SCAMC Student Paper Competition Finalist ■

Client-server, Distributed Database Strategies in a Health-care Record System for a Homeless Population

HENRY C. CHUEH, MD, G. OCTO BARNETT, MD

Abstract **Objective:** To design and develop a computer-based health-care record system to address the needs of the patients and providers of a homeless population.

Design: A computer-based health-care record system being developed for Boston's Healthcare for the Homeless Program (BHCHP) uses client-server technology and distributed database strategies to provide a common medical record for this transient population. The differing information requirements of physicians, nurses, and social workers are specifically addressed in the graphic application interface to facilitate an integrated approach to health care. This computer-based record system is designed for remote and portable use to integrate smoothly into the daily practice of providers of care to the homeless. The system uses remote networking technology and regular phone lines to support multiple concurrent users at remote sites of care.

Results: A stand-alone, pilot system is in operation at the BHCHP medical respite unit. Information on 129 patient encounters from 37 unique sites has been entered. A full client-server system has been designed. Benchmarks show that while the relative performance of a communication link based upon a phone line is 0.07 to 0.15 that of a local area network, optimization permits adequate response.

Conclusion: Medical records access in a transient population poses special problems. Use of client-server and distributed database strategies can provide a technical foundation that provides a secure, reliable, and accessible computer-based medical record in this environment.

■ **J Am Med Informatics Assoc.** 1994;1:186–198.

A computer-based health-care record system is being developed for Boston's Healthcare for the Homeless Program (BHCHP) to help providers manage the diverse health-care needs of Boston's homeless population. Recent studies of the homeless population have emphasized the complexity of health-care issues in this group.[1] Their use of the health-care system is sporadic, and access to health services does not occur through traditional channels. Patients are often seen by different BHCHP providers at multiple sites. Effective treatment of this transient population requires better access to medical records than the current paper-based records provide. The growing number and diversity of the homeless population also creates a challenge for providers managing these patients. In 1992 alone, BHCHP providers delivered health care to over 5,000 homeless persons in more than 37,000 encounters. Approximately half of these encounters occurred at the medical respite unit, a 50-bed subacute, inpatient facility for patients who might otherwise require hospital admission.

This project addresses the health-care information needs of these providers and patients with a computer-based health-care record using client-server, distributed database strategies. Low-cost regular telephone lines are used for remote access to a central database. A major focus of the project is the attention to a flexible application interface that is patient-oriented in terms of both accessibility and design. This paper describes the architecture and application interface of the system. Experience with a stand-alone

Affiliation of the authors: Harvard Medical School and Massachusetts General Hospital, Boston, MA.

Correspondence and reprints: Henry C. Chueh, MD, Laboratory of Computer Science, Massachusetts General Hospital, 50 Staniford Street, 5th Floor, Boston, MA 02114.

Received for publication: 11/4/93; accepted for publication: 11/15/93.

Table 1 ■

Percentages of 129 Homeless Patients Having Different Kinds of Identifying Information Recorded in the Respite Unit Database

Feature	%
First and last names	100
Sex	100
Physical attribute (e.g., eye color) (at least one)	98
Date of birth	85
Race	59
Social security number	50
Alias/pseudonym	1

pilot implementation and the performance characteristics of elements of the client-server architecture are reported.

Background

Patterns of Care in the Homeless Population

Homeless patients are often seen by different BHCHP providers at multiple sites in the city. These sites are diverse and include homeless shelters, clinics at major hospitals, soup kitchens, motels, the BHCHP medical respite unit, and a mobile van. While acute hospitalizations for these patients may be similar to usual patterns of care, the outpatient care patterns of the homeless are quite different. Most importantly, patients are seen frequently on an ad hoc basis for symptomatic complaints without regard to a primary site or provider of care.

The multiplicity of time, place, and person in these patients' *routine* health-care delivery makes traditional practice-based paper records all but unusable. Evidence that this can be a problem comes from a survey done by BHCHP last year. In the survey, 16 patients were randomly selected from the BHCHP respite unit, and their access to acute care facilities was reviewed. In a period of 18 months, these 16 patients accounted for 80 separate acute care admissions at 27 different facilities. While some patients do have a primary care provider or primary site of care, most of the population remains disconnected from traditional health-care access.[2] Follow-up care is also difficult to arrange, and will often occur as an impromptu visit with a different provider at another site. Although most non-homeless ambulatory care patients also have multiple care providers, a primary care provider usually coordinates these referrals. Finally, the disposition of homeless patients is particularly complex, since a physician might not be willing to discharge a homeless patient to the streets or a shelter although the patient might be well enough to be discharged to home were that an option.

