other information systems at the Beth Israel Deaconess Medical Center contain repositories of multimedia information. To extend the CareWeb architecture, we wish to test the hypothesis that providing a multimedia electronic patient record through CareWeb will improve the quality of emergency care.

We believe that providing immediate on-line access to clinical multimedia will greatly enhance the effectiveness of computer systems for emergency clinicians. The amount of information available from multimedia sources for decision making far exceeds text-based representations of the same data [30]. We will incorporate multimedia objects such as full-motion video of patients, clinical audio such as heart sounds, and graphical medical information such as EKGs and radiology studies, and will make these objects available electronically in the emergency room. To accomplish this, we will

1. Bring a multimedia electronic patient record system into use in the Emergency Department of Beth Israel Deaconess Medical Center.
2. Design an architecture that allows emergency clinical data to be updated through the World Wide Web and shared among geographically dispersed emergency treatment areas.
3. Implement a data base of emergency medical record data and integrate it with the hospital's existing electronic patient record.
4. Integrate multimedia objects into the system, including medical images, full-motion video of patients, and clinical audio (heart sounds, patient speech)
5. Design and implement security and confidentiality methods for distributed access to clinical data, to achieve an appropriate balance between patient confidentiality and the need of medical practitioners to have immediate access to medical data in emergency situations.
6. Develop metrics to assess the use and usefulness of multimedia electronic patient records for emergency services.

7.3 Vocabulary Server

Working closely with Lexical Technologies Inc., we are investigating the use of the MetaThesaurus to unify the cross institutional content of the CareWeb virtual medical record. A thesaurus server capability will be implemented that enables users to retrieve terms and concepts that match an input string, definitions of terms, and terms with related meanings when creating the CareWeb virtual medical record.

As an important example of a clinically relevant problem domain, the CareWeb thesaurus server project will focus on choosing problem names to describe clinical conditions. The problem list, the set of clinically relevant conditions that describe a patient, is at the core of a medical record. Treatment guidelines, clinical staffing decisions, and under diagnosis-based managed care, patient billing, are all driven by the problem list. Accurate definition of problems is essential to hospital-based outcomes analysis, to continuous quality improvement programs, and to public health assessments of the frequency of occurrence of disease and the efficacy of treatment approaches.

The thesaurus server will be implemented as a CareWeb-compliant middleware component designed to help caregivers conceptualize problem list entries, and to support enterprise use of problem lists to trigger alerts and guidelines, capture indications, aggregate patients, and preserve the categorization of patients across changes in terminologies.

Problems such as “cubital tunnel syndrome,” “brain metastases of melanoma,” or “psychoses associated with temporal lobe epilepsy” are an effective way for caregivers to describe their patients even though there are variations in the meaning and purpose of problem lists from institution to institution. These descriptions remain useful whether they are re-utilized later by the caregiver who created them, or utilized by other caregivers during collateral or follow-up care.

Nevertheless, problems in a problem list in a CareWeb Virtual Medical Record are difficult to manipulate computationally. Specifically, they are non-comparable: the current version of CareWeb is unable to determine when two different descriptors describe the same or a related problem.

One way to increase the comparability of clinical problem lists is to provide an easy and reliable means for caregivers to convert relatively casual entries - that are easy to remember, type, and spell - into authoritative entries in one or more reference terminologies. For example, “CHF” can be converted to “Congestive heart failure,” “mad cow disease” to “Bovine spongiform encephalopathy,” and “amenorhea” into “Amenorrhea”. However else they may differ, most enterprise electronic medical records support the creation of caregiver-entered problem lists. CareWeb compliant reusable middleware components may be readily applied to increase problem list formalization within institutions.

The thesaurus server will be designed to offer a focused, ordered list of suggestions as quickly as possible. Specifically, it will provide a function, accessible through the world-wide web, to accept a string as input and to return suggested terms. This function will support word completion, correction of simple spelling errors and semantic navigation.

