Web Client and ODBC Access to Legacy Database Information: A Low Cost Approach

Neal W. Sanders[a,b], N. Horace Mann III[a], Dan M. Spengler[b]
Departments of [a]Biomedical Engineering and [b]Orthopaedics and Rehabilitation
Vanderbilt University
Nashville, TN

A new method has been developed for the Department of Orthopaedics at Vanderbilt University Medical Center to access departmental clinical data. Previously this data was stored only in the medical center's mainframe DB2 database, it is now additionally stored in a departmental SQL database. Access to this data is available via any ODBC compliant front-end or a web client. With a small budget and no full time staff, we were able to give our department on-line access to many years worth of patient data that was previously inaccessible.

INTRODUCTION

The Department of Orthopaedics at Vanderbilt University Medical Center required access to their clinical data for purposes of research. Unfortunately, this data was stored in a legacy database that made retrieval very difficult. This problem is not uncommon[1,2]. Although there have been many different solutions to this problem[3,4,5], the approach that best fit our situation was a SQL database accessed by a combination of web server and ODBC link.

Our implementation is noteworthy for several reasons. First, it was a departmental endeavor that required very little effort by the hospital's information management department. Second, the cost of implementation was very low (approximately $1,900). Third, it provided easy access to important data that would otherwise have been unobtainable.

In addition to satisfying the initial objectives of research, this database design has proven flexible enough that it is being utilized for several other non-research purposes. For example, it is used to aid in the generation of resident surgeon educational review reports, to provide patient geographic information (for marketing purposes), and as a source of demographic information for sub-specialty specific data needs.

This paper focuses on how we developed the system, from the database to the user interfaces, and the capabilities it provides.

BACKGROUND

Legacy

In the fall of '94 we were asked to evaluate a legacy database that contained information believed pertinent to orthopaedic outcome studies. This database was stored in a DB2 database residing on the medical center's IBM mainframe. Due to personnel changes in the department, we had no one who understood how to access this database. Eventually we found one person who believed they knew how to access it; however, upon demonstration they were unable to actually access any data (it was complicated to the extent that it was unusable). Simply training another person to access this legacy system was not a viable solution due to the fact that we required much simpler access and better report generation than was available with the legacy mainframe system.

Initial Effort

In an effort to provided improved access and usability the DB2 tables and data were ported to an SQL database residing on the department's Novell server. Although this was a great improvement over the DB2 database and the data could now be queried by an ODBC link from Microsoft Access, it was not simple enough for easy use. It did, however, provide us with several important lessons: 1) A cheap database ($140) was not good enough. 2) Our data needs were expanding beyond the initial purposes of the DB2 tables. 3) The existing DB2 table structure

1091—8280/97/$5.00

was poor and the data content was insufficient. 4) We required a much simpler method of access.

DESIGN CONSIDERATIONS

Based on the lessons learned from our initial effort, we had better defined goals for our second attempt. First, and most important, the data must be accurate and access must be reliable. Second, we needed a way for both ad-hoc queries and very simple "anyone can do it" access to common queries and reports. Third, we needed a more complete set of data so that we could more easily adapt to additional requirements.

Fortunately, at this time (early '96) there were great improvements being made in relevant technologies. Better SQL databases were available at reasonable costs. The web servers were improving and their ability to access databases was being enhanced almost monthly. In addition, it was at this time that decisions were being made to provide a web client for VUMC's MARS system[6].

If we provided a web client interface to the legacy data, we could then add the ability to hyper-link to the patient's MARS record (lab reports, discharge summaries, etc.) and thus have a much richer and potentially more useful tool. Although we already could easily justify having a web based front-end to our database, this potential capability was enough to end any consideration of other methods.

SYSTEM ARCHITECTURE

We tested two SQL databases: Watcom SQL and Microsoft SQL, both running under Microsoft Windows NT Server. Microsoft SQL was chosen because of its features and reputation. Both Novell web server and Microsoft's Internet Information Server (IIS) were tested. We also tested WebDBC (StormCloud Corp.) as a way to connect the web server to the SQL database. Microsoft's IIS was chosen since it had database connection features similar to WebDBC's and because it would easily integrate with our existing operating system and database choices.