Demographics

Initially, one major concern was that patient identification would be difficult in this population, since the use of pseudonyms or nicknames was thought to be common. Table 1 shows the percentages of different types of identifying information that were found for 129 patients from the database. The use of first and last names, sex, date of birth, and additionally one or more physical characteristics appears to be sufficient, and the use of pseudonyms does not seem to be a major issue. The most common medical problems and treatments of these patients are listed in Table 2.

Healthcare Management Needs

The patient population at BHCHP has unique needs for medical-record information because of the prevalence of specific disease entities in the homeless population coupled with the frequent lack of follow-up care.[2,3] These problems include such diagnoses as *substance abuse* and *trauma*. Other problems, such as AIDS and tuberculosis, can pose an increased health hazard to the general population if such problems become endemic in the homeless population. Because of the current difficulty in obtaining medical-record information about patient history, test results, chest x-rays, and treatments, managing these types of problems in this high-risk population is difficult.[4]

Some argue that the management of these patients' social welfare, rather than medical problems, is often the critical issue in the success of their overall health care.[5] Coping with these social issues requires a coordinated effort of care between medical and social

Table 2 ■

The Ten Most Common Problems and Medications of 129 Homeless Patients*

Problems	Medications
Alcohol abuse	Antibiotics
Fractures	Narcotics
HIV infection	NSAIDs
Psychiatric illness	H2-blockers
Substance abuse	Benzodiazepines
Skin infections	Dilantin
Hypertension	Albuterol inhalers
Diabetes	Anti-hypertensives
Seizure disorder	Dietary supplements
Asthma	Insulin

*Extracted from the respite-unit system database.

providers.[1] Integrated efforts can aid these patients in obtaining temporary shelter, insurance, transportation passes, low-cost housing, social security, and other benefits available to them through state-sponsored programs. A computer-based health-care system for this population must integrate these critical tasks and their associated information demands.

Design and Methods

The design and development of a computer-based record for this population has been approached in two ways. First, a pilot project had a goal of providing a basic computer-based record for BHCHP's medical respite unit. This project gave the authors an early opportunity to work with the BHCHP providers to understand the nature of the work they do with the homeless. The computer-based record we designed collects and maintains demographic, referral, admission, and discharge information, a problem and medication list, a list of patient appointments/events, and a patient census. The system was implemented in Microsoft FoxPro* as a stand-alone database system. While FoxPro does not support a full ANSI-standard structured query language (SQL) for client-server development, it does provide partial SQL support. Client-server techniques were used when possible in anticipation of the development of a more comprehensive system. The user interface elements were prototyped in the respite-unit system. Discussion about vocabulary for problems was initiated with the BHCHP staff during the design of the respite-unit system.

Second, a full client-server implementation was started based upon experience with the respite-unit system. Some of the problems associated with the non–client-server nature of the FoxPro-based system convinced us to use a full client-server architecture. On the other hand, the rapidity of local vocabulary matching and browsing in FoxPro influenced the use of distributed databases for rapid response. Entering information into the respite-unit system by using controlled vocabularies was a major obstacle, and less or slower functionality in this area could not be tolerated. The design and development of the full client-server solution is described in detail below.

Database Strategy

Central database server. Our solution for access to medical records from multiple sites by multiple providers is based on a client-server architecture. A transaction-based, relational database server maintains all patient medical record information. In addition to being a central repository for patient data, the server handles security issues by supporting techniques that grant access to the data, such as the use of provider password tables and database table access. The database server handles only data retrieval and update transactions, and does not participate in the applications interface in any way. We are using ORACLE7† database-server software running as a Novell NetWare Loadable Module (NLM)‡ on a 80486-based PC network server. This combination of software and hardware supports the design of a medical-record schema that includes a combination of both specific data tables and tables to handle generic, typed data.[6–8] The use of a PC-based CASE tool, ERwin/SQL,§ has been helpful for managing the schema.