The thesaurus server will make use of the current version of the Unified Medical Language System Metathesaurus, enhanced with the local problem dictionaries. The Metathesaurus establishes a set of clinically-relevant concepts, and for each concept provides a concept unique identifier (CUI), a semantic category, and relationships to one or more other concepts.

7.4 Bili Light

In the Boston metropolitan area, it is common for a baby to be born at one hospital such as the the Beth Israel Deaconess Medical Center or the Brigham and Women’s Hospital, but to seek followup or emergency care at Children’s Hospital of Boston. On presentation to the emergency department at Children’s, the evaluating care providers do not have the benefit of the child’s birth records, neonatal testing or prenatal evaluation.

The Bili Light project is a collaboration with Childen’s Hospital of Boston. Working closely with researchers at Children’s Hospital, we will deploy a proof of concept pilot which provides access to

prenatal, maternal and neonatal data for patient clinical care, protects the confidentiality and privacy of all patient information and uses the CareWeb, W3EMRS and Bili Light uniform standards for secure electronic data exchange

Two types of patient lookups will be supported by the Bili Light Site Server.

The first is the Identification dialog. The Bili Light consolidator at Children's Hospital requests information by sending Beth Israel Deaconess Medical Center an HL7 message specifying the mother's name, the mother's date of birth and the child's date and time of birth. The Bili Light site server returns combinations of mothers and their children that meet or approximate these criteria.

The second is an information dialog which queries for all laboratory results from both mother and child including blood type, Rh, coombs testing, and complete blood count information.

We anticipate that providing such information to caregivers at Children's Hospital will both reduce costs and enhance the quality of patient care.

7.5 Affiliated Health Networks of New England

CareWeb was designed to unite the clinical data of the Beth Israel Deaconess Medical Center East and West Campus. As a further evaluation of the scalability of the architecture, we wish to extend the data exchange from a single integrated healthcare delivery system to a regional collaborative of health care providers, insurers and government agencies.

The Massachusetts Health Data consortium created the Affiliated Health Networks of New England project as an organizational model to facilitate such information exchange. Working with the Affiliated Networks, we have proposed a pilot project to use CareWeb to provide appropriate access to an interconnected infrastructure of fiscal, clinical and outcomes databases for both public and private sector constituencies throughout New England.

This project will extend the CareWeb common medical record from its existing Demographics, Problems, Medications, Allergies, Visits and Notes to include the 156 data elements of the Center for Disease Control's DEEDS dataset.

7.51 DEEDS common medical record

The DEEDS dataset was created to foster the collection and distribution of uniform emergency data. The elements in the DEEDS dataset are as follows:

Section 1 - Patient Identification Data

- | | |
|-----------------------|---|
| 1.01 Internal ID | 1.10 Account Number |
| 1.02 Name | 1.11 Social Security Number |
| 1.03 Alias | 1.12 Occupation |
| 1.04 Date of Birth | 1.13 Industry |
| 1.05 Sex | 1.14 Emergency Contact Name |
| 1.06 Race | 1.15 Emergency Contact Address |
| 1.07 Ethnicity | 1.16 Emergency Contact Telephone Number |
| 1.08 Address | 1.17 Emergency Contact Relationship |
| 1.09 Telephone Number | |

Section 2 - Facility and Practitioner Identification Data

- | | |
|--|---|
| 2.01 ED Facility ID | 2.09 ED Practitioner Type |
| 2.02 Primary Practitioner Name | 2.10 ED Practitioner Current Role |
| 2.03 Primary Practitioner ID | 2.11 ED Consultant Practitioner ID |
| 2.04 Primary Practitioner Type | 2.12 ED Consultant Practitioner Type |
| 2.05 Primary Practitioner Address | 2.13 Date/Time ED Consult Request Initiated |
| 2.06 Primary Practitioner Telephone Number | 2.14 Date/Time ED Consult Starts |
| 2.07 Primary Practitioner Organization | |
| 2.08 ED Practitioner ID | |

