WebCIS: Large Scale Deployment of a Web-based Clinical Information System

George Hripcsak, M.D., James J. Cimino, M.D., Soumitra Sengupta, Ph.D.

Department of Medical Informatics, Columbia University, New York

WebCIS is a Web-based clinical information system. It sits atop the existing Columbia University clinical information system architecture, which includes a clinical repository, the Medical Entities Dictionary, an HL7 interface engine, and an Arden Syntax based clinical event monitor. WebCIS security features include authentication with secure tokens, authorization maintained in an LDAP server, SSL encryption, permanent audit logs, and application time outs. WebCIS is currently used by 810 physicians at the Columbia-Presbyterian center of New York Presbyterian Healthcare to review and enter data into the electronic medical record. Current deployment challenges include maintaining adequate database performance despite complex queries, replacing large numbers of computers that cannot run modern Web browsers, and training users that have never logged onto the Web. Although the raised expectations and higher goals have increased deployment costs, the end result is a far more functional, far more available system.

Introduction

The emergence of the World Wide Web promises to make clinical information available to users wherever and whenever they need it. Informatics researchers and vendors have already begun to extend or replace their systems with Web-based user interfaces.[1-3] Although the Web has clearly sped application development, its effect on large-scale deployment still remains to be seen.

WebCIS is a Web-based clinical information system. It will replace a legacy system, known as "DHIS," at the Columbia-Presbyterian center of New York Presbyterian Healthcare, and it will be extended to other hospitals in the network. Deployment began in August 1998. As of June 1999, there were 810 physicians using WebCIS; the target is 4300 users by the end of the year. In this paper, we report on our experience building and deploying WebCIS in a large, complex environment.

Application

The main screen of WebCIS is shown in Figure 1. WebCIS is a sophisticated clinical repository browser that displays information from ancillary, registration, and ambulatory systems. Users retrieve information sorted by source department and then time; sorted purely by time; or aggregated in one of several views, including spreadsheets, cross patient summaries, and graphs. Users can define and request new spreadsheets over the electronic mail-based feedback function. Since its inception, several data entry functions have been added. Users may enter clinical notes that are signed electronically and archived permanently, they may create lists of patients, and they may enter work-list summaries for generating signout sheets (similar to Partners[4]).