Local database. A distributed database strategy is used to enhance performance of the application interface. Local databases reside on the client computers, which are fixed workstations at designated sites or notebook computers for portable use. The local database is also implemented using relational database software. We use WATCOM SQL¶ software as the local database engine. These local relational tables provide several capabilities: first, they support temporary but disk-based storage for records retrieved from the central server; second, they allow the large controlled vocabularies (used to enhance direct data entry by providers) to be stored locally and also provide storage for additional resources such as United States Pharmacopeia (USP) drug-information text; third, they provide storage for data entered by providers that have not yet been transmitted to the central server. Using the WATCOM database engine, a large diagnosis/problem vocabulary consisting of over 5,000 indexed terms can be used to validate entries into a problem list in one or two seconds.

Vocabularies and resources could be located on the central server and still provide the same functionality. However, our initial benchmark tests showed that queries against the server can be significantly slower than local database queries if the user enters a term that is not an exact match and an interactive display of possible matches is required. This delay

*FoxPro is a registered trademark of Microsoft Corporation.

†ORACLE and ORACLE7 are registered trademarks of Oracle Corporation.

‡Novell, NetWare Loadable Module, and NLM are registered trademarks of Novell COrporation.

§ERwin/SQL is a registered trademark of Logic Works.

¶WATCOM SQL is a registered trademark of WATCOM International Corporation.

occurs because of the long list of possible matches retrieved over the low-bandwidth telephone link in our system architecture. Low-bandwidth refers to the limited data-transfer rate of the communications link: 14,400 bits per second (bps) for the modem link (up to 57,000 bps with maximum data compression), compared with 8–10 million bps over a typical Ethernet network. This performance problem is attributable completely to the phone link and is not a problem intrinsic to the client-server model. Investigation by us and others using server-based vocabularies on a local area network achieved performance similar to local database performance.[9] Using local vocabularies provides rapid response even when queries result in a large number of possible matches. This in turn allows a provider to browse the list to find the best match, rather than forcing the provider to reformulate a more narrow query because too many terms matched (to be retrieved in a reasonable amount of time). An example query that returns many possible matches is "acute UTI," which returns over 100 entries from the vocabulary (see Figure 3).

Connectivity

Hardware. The client-server architecture is ideal for minimizing network traffic if remote access to a central server is a necessity. Connecting the client computer with the server is a central issue in these situations. Given the high cost and inaccessibility of dedicated wide-area networks and the infancy of wireless communications, we use regular phone lines with high-speed modems to connect remote sites to the server. Both modems and regular phone lines are relatively low-cost, tested technologies. Because of the low bandwidth of this type of connection, the transfer of data must be minimized to achieve respectable performance.

Multi-user, concurrent access to the database is handled by the network operating system. The relational database software runs on a dedicated database-server computer running the network operating system. This scheme allows any node on the network to be a client. Remote users are attached to the network over regular telephone lines by using remote networking technology from Shiva Corporation. Their NetModem and LanRover** products provide the hardware and software to negotiate the translation between network protocols and modem protocols.

Software. Another way we are achieving reasonable real-time performance is by using SQL to request information from the central server. SQL is a query language based on relational calculus, which is used by many relational databases as a standard query language.[10] The main advantage of SQL is the ability to query a database without procedural code. Only an SQL text string is sent to the database server, and the database server software decides the optimal way to satisfy the query. In the specific case of our low-bandwidth client-server link, using SQL limits data traffic over the link to SQL statements sent to the server and query-result data sets retrieved in return. Using transaction-based servers can, however, place increased requirements on the client software, especially when data-browsing facilities need to be implemented.[11]

Database Security

Physical access. Secured access to the central database from a remote site will be achieved through a validated callback mechanism. When a remote user dials into the system, the user will provide a user name, personal password, and site password, and then disconnect. The user will be validated from private database tables; if authorized, the central server will perform a callback to the corresponding phone number to establish a connection. The remote networking technology supports this callback mechanism directly. This connection method helps to protect the system against use from unauthorized phone numbers, which can be a major problem for dial-up systems. Maintaining a single central repository for all patient-based data minimizes physical data-security issues. Electronic data transmission is always a source for possible security problems; however, the use of compression in high-speed modem protocols makes the decoding of these signals extremely time-dependent and difficult to interpret even if intercepted.