The server architecture is shown in Figure 1. It shows NT server (version 4.0) hosting both the web server (IIS 3.0) and the SQL server (version 6.5). Open Database Connectivity (ODBC) is the communication method between the servers. Although both of these are currently running on a single machine, the architecture does not require this and it may be scaled to two or more machines.

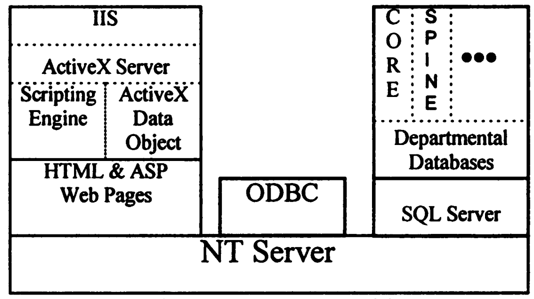


Figure 1: Server Architecture

Next we analyzed our data needs and redesigned the table structure. The primary goal of this process was to get the data we needed without introducing problems due to incompatibilities with the legacy design. We worked with information management to achieve this. To the extent it was practical, we recreated the table structure from DB2's original tables (not the ones designed specifically for orthopaedics mentioned in the background section.) This resulted in a database (referred to as the "Core" database) with twelve tables – six of which are "support" tables.

Figure 2 shows the process of importing the desired data from the legacy system. A program was written by information management personnel to collect any updates or modifications to data related to orthopaedics. On a weekly basis, they send text files containing that data to our FTP server (same machine as shown in figure 1). We wrote custom software to parse this data and properly update our database.

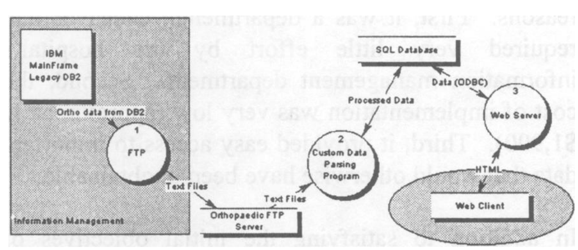


Figure 2: Data Flow and Processes

We knew a priori that the greatest challenge to storing the data would be determining which records were updates and which ones were new. The DB2 database was not designed to provide this information. With the help of information management the problem was overcome using a set of rules, an extra field, and intelligence in the custom parsing program.

The custom parsing program had to be able to properly distinguish a new record from an update, and if it was an update, how to update the record. It had to be robust enough to run unattended as a scheduled service under NT. This required the ability to log error messages and records that could not be parsed. This program was written in C++.

The first step in loading the data into the tables was to load the historical data ('86 to present). At each step in the initial loading process, there was an intensive effort to detect any errors that could be causing bad data – we found several. We also went to great pains to understand each of the errors. Most of the problems were related to "rules" in the legacy system that in fact were not enforced. The second step in the data loading process is performing weekly updates from DB2.

It could be argued that an ODBC (or similar) link directly to the DB2 tables would have been a more satisfactory solution than the FTP process. However, this would have required both a policy change and expenditures by information management. Therefore, we found an acceptable way to achieve our goals with minimal effort and no expense required of information management.

IMPLEMENTATION

With the database and web server in place and stable, we were ready to allow user access to the system. Due to the complexity of data coding and the fact that most physicians do not want to perform their own ad-hoc queries, we have trained an existing employee to do them. Most of the ad-hoc queries are done using Microsoft Access with tables ODBC linked to the SQL server.

Web Client

The more exciting part of the implementation is the web-based interface. It is with this interface that any authorized person can access patient information in a very simple manner.

Starting from an introductory page, the user has several choices, one of which is to go to the Core database. On this page is a menu containing query, report and utility choices (see Figure 3). The focus of the following explanation of capabilities will be on the query menu.