Section 3 - ED Payment Data

- | | |
|---|---|
| 3.01 Insurance Coverage or Other Expected Source of Payment | 3.10 Date/Time of ED Payment Authorization Decision |
| 3.02 Insurance Company | 3.11 Entity Contacted to Authorize ED Payment |
| 3.03 Insurance Company Address | 3.12 ED Payment Authorization Code |
| 3.04 Insurance Plan Type | 3.13 Person Contacted to Authorize ED Payment |
| 3.05 Insurance Policy ID | 3.14 Telephone Number of Entity or Person Contacted to Authorize ED Payment |
| 3.06 ED Payment Authorization Requirement | 3.15 Total ED Facility Charges |
| 3.07 Status of ED Payment Authorization Attempt | 3.16 Total ED Professional Fees |
| 3.08 Date/Time of ED Payment Authorization Attempt | |
| 3.09 ED Payment Authorization Decision | |

Section 4- ED Arrival and First Assessment Data

4.01	Date/Time First Documented in ED	4.16	First ED Glasgow Motor Component Assessment
4.02	Mode of Transport to ED	4.17	Date/Time of First ED Glasgow Coma Scale Assessment
4.03	EMS Unit that Transported ED Patient	4.18	First ED Systolic Blood Pressure
4.04	EMS Agency that Transported ED Patient	4.19	Date/Time of First ED Systolic Blood Pressure
4.05	Source of Referral to ED	4.20	First ED Diastolic Blood Pressure
4.06	Chief Complaint	4.21	First ED Heart Rate
4.07	Initial Encounter for Current Instance of Chief Complaint	4.22	First ED Heart Rate Method
4.08	First ED Acuity Assessment	4.23	Date/Time of First ED Heart Rate
4.09	Date/Time of First ED Acuity Assessment	4.24	First ED Respiratory Rate
4.10	First ED Acuity Assessment Practitioner ID	4.25	Date/Time of First ED Respiratory Rate
4.11	First ED Acuity Assessment Practitioner Type	4.26	First ED Temperature Reading
4.12	First ED Responsiveness Assessment	4.27	First ED Temperature Reading Route
4.13	Date/Time of First ED Responsiveness Assessment	4.28	Date/Time of First ED Temperature Reading
4.14	First ED Glasgow Eye Opening Component Assessment	4.29	Measured Weight in ED
4.15	First ED Glasgow Verbal Component Assessment	4.30	Pregnancy Status Reported in ED
		4.31	Date of Last Tetanus Immunization
		4.32	Medication Allergy Reported in ED

Section 5 - ED History and Physical Examination Data

5.01	Date/Time of First ED Practitioner Evaluation	5.11	Current Therapeutic Medication Ut
5.02	Date/Time of Illness or Injury Onset	5.12	Current Therapeutic Medication Schedule
5.03	Injury Incident Description	5.13	Current Therapeutic Medication Rt
5.04	Coded Cause of Injury	5.14	ED Clinical Finding Type
5.05	Injury Incident Location Type	5.15	ED Clinical Finding
5.06	Injury Activity	5.16	Date/Time ED Clinical Finding
5.07	Injury Intent	5.17	ED Clinical Finding Practitioner ID
5.08	Safety Equipment Use	5.18	ED Clinical Finding Practitioner Type
5.09	Current Therapeutic Medication	5.19	ED Clinical Finding Data Source
5.10	Current Therapeutic Medication Dose		

Section 6 - ED Procedure and Result Data

6.01	ED Procedure Indication	6.07	ED Procedure Practitioner Type
6.02	ED Procedure	6.08	Date/Time ED Procedure Result Reported
6.03	Date/Time ED Procedure Ordered	6.09	ED Procedure Result Type
6.04	Date/Time ED Procedure Starts	6.10	ED Procedure Result
6.05	Date/Time ED Procedure Ends		
6.06	ED Procedure Practitioner ID		