Record retrieval and integrity. Once a secure connection is established with the central database server, a provider identifies a patient record to retrieve. Retrieval of a single record is achieved by identifying information such as name, age, sex, and birthdate. For this population, we provide additional identification such as pseudonyms and identifying physical characteristics such as eye color. Access to a patient record involves the retrieval of the most recent information entered in such places as the current problem list and medications. A record is available for retrieval by multiple providers at any given time. Concurrency and locking issues are problematic when multiple providers attempt to *update* existing data in the same patient record simultaneously. Most of the data entered by providers are treated as new. Consequently, they are *added* historically to the database

**NetModem and LanRover are registered trademarks of Shiva Corporation.

through SQL INSERT commands, and do not suffer from the concurrent record locking problems of SQL UPDATEs. The coordination of simultaneous database access is discussed in detail below. Since the ORACLE7 database supports transactions, data such as an entire patient encounter are treated as a single transaction. If the transaction is not completed for any reason, it can be "rolled back" automatically by issuing a ROLLBACK command which returns the database to its state prior to the attempted transaction. The transaction can then be reattempted after a predetermined waiting period. This transparent rollback feature was one main reason for selecting a transaction-oriented relational database. One reason that a transaction may not be completed is a prematurely terminated, or "dropped," connection between the client and the central database server. Since all new patient data are created on the client, a dropped connection can be handled by maintaining the data in the local database until a connection can be re-established. A patient encounter can be added to the database safely at a later time for the reasons stated above. The main disadvantage of delaying an update is that the information is not available to other providers. In general, time-critical, or "short," transactions (discussed below) cannot be delayed safely. These transactions will fail if a connection is dropped, and the data will need to be re-entered.

Graphic Application Interface

A chart metaphor. The user interface to the computer-based record is designed for ease of use and hides the underlying connection to the central database. Figure 1 shows the chart metaphor we have used for the interface. Sections of the chart are labeled at the top by colored "tabs." Clicking on these tabs using a mouse or other pointing device brings forward that section of the chart. The Patient section of the chart, for instance, is dedicated to patient demographics.