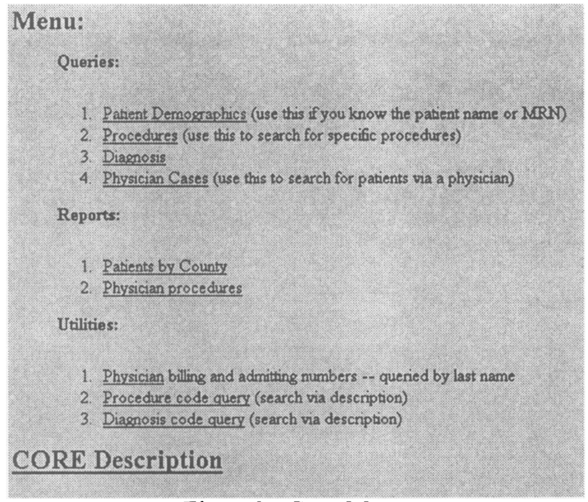


Figure3: Core Menu

All four of the query options will lead to the same information; the selection depends on what is known up front and if the focus is on a patient, physician, or the type of care.

If a procedure query is selected, the user is presented with a screen (see Figure 4) allowing them to select a physician, procedure code (ICD-9 or CPT), date range and from what source the search is started. The choice of "Medical Records" or "Physician Billing" is important since they are coded differently.

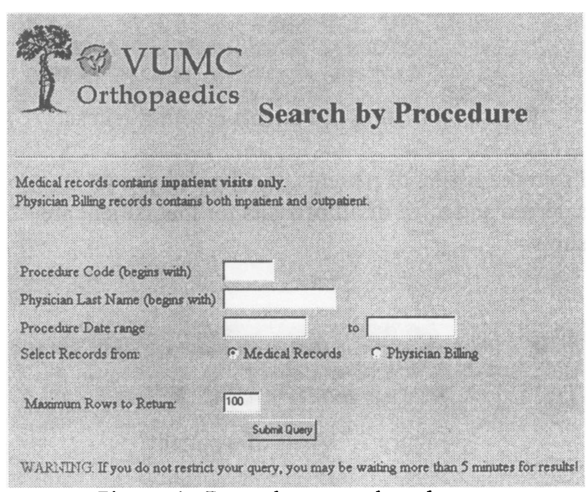


Figure 4: Procedure search web page

Figures 5 through 8 demonstrate a typical query scenario for a fictitious patient whose last name is "Brooks" and first name starts with "G."

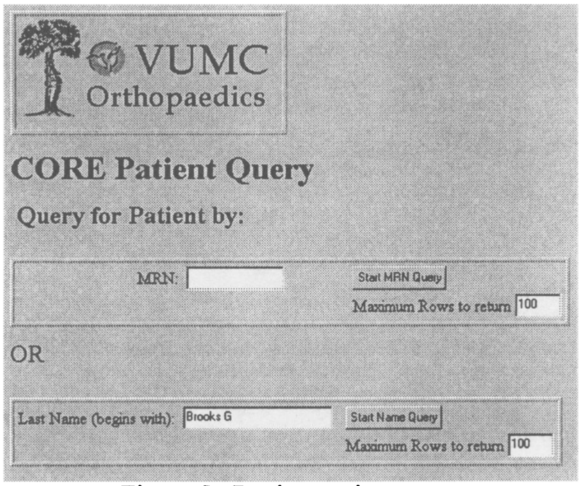
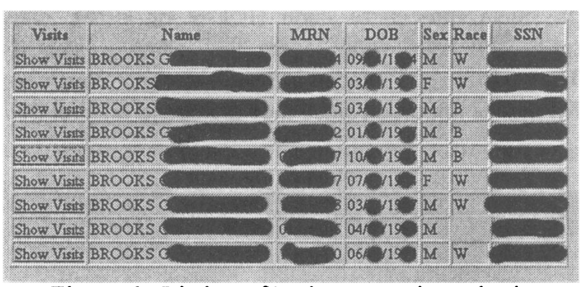


Figure 5: Begin a patient query

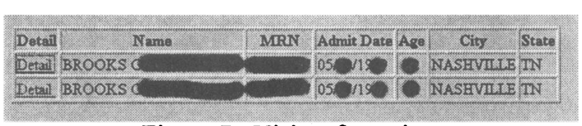
This query request starts an active server page (ASP) process on the web server. The ASP converts the request into SQL and utilizes ActiveX and ODBC technology to query the database and return the results as an HTML page as shown in figure 6.