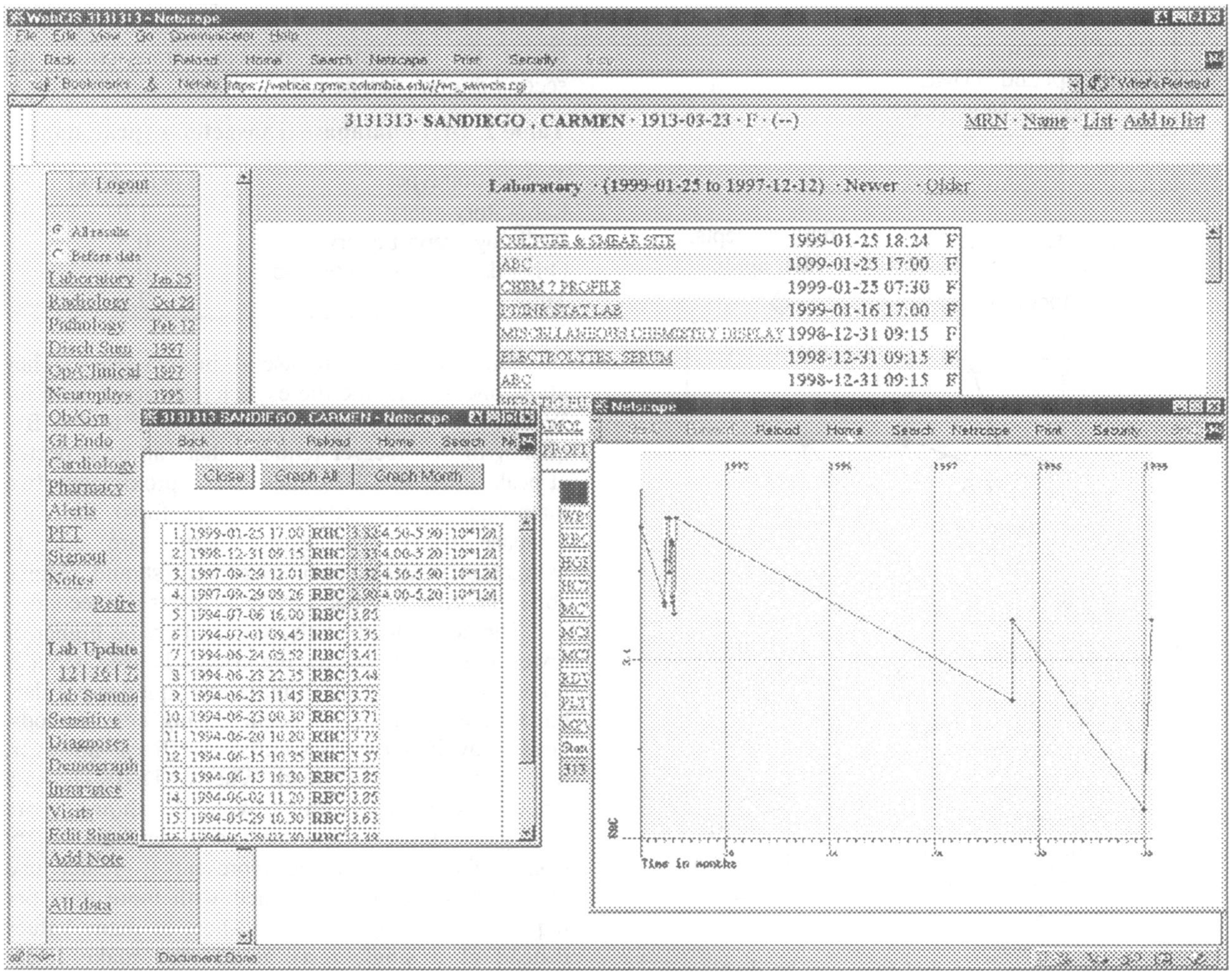
Architecture

A clinical information system architecture allows an institution to manage change.[5,6] The legacy clinical repository browser, DHIS, sits atop the Columbia clinical information architecture[7], shown in Figure 2. There is a centralized clinical repository[8] that collects data from all clinically relevant applications in the institution, including departmental systems and interactive clinical user interfaces. A Health Level Seven (HL7) interface engine mediates all data transfer. It uses information from the institutional vocabulary, known as the Medical Entities Dictionary (MED),[9] which defines all coded data stored in the database, translates between application coding systems, and provides a classification hierarchy and semantic relationships that simplify coding and vocabulary maintenance. As data are stored in the repository, messages are sent to an Arden Syntax-based event monitor[10] that provides automated decision support.

DHIS is a character-based system that runs on 3270 emulation software. It was built using an IBM mainframe tool set for health applications (PCS-ADS, IBM, Hawthorne, New York). This software, which was written in the 1980's, will not run on year 2000 compliant versions of the mainframe operating system and therefore must be replaced.

WebCIS is DHIS's replacement. It sits atop the same clinical information system architecture, which has required only minimal changes to accommodate the new repository browser. WebCIS's design, which is shown in Figure 2, is similar to that of two other Web-based applications, PolyMed[11] and PatCIS.[12] WebCIS is implemented as a set of Common

Figure 1. WebCIS main screen.



Gateway Interface (CGI) programs written in C and running on a UNIX Web server. The Web server communicates with the mainframe-based repository using the TCPIP socket protocol. The CGI programs generate HTML and JavaScript, which then execute on the client Web browser.