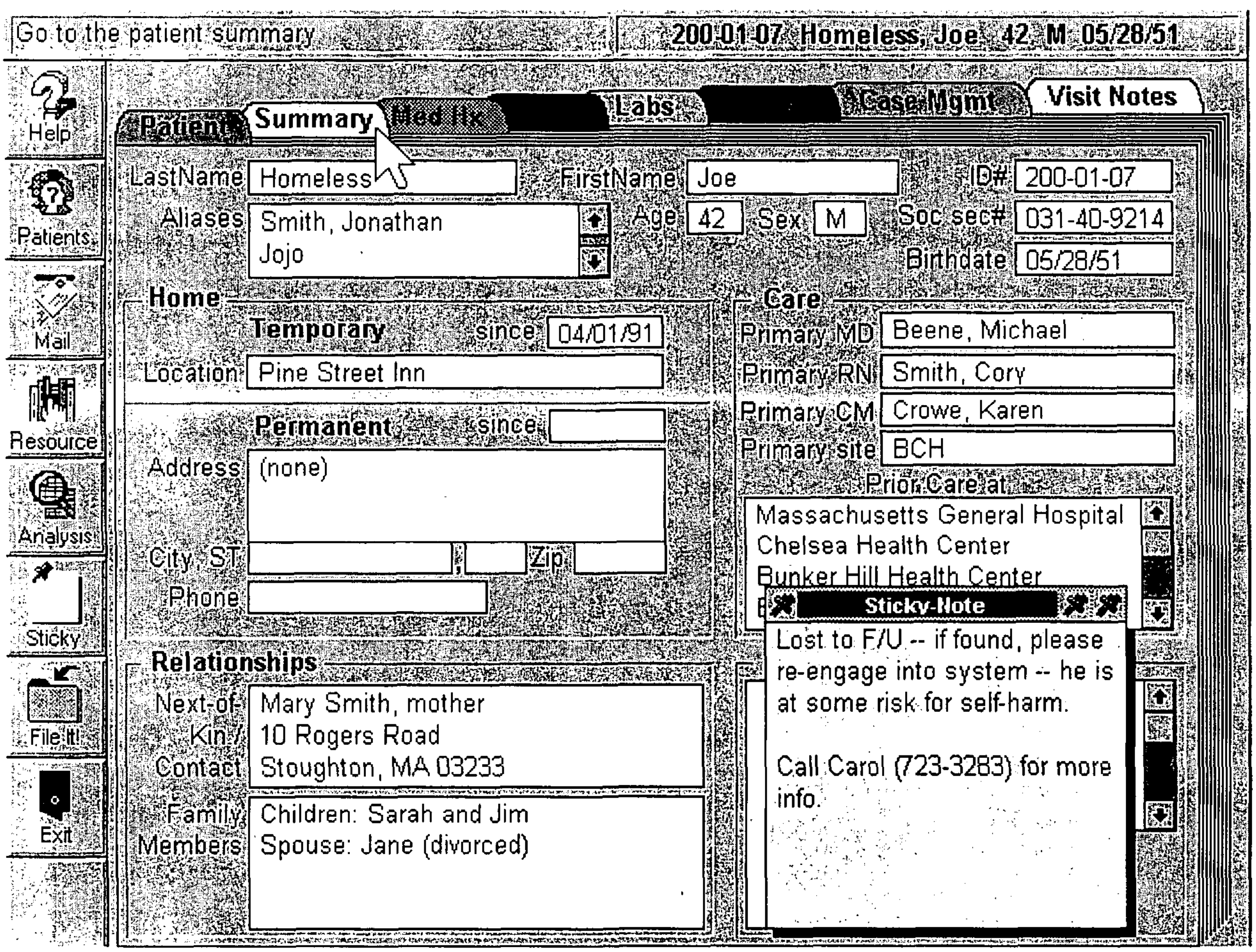

Figure 1 The graphic interface of the computer-based record is based on a chart metaphor with tabs along the top to indicate different sections of the record. The buttons along the left side are for "action" items such as retrieving a patient record or filing an encounter note. This is the Patient section of the record that shows patient demographic information.