Section 7 - ED Medication Data

7.01	Date/Time ED Medication Ordered	7.08	ED Medication Route
7.02	ED Medication Ordering Practitioner ID	7.09	Date/Time ED Medication Starts
7.03	ED Medication Ordering Practitioner Type	7.10	Date/Time ED Medication Stops
7.04	ED Medication	7.11	ED Medication Administering Practitioner ID
7.05	ED Medication Dose	7.12	ED Medication Administering Practitioner Type
7.06	ED Medication Dose Units		
7.07	ED Medication Schedule		

Section 8 - ED Disposition and Diagnosis Data

8.01	Date/Time of Recorded ED Disposition	8.21	Amount of ED Discharge Medication to be Dispensed
8.02	ED Disposition	8.22	Number of ED Discharge Medication Refills
8.03	Inpatient Practitioner ID	8.23	ED Disposition Diagnosis
8.04	Inpatient Practitioner Type	8.24	ED Disposition Diagnosis Code
8.05	Facility Receiving ED Patient	8.25	ED Disposition Diagnosis Practitioner ID
8.06	Date/Time Patient Departs ED	8.26	ED Disposition Diagnosis Practitioner Type
8.07	ED Follow-Up Care Assistance	8.27	ED Service Level
8.08	Referral at ED Disposition	8.28	ED Service Level Practitioner ID
8.09	ED Referral Practitioner Name	8.29	ED Service Level Practitioner Type
8.10	ED Referral Practitioner ID	8.30	Patient Problem Assessed in ED Outcome Observation
8.11	ED Referral Practitioner Type	8.31	ED Outcome Observation
8.12	ED Referral Organization	8.32	Date/Time of ED Outcome
8.13	ED Discharge Medication Order Type	8.33	ED Outcome Observation
8.14	ED Discharge Medication Ordering Practitioner ID	8.34	ED Outcome Observation Practitioner
8.15	ED Discharge Medication Ordering Practitioner Type	8.35	ED Patient Satisfaction Report Type
8.16	ED Discharge Medication	8.36	ED Patient Satisfaction Report
8.17	ED Discharge Medication Dose		
8.18	ED Discharge Medication Dose Units		
8.19	ED Discharge Medication Schedule		
8.20	ED Discharge Medication Route		

7.52 Visual Presentation Layer

Using CareWeb and the DEEDS dataset a more complete Emergency Department record such as the following can be generated. (Figure 16)

Visits

Site	Date		Observation	Value	Time	
South Shore	4/6/97	Pre Hospital	Mode of Transport	EMT	8:05am	
			EMS Unit	60		
			EMS Agency	Quincy Fire		
		Triage	Chief Complaint	Chest Pain	8:10am	
			Acuity	1		
		Physical Exam	ED Practioner	M. Jones MD	8:10am	
			Responsiveness	Alert		
			SBP/DBP (E/M/D)	140/80		
			Pulse	110		
			Resp Rate	20		
			Temp	99		
			HEENT	+JVD		
			Chest	Rales 1/3		
			ED Meds	ECASA	325mg PO	8:10am
				Heparin	5000u IV	8:10am
		Nitroglycerine		.4mg SL	8:10am	
		Diagnoses	Chest Pain	733.12	8:10am	
			Pulmonary Edema	727.12		
		Disposition	Admit	ICU	8:10am	
			Admitting Practioner	D. Roberts	8:10am	

Figure 16 - CareWeb Presentation Layer using DEEDS

8. Organizational and Institutional issues

During the development CareWeb™, we encountered many technical challenges which we solved by adhering to existing standards and maintaining an object oriented architecture. However, implementing a multi-institutional medical record using new technologies required an equal effort to address organizational and institution challenges.

The first challenge was to create consensus among all constituencies impacted by the deployment of a web-based medical record. Implementing CareWeb required collaboration with and approval from the Information Systems organization including the directors of Informatics and Advanced Technologies, Technology Management, Clinical Systems, Financial Systems, Strategic Projects, Community Practices, Information Resource Management and the Chief Information Officer. Deploying CareWeb required collaboration with and approval from the Security Officer and Corporate Communications. Finally, CareWeb required the support of the ultimate users of the system, the physician and nursing staff of the CareGroup Hospitals.