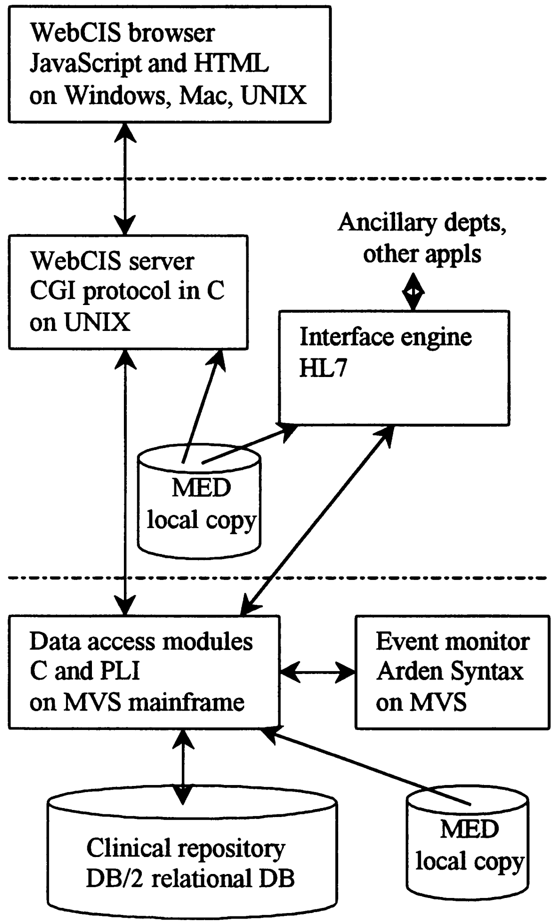
WebCIS exploits the existing architecture in several ways. Spreadsheets are not hard coded, but are defined in the MED. The class "clinical view" contains all spreadsheets, which are in turn composed of classes of results such as "intravascular sodium test." Several different applications share the same set of spreadsheets, improving consistency across the institution's suite of clinical applications and reducing maintenance. New user requests for spreadsheets are sent to a knowledge engineer, who can quickly incorporate the changes into the MED.

Multi-institution support

The Web promises to facilitate inter-institution communication.[1,2,13] The recent merger of Presbyterian Hospital, New York Hospital, and 15 other hospitals to form New York Presbyterian Healthcare has created the need to coordinate information for five million patients. There are many different clinical applications installed among the hospitals, and it is unlikely that a single suite of applications will suffice. Therefore, the goal is to merge clinical information from the institutions into a single repository and to use WebCIS as a common viewer for this information.

The Columbia clinical information architecture is facilitating this task. New data definitions are stored in the MED, data is uploaded via the HL7 interface engine into the repository, and WebCIS displays the information without requiring further coding. For example, new sodium tests are placed under the class

Figure 2. Architecture



"intravascular sodium" so that they can be displayed along with the sodiums from other institutions. (As of this writing, data from the New York-Cornell center has been stored on a pilot basis.)

Performance requirements

One of the primary challenges facing any clinical information system is performance. DHIS has sub-second response time, and users expect similar performance from its replacement. We therefore assessed the recent load on DHIS. There have been about 4300 distinct clinical users per month. The overall breakdown of their roles is shown in Table 1.

Table 1. DHIS distinct users per month.

User type	Number
Physicians	1500
Nurses	1000
Students	400
Administrative, other	1400
Total	4300

The performance of the clinical information system is bound by the database queries. Table 2 shows the number of database transactions per second during peak periods.

Table 2. DHIS database transactions (peak hours).

Transaction type	# per second
Laboratory query	1.0
Radiology report query	0.5
Other data query and upload	0.5
Total	2.0

These numbers are useful in two ways. They allow one to estimate the expected database load for WebCIS, and they provide an estimate to the frequency of Web server requests that can be expected at peak times. In DHIS, there are approximately two mainframe server requests (screen flips) for each database query; the extra queries are due to navigation screens. WebCIS servers should receive about the same number, or four requests per second.

The actual load on the system may be higher or lower, depending on how the system is used. Greater home usage may result in more total queries per day, but they are unlikely to occur during peak periods. Greater availability from outlying offices, however, may increase the peak load. Whereas DHIS shows a single result at a time, WebCIS shows more aggregate views. Therefore, each database query from WebCIS, on average, represents a greater load on the database server. Armed with more data per request (for example, a spreadsheet of many values instead of a single test), users may make fewer requests by spending more time on each aggregate view. Or users may move through the requests just as quickly, increasing the load on the database. So far, a significant change in load has not been detected.

Therefore, it appears that existing Web servers will easily meet the demands of WebCIS (even using the somewhat inefficient CGI protocol). Achieving sufficient throughput at the database may be a challenge, however.

Network performance is also an issue. Many users are accessing the system from homes or offices via older modems as slow as 14.4 kilobaud. Server requests and replies have therefore been kept as short as possible, and unnecessary images have been eliminated.