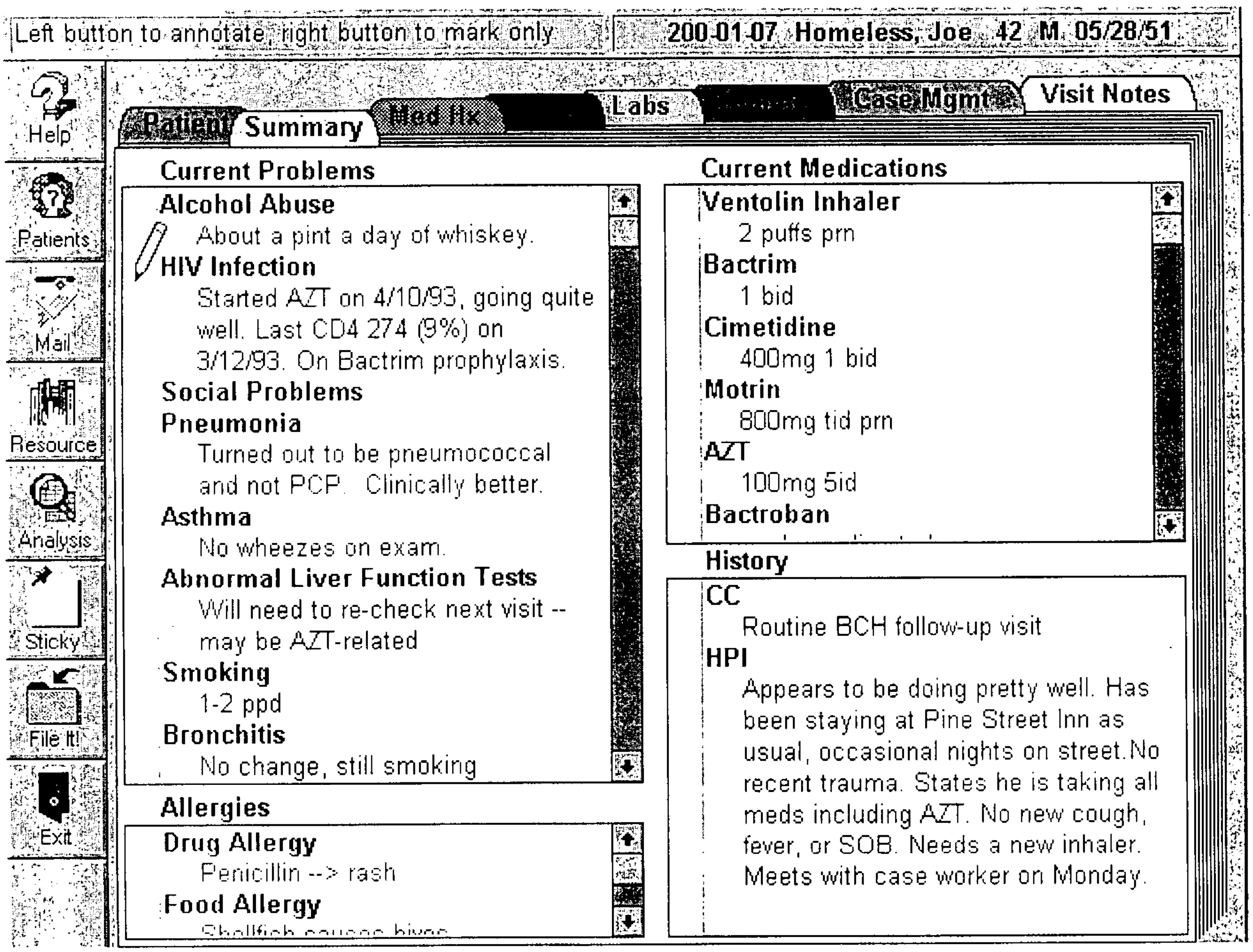

Figure 2 The Summary section of the computer-based record contains the current problem and medication lists. Notice the cursor positioned in the margin next to "HIV Infection." Pressing the mouse button at this point will bring up an editing window to annotate this topic.

As the provider moves from section to section, the appropriate patient-record information is retrieved from the central database server. Once information has been retrieved from the central database, no additional access is required because the data are buffered in the local database.[12] This allows the health-care record to be browsed interactively without subsequent delays. Repeated server access is required to retrieve additional historical information, such as previous complete visit notes or a list of all notes on a specific topic, such as "hypertension." The record is divided into logical sections including Patient Demographics, Summary, Medical History, Procedures and Therapies, Visit Notes, Case Management, Guidelines, and Labs. The graphic interface is implemented in Asymetrix Toolbook†† in Windows 3.1 with some C/C+ + routines added for speed.

††Toolbook is a registered trademark of Asymetrix Corporation.

The problem list. The Summary section is a major component of the system and consists of flexible problem and medication lists maintained by the providers (see Fig. 2). These lists are maintained over time, so that historical "threads" of particular topics are created.[13] Detailed vocabularies maintained in the local database help to validate terms that can be entered into the lists, while also maintaining maximum flexibility of data entry. To enter a new term into the problem list, a provider types a phrase of one or more word or parts of words. This phrase is then decomposed into single words, converted using stop word, acronym, and synonym vocabularies, and then matched to final terms. The problem vocabulary is based on a combination of the COSTAR[14] clinical vocabulary, the North American Nursing Diagnosis Association's (NANDA) Approved List of Nursing Diagnoses, the Nursing Intervention Classification (NIC), and the Diagnostic and Statistical Manual of Mental Disorders

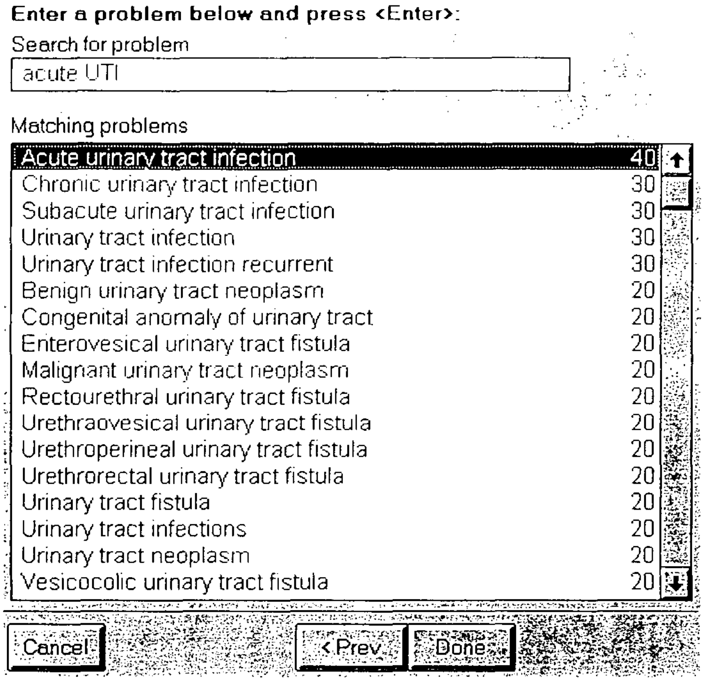

Figure 3 Over 100 possible matches of the phrase "Acute UTI" are retrieved through the use of the problem finder and its controlled vocabulary. The numbers on the right are derived from the matching algorithm and show the relative ranking of matches. These numbers are not shown to the user.