Visits	Name	MRN	DOB	Sex	Race	SSN
Show Visits	BROOKS G[redacted]	[redacted]4	09[redacted]/1[redacted]	M	W	[redacted]
Show Visits	BROOKS[redacted]	[redacted]5	03[redacted]/19[redacted]	F	W	[redacted]
Show Visits	BROOKS[redacted]	[redacted]5	03[redacted]/19[redacted]	M	B	[redacted]
Show Visits	BROOKS G[redacted]	[redacted]2	01[redacted]/19[redacted]	M	B	[redacted]
Show Visits	BROOKS [redacted]	[redacted]7	10[redacted]/19[redacted]	M	B	[redacted]
Show Visits	BROOKS [redacted]	[redacted]7	07[redacted]/19[redacted]	F	W	[redacted]
Show Visits	BROOKS G[redacted]	[redacted]8	03[redacted]/19[redacted]	M	W	[redacted]
Show Visits	BROOKS [redacted]	[redacted]5	04[redacted]/19[redacted]	M		[redacted]
Show Visits	BROOKS G[redacted]	[redacted]0	06[redacted]/19[redacted]	M	W	[redacted]

Figure 6: Listing of Patients meeting criteria

From the listing of patients, the desired patient is selected and a list of all of visits for that patient are shown.

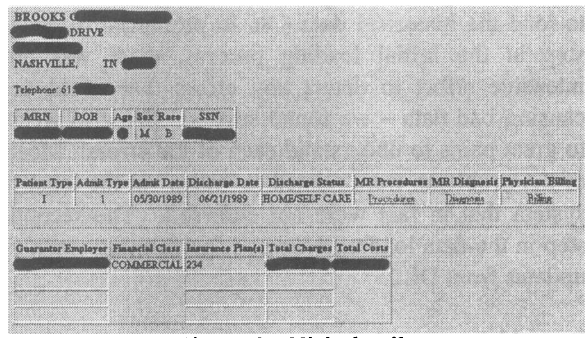


Detail	Name	MRN	Admit Date	Age	City	State
Detail	BROOKS G[redacted]	[redacted]	05[redacted]/19[redacted]	[redacted]	NASHVILLE	TN
Detail	BROOKS G[redacted]	[redacted]	05[redacted]/19[redacted]	[redacted]	NASHVILLE	TN

Figure 7: Visits of a patient

From the list of patient visits, the user selects the visit of interest by clicking on the "details" link, and is shown what happened to the patient on that visit (see figure 8). The details include the patient's complete demographics as well as the type of visit, date(s) of care, discharge status, financial information, and links to details of procedures and diagnosis via either medical records coding or billing coding. In addition, there are fields for total cost and charge. These, in turn, are linked to a page showing the breakdown of the costs and charges.

This page and the previous one is the most likely place where a link to the MARS system will be included.



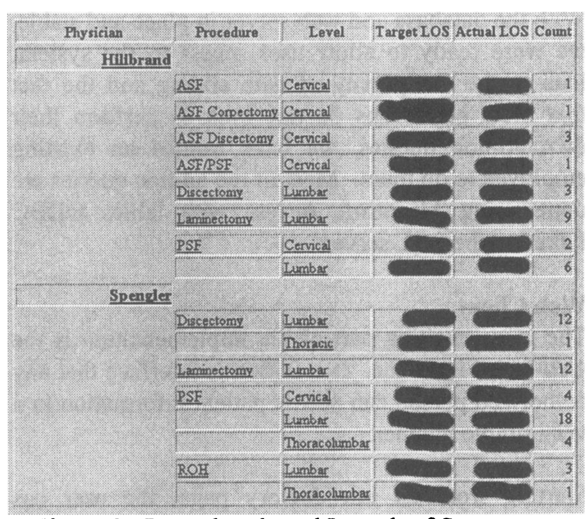
MRN	DOB	Age	Sex	Race	SSN
[redacted]	[redacted]	[redacted]	M	B	[redacted]

Patient Type	Admit Type	Admit Date	Discharge Date	Discharge Status	MR Procedures	MR Diagnosis	Physician Billing
I	1	05/30/1989	06/21/1989	HOME/SELF CARE	Procedures	Diagnosis	Billing

Guarantor Employer	Financial Class	Insurance Plan(s)	Total Charges	Total Costs
[redacted]	COMMERCIAL	234	[redacted]	[redacted]

Figure 8: Visit details

In addition to the basic types of queries and reports shown above, this data has other uses. The spine group within orthopaedics has used this data and added to it creating a procedure-based length of stay report (see figure 9). Because it is linked to the full patient information, it is easy to "drill down" into the data to see what caused the numbers to be what they are.