The second challenge was to mobilize the resources to transition CareWeb™ from proof of concept to a production system. Issues include administration of security hardware tokens, providing twenty-four hour per day, seven day per week support and providing dedicated hardware systems. We are currently allocating these tasks to existing information systems personnel.

The final challenge was to create an organizational security policy. Any technical implementation must be complemented by a strong organizational policy to sanction those who inappropriately access healthcare data. The Beth Israel Deaconess Medical Center has a long standing tradition of protecting patient confidentiality within its legacy systems and organizational policies are already in place. This greatly facilitated the implementation of the CareWeb security architecture and minimized the barriers to implementation which would be found at institutions without such existing policies.

9. CONCLUSION

CareWeb™ provides a rapidly deployable, low cost architecture which leverages the strengths of existing institutions merging to form an integrated health care delivery system.

Our experience with the CareWeb™ system has demonstrated the feasibility of using the web to allow access to longitudinal patient record data which is distributed across multiple sites, providers and institutions. We have demonstrated that a security architecture can be built around this system to provide a balance between ease of access to emergency health care data and protecting patient confidentiality. This security architecture builds on the work of others to create the first web-based implementation of the National Research Council's recommendations for present and future security practices.

We have learned many valuable lessons during the development of CareWeb™. First, the organizational barriers to deploying a secure web-based medical record can outweigh the technical challenges. Continuing reports of flaws in internet security give a public impression that the web is not a suitable environment for sensitive information and this creates difficulty in obtaining institutional support. Consensus for deploying such a system must include information systems personnel, hospital administrators, public relations and the clinicians themselves.

Secondly, the existing hospital infrastructure provides a strict limitation on the types of technology that may be deployed. Although the Beth Israel Deaconess Medical Center has a campus-wide high speed network infrastructure, many machines are incapable of running the current versions of web browsers, preventing the use of Java and browser-side scripting languages.

The CareWeb™ security architecture offers an early trial implementation of several potential strategies for delivering secure medical records via the Internet. This is especially relevant given that the Health Insurance and Portability and Accountability Act of 1996 (Kassebaum-Kennedy) requires that the Secretary

of Health and Human Services submit to Congress, by August of 1997, detailed recommendations on protecting the privacy of individually identifiable health information.

The CareWeb implementation project was a diverse and challenging experience. As indicated in this report, deploying a cross institutional medical record via the world wide web required :

- Legacy System interfacing techniques
- Implementation of numerous of new internet technologies
- Translation of provider requests and legacy responses into Health Level Seven
- Token-based security technologies
- Leveraging existing hardware and software infrastructure
- Evaluation of organizational special interests
- Study of cryptography technologies
- Local and multi-institutional auditing strategies
- Enterprise-wide collaboration
- Evaluation of health care provider needs and behaviors
- Public relations efforts to maximize patient education and achieve consent for use of CareWeb

Substantial progress has been made on projects derived from the CareWeb technology. It is our hope that these projects will to improve the functionality and utility of CareWeb and most importantly, will benefit patient care.