(DSM-IIIR) vocabulary. It consists of over 5,000 entries. We have chosen to use a combination of vocabularies because of the importance of giving the provider many flexible terms for direct entry into the problem list.[15] The therapy vocabulary is a combination of COSTAR terms and USP drug names. The use of standard vocabularies will help us map our recorded problems to alternate vocabularies through the use of resources such as the Unified Medical Language System (UMLS) Metathesaurus.[16] All of our current vocabularies are source members of UMLS Metathesaurus 1.2, except for some additional vocabulary terms we have identified that meet the special needs of the homeless population. The continued addition of vocabularies such as NANDA and NIC to the Metathesaurus should enhance the representation of nursing knowledge that Zielstorff and others initially found lacking.[17]

A typical search on a user-entered phrase will return a list of possible terms in less than two seconds. These terms are ranked by how closely they match the original phrase. In simplest terms, the more words that are common between the user phrase and a term, the better the match. Figure 3 shows the results of entering "Acute UTI" in the problem finder. This type of processing, where the user identifies a few key words to select from thousands of terms in a vocabulary, provides great flexibility when using a controlled vocabulary.[9] The specific matching algorithm we use is devised to work within the constraints of SQL, and though it is not detailed here, is similar to other algorithms previously described.[18] Using these word-indexed vocabulary tables to support real-time matching allows us to include many actual, descriptive problems seen in this population. Problems such as "Recently arrested," "Recovering alcoholic," and "Residence with inadequate heat" are difficult to categorize diagnostically, but can be of great importance to the overall welfare of these patients. There is no specific categorization of problems. Providers simply search for terms directly using word phrases, so these types of descriptive problems can be included easily in the vocabulary.

To support case managers, we have designed a case management section (Fig. 4). This section is analogous to the medical Summary section, except that problems entered here may be topics such as "Subway pass" and "Lack of shelter." In the patient scenario indicated in Figure 4, the social worker has negotiated an inpatient detoxification stay with the patient. Such inpatient stays often require a direct referral from a physician and cannot be delayed; that is, the referral must be made as the patient is ready to enter the facility. In such an instance, the social worker could annotate the topic "Social Problems" (refer to Fig. 2) in the problem list in the Summary section with information about the detoxification referral. The next physician to review the record with the patient present would be reminded of this opportunity to enroll the patient into the detoxification program. Including this type of information on the problem list in a centrally accessible record may increase the likelihood that follow-up actions take place, especially since the next provider to see the patient cannot be predicted with confidence.

Annotations. Providers can add a note on any topic by simply clicking the pointing device in the margin next to a specific topic. This process of "annotating" the record is the method providers use to create a new visit note while browsing the record. For example, in Figure 5 the topic "HIV Infection" has been annotated (compare with Fig. 2). The system places check marks in the margins of all annotated topics as a reminder to the provider. Annotations can be revisited and edited or removed as necessary. This minimal disruption between browsing and editing was designed as an attempt to integrate smoothly into the work flow styles of many types of providers.

No action is taken with the annotations to the record until the provider selects the Visit Note section. When this happens, the system performs two actions (Fig. 6). First, all annotations are collected and formatted into a new encounter note. Second, the most recent visit note is retrieved from the central database. This new encounter note can be edited further for more detail, perhaps when the patient has left. When the note is complete, the provider can commit it to the central database by pressing the "File It!" button.

Additional features that do require new queries to the central server include the ability to view all the notes on a single topic over time and the review of complete, previous visit notes. Since these options generally require the retrieval of a large amount of data, we retrieve this type of historical data only on demand. The delay experienced by the provider is essentially proportional to the amount of data retrieved, and can range from several seconds to half a minute.

Coordination of Databases

In this client-server, distributed-database environment, two kinds of data must be coordinated between the central host server and the remote clients: static and dynamic. Static data consist of databases such as the controlled vocabularies for problems and therapies. These can be maintained by issuing a maintenance query from the client to update the vocabularies on a periodic basis. In general, a "delta" update can be performed so that only new vocabulary items are added, and old ones removed. This allows updates to occur relatively quickly.