Security

The expansion of the World Wide Web has created enormous sensitivity over patient privacy and security for a number of reasons. The popularity of the Web and ease of access to many information resources has made the issue concrete to people who otherwise would not have thought about or understood the issue. More clinical applications are being made available over the Internet. Many more people, including potential intruders, are connected to the Internet today than a few years ago. Web search engines make it easier to find many Web-based clinical information systems, thus making them targets for attack. Although better software to protect security has appeared over time, more software to compromise security (for example, password cracking programs) are also available. The greater complexity of computer systems and the ease of placing information on the Web (even by mistake) have made institutions more vulnerable. For example, a university health system accidentally placed a 10 megabyte log of the activity of its patient scheduling system on the World Wide Web, exposing patient names, social security numbers, employment, diagnoses, and treatment records to the public[14]; it was discovered via the university's Web site search engine. As more people gain even legitimate access to electronic medical records, sensitivity to privacy is increasing. For example, at another institution, patient records and audit logs were made available electronically to patients who were employees of the institution in a pilot program. One employee, on discovering that a person had inappropriately accessed his record, sued the institution and settled out of court. The pilot was discontinued.

There has therefore been enormous sensitivity locally over patient privacy and security, and a great deal of effort has gone into security. WebCIS employs a number of security features.[3,12] Authentication is based on a password and a token. On entering the application from the Internet, a user is asked to enter his user ID, a secure password, and the number from a smart card (SecurID, Security Dynamics Technologies, Inc., Bedford, MA). Access from within the campus requires only the user ID and password. When users create passwords, the passwords are immediately run through password cracking programs to ensure that non-trivial passwords are used. User accounts are inactivated after six months of nonuse.

Authorization information is stored in a Lightweight Directory Access Protocol (LDAP) server. Authorization is mainly role-based with the ability to make individual exceptions. Access to individual types of clinical data can be set with a fine granularity. Encryption is accomplished with Netscape's implementation of the Secure Sockets Layer (SSL) protocol, and it is applied both over the Internet and within the campus. All access is audited in a permanent audit trail. Recorded information includes user, IP address, patient, data type, and time of access. Audit trails are monitored regularly for inappropriate access, and serve as documentation for patient complaints of privacy breaches. To reduce tailgating, in which a person uses someone else's account after he fails to log off, sessions time out after five minutes on all locations except bedside terminals. Bedside terminals, which are placed in secure areas inside of intensive care units, have unlimited timeouts but they require re-entering the ID and password every time a patient is changed. The hospital has hired a full time security officer to oversee all information system security operations.

Deployment and training

Use of the World Wide Web promises to reduce deployment and training costs in several ways. Home computers come preinstalled with Web browser software, and many potential users have gotten familiar with using Web-based applications and have already learned skills necessary to use the Web: a pointing device, links, pull down menus, etc. Web standards allow the same application to be viewed over a multitude of hardware platforms, software, and communications media.

In practice, deployment and training have become more difficult, however. This is due in part to the attempt to deploy applications much more broadly than ever before, reaching into homes and offices on an unprecedented scale. The WebCIS deployment involves computers and printers on nursing stations, clinics, administrative offices, ancillary department offices, physician's private offices, and homes. Replacement of outdated computers represents a large part of the deployment cost: the initial WebCIS roll out requires 300 new computers at nursing stations and clinics alone to replace 286-based PCs. The deployment staff is not directly responsible physician's private offices and homes, but successful deployment requires that these areas be functioning successfully. Installation of unrelated software (including games) by the computer owner often leads to unexpected problems.

Furthermore, Web standards are still not mature. Due to incompatibilities among Web browser JavaScript implementations, WebCIS requires Netscape version 4.04 or higher running on Windows 95, Windows 98, Windows NT, Macintosh, or UNIX, but not Windows 3.1. Attempts to use WebCIS with an unsupported browser results in a warning message, and then usually a series of JavaScript errors that may or may not stop WebCIS from running. Users who favor browsers such as Internet Explorer often refuse to switch to Netscape, even though they can install both browsers on their machine.