Physician	Procedure	Level	Target LOS	Actual LOS	Count
Hillbrand					
	ASF	Cervical	[redacted]	[redacted]	1
	ASF Corpectomy	Cervical	[redacted]	[redacted]	1
	ASF Discectomy	Cervical	[redacted]	[redacted]	3
	ASF/PSF	Cervical	[redacted]	[redacted]	1
	Discectomy	Lumbar	[redacted]	[redacted]	3
	Laminectomy	Lumbar	[redacted]	[redacted]	9
	PSF	Cervical	[redacted]	[redacted]	2
		Lumbar	[redacted]	[redacted]	6
Spengler					
	Discectomy	Lumbar	[redacted]	[redacted]	12
		Thoracic	[redacted]	[redacted]	1
	Laminectomy	Lumbar	[redacted]	[redacted]	12
	PSF	Cervical	[redacted]	[redacted]	4
		Lumbar	[redacted]	[redacted]	18
		Thoracolumbar	[redacted]	[redacted]	4
	ROH	Lumbar	[redacted]	[redacted]	3
		Thoracolumbar	[redacted]	[redacted]	1

Figure 9: Procedure based Length of Stay report

CURRENT STATUS

In the fall of '96 the database began operating continuously and reliably. As of March '97, the database has 1.5 million records on over 250,000 patient visits with approximately 25,000 records being added per month.

Ad-hoc query capability has been available since mid fall '96. Typically, there are several query requests each week. Although the systems availability has not been widely advertised, we receive an increasing number of requests from a growing number of physicians.

To date (March '97) the web interface has been used by 8-10 people and all have been excited that it is simple to use and amazed that we are finally able to see information that has been hidden for so long. There are several planed features that have not been fully implemented and we are still experimenting with what works best before inviting everyone to use the system.

FUTURE WORK

We recently changed our web pages to Active Server Pages utilizing ActiveX technology. With this change our web "browser" can now be used as a web "client." We can easily integrate advanced features into our application that should allow user input of information. Currently under development is a program to allow patient information to be entered via the web client. Although others have done this, the limitations in web technology have prevented program development of the quality of traditional client/server programs[5], but with ActiveX it should now be possible.

The web pages are constantly changing as new features are added and soon we hope to make one of those features a link to the MARS system.

CONCLUSION

With a small budget and no full time staff, we were able to give our department access to many years worth of patient data that resided on a legacy system.

Attaching tables to Microsoft Access for ad-hoc queries has worked very well. Its report capabilities meet our needs.

For simplicity, the web client can not be beat. Although it has it limitations, those limits are shrinking almost daily. We will expand our use of this access method.

Acknowledgments

We would like to thank Johniene Doran in the department of information management for her help in providing the data from DB2.

References

1. Prather JC, Loback DF, Hales JW, Hage ML, Fehrs SJ, Hammond WE. Converting a Legacy System Database into Relational Format to Enhance Query Efficiency. *Proc 19th Annu Symp Comp Appl Med Care.* 1995:372-6.

2. Willard KE, Sielaff BH, Connelly DP. Integrating Legacy Laboratory Information Systems into a Client-Server World: the University of Minnesota Clinical Workstation Project. *Methods of Information in Medicine.* 1995:289-96.

3. Flanagan JR, Chun J, Wagner JR. Evolution of a Legacy System to a Web Patient Record Server: Leveraging Investment While Opening the System.

4. 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(5):273-84.

5. Sittig DF, Kuperman GJ, Teich JM. WWW-based Interfaces to Clinical Information Systems: The State of the Art. *Proc 1996 AMIA Annu Fall Symp.* 1996:694:8

6. Giuse DA, Mickish A. Increasing the Availability of the Computerized Patient Record. *Proc 1996 AMIA Annu Fall Symp.* 1996:633:7.