Dynamic data such as medical-record information can require concurrent updates to maintain database integrity and to be useful to other users of the system. Therefore, these data are handled in a transaction-based fashion, with new data committed to the database once a transaction is complete. We have chosen to implement our system to handle both "short" and "long" transactions. Short transactions are those

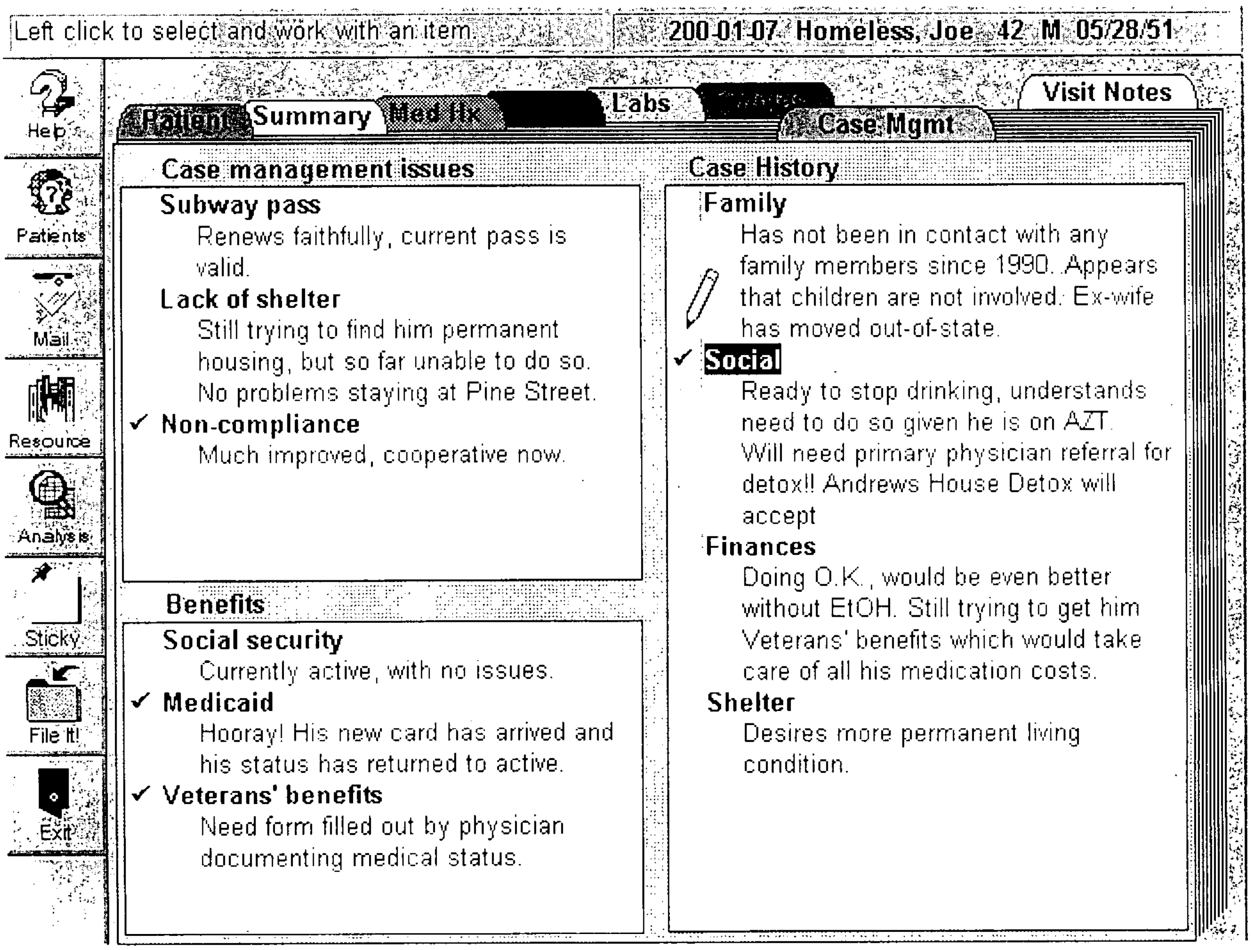

Figure 4 The Case Management section of the record, which is analogous to the Summary section, but is tailored for