Many of the target users have simply not explored the Web yet. Due to limitations on the user's time and limitations on the budget, most of the training has been accomplished as demonstrations at departmental meetings and in small informal sessions. We have also minimized training by making the core of WebCIS look analogous to DHIS—a series of departmental data sources that one can inspect to view results and documents one at a time. Parallel to this are a series of more sophisticated clinical information views that are supported only in WebCIS: spreadsheets, graphs, temporal filters, and cross-patient summaries. Naïve users start with basic features and move to advanced features as they become more comfortable with the Web paradigm and alternate views of information.

Discussion

The World Wide Web has certainly made clinical information more available to more people. Even for the initial 810 active users, there is greater interest in using WebCIS from home and remote offices. Compared to the effort to deploy an X-Windows based application, WebCIS has been far easier.

Nevertheless, use of the Web has increased expectations, goals, and deployment costs. Although Web-based application development is very fast, other aspects of the application life cycle have not been shortened: developing a business plan, getting buy-in, raising capital, buying and installing hardware, training users, etc. An application that takes weeks to develop can take years to deploy in a large, complex, multi-institution environment. By the time an application is fully deployed, its technology may be obsolete. For example, WebCIS is based on CGI programs and JavaScript; if the design were redone today it would look very different. If newer Web browsers do not continue to support the previous generation technology well—for example, if JavaScript support becomes unreliable—then the application will have to be rebuilt before it is fully deployed.

The best one can do is stick to a good clinical application architecture, with well-isolated modules, well-defined layers, and industry standard protocols. As breakthroughs occur at each level or within each module, they can be incorporated without replacing the whole system.

References

1. McDonald CJ, Overhage JM, Dexter PR, et. al. Canopy computing: using the Web in clinical practice. JAMA 1998;280:1325-9
2. Kohane IS, Greenspun P, Fackler J, Cimino C, Szolovits P. Building national electronic medical record systems via the World Wide Web. J Am Med Inform Assoc 1996;3:191-207.
3. Cimino JJ, Socratous SA, Clayton PD. Internet as clinical information system: application development using the World Wide Web. J Am Med Inform Assoc 1995;2:273-84.
4. Hiltz FL, Teich JM. Coverage List: a provider-patient database supporting advanced hospital information services. Proc Annu Symp Comput Appl Med Care 1994;:809-13.
5. Stead WW, Borden R, McNulty P, Sittig DF. Building an information management infrastructure in the 90s: the Vanderbilt experiment. Proc Annu Symp Comput Appl Med Care 1993;534-8.
6. Hripcsak G. IAIMS Architecture. Journal of the American Medical Informatics Association 1997;4:S20–30.
7. Johnson SB, Forman B, Cimino JJ, et. al.. A Technological Perspective on the Computer-Based Patient Record. In: Steen EB (ed). Proceedings of the First Annual Nicolas E. Davies CPR Recognition Symposium, 1995; 35–51.
8. Johnson SB, Hripcsak G, Chen J, Clayton P. Accessing the Columbia Clinical Repository. In: Ozbolt JG, editor. Proceedings of the Eighteenth Annual Symposium on Computer Applications in Medical Care, 1994; 281–5.
9. Cimino JJ, Clayton PD, Hripcsak G, Johnson SB. Knowledge-based approaches to the maintenance of a large controlled medical terminology. J Am Med Inform Assoc 1994;1:35–50.
10. Hripcsak G, Clayton PD, Jenders RA, Cimino JJ, Johnson SB. Design of a clinical event monitor. Comput Biomed Res 1996;29:194–221.
11. Cimino JJ, Socratous, SA. Just tell me what you want!: the promise and perils of rapid prototyping on the World Wide Web. Proc AMIA Annu Fall Symp 1996;719-23.
12. Cimino JJ, Sengupta S, Clayton PD, Patel VL, Kushniruk A, Huang X. Architecture for a Web-based clinical information system that keeps design open and the access closed. Proc AMIA Symp 1998;121-5.
13. Teich JM. Clinical information systems for integrated healthcare networks. Proc Amia Symp 1998;19-28.
14. Glave J. Medical records exposed. In: Wired News, 1999 Feb 8 2:20 p.m. PST, http://www.wired.com/news/news/politics/story/17799.html.