References

1. Kohane I, Greenspun P, Fackler J, Cimino C, Szolovits P, W3EMRS - Access to Multi-Institutional Electronic Medical Records with the World Wide Web, Spring Congress of the American Medical Informatics Association, Boston, MA 1995.
2. McDonald CJ, Overhage JM, Tierney WM et al, The RMRS: Physician use for input and output and web browser based computing, Proceedings 1996 Fall AMIA, 989.
3. Jagannathan V, Reddy YV, Srinivas K et al, An overview of the CERC ARTEMIS Project, Proceedings 1995 Fall AMIA, 12-16.
4. National Information Infrastructure Health Information Network, Essential Emergency Dataset, NIIHIN Consortium, Mequon, WI, 1995.
5. Pollock D, Data Elements for Emergency Department Systems Draft 3 Dataset, Center for Disease Control and Prevention, 1996.
6. Health Level Seven: An application protocol for electronic data exchange in healthcare environments, version 2.2, Chicago, Illinois Health Level Seven 1990.
7. Grossman JH, Barnett GO, Koepsell Td et al, An automated medical record system, JAMA 1973; 224:1616-21.
8. Chessare JB, Torok KE, Implementation of COSTAR in an academic group practice of general pediatrics. MD Comput 1992; 10:23-7.
9. Campbell JR, Civer N Seelig CB, Computerized medical records and clinic function. MD Comput 1992; 9:149-55.
10. McDonald CJ, Bleveins, Tierney WM, Martin DK, The Regenstrief Medical Records, MD Comput 1989; 5:34-47.
11. Stead WW, Hammond WE, Computerized medical records: A new resource for clinical decision making, J Med Syst 1983; 7:213-20.
12. Whitting O'Keefe EQ, Whitting A, Henke J, The STOR clinical information system, MD Comput 1988; 5:8-21.
13. Rind DM, Safran C. The development and evaluation of computer-generated alerts in an inpatient setting In Lun KC, Degoulet P et al MEDINFO '92, Proceedings of the 7th world conference on medical informatics, Amsterdam, Elsevier Science Publishers BV 1992 pp 249-54.
14. Pryor TA, Gardner RM, Clayton PD, Warner HR, The HELP System, J Med Sys 1983; 7:87-102.
15. Bolens M, Borst F, Scherrer JR, Organizing the clinical data in the medical record. MD Comput 1992; 9:149-55.
16. Baaker AR. The development of an integrated and cooperative hospital information system Med Inf 1984; 9:135-42
17. Kohane I, Greenspun P, Fackler J, Cimino C, Szolovits P, Building National Electronic Medical Record Systems via the World Wide Web, JAMIA 1996; 3(3) 191-207.

18. Kohane, I, van Wingerde FJ, Fackler JC, et. al, Sharing Medical Records Across Multiple Heterogeneous and Competing Institutions, Proceedings 1996 Fall AMIA 608-612.
19. McDonald CJ, Overhage JM, Tierney WM et al, The RMRS: Physician use for input and output and Web browser based computing, Proceedings 1996 Fall AMIA, 989.
20. Willard KE, Hallgren JH, Sielaff B, Connelly DP, The Deployment of a World Wide Web (W3) Based Medical Information System, Proceedings 1995 Fall AMIA 771- 775.
21. Klimczak JC, Witten DM et all, Providing Location Independent Access to Patient Clinical Narratives using Web Browsers and a Tiered Server Approach, Proceedings 1996 Fall AMIA 623-627.
22. Safran, C, Role of Computing in Patient Care in Two Hospitals, MD Computing 6:3 1989.
23. Duncan K, Patient Indexing Strategies, ComNets, February/March 1996.
24. Microsoft, Active Server Pages White Paper, Microsoft, 1996.
25. Clinton, William, State of the Union Address, February 4, 1997.
26. For the Record: Protecting Electronic Health Information, Computer Science and Telecommunications Board, Commission on Physical Sciences, Mathematics and Applications, National Research Council, National Academy Press, 1997.
27. Safran C, Rind D, et al, "Protection of Confidentiality in the Computer-based Patient Record", MD Computing Vol 12, No. 3, 1995.
28. Hickman KEB, Elgamal T, The Secure Sockets Layer Protocol 3.0, Internet Draft, Netscape Communications Corporation 1996
29. Schneier H , Applied Cryptography, John Wiley and Sons, 1996
30. Keen, PG, Business Multimedia Explained, Harvard Business School Press, Boston, Massachusetts (in press.)
31. Johnson SB, Friedman C, Cimino JJ, Hripcsak G, Clayton PD, Conceptual data model for a central patient database, Proceedings of the Fifteenth Annual Symposium on Computer Applications in Medical Care, 381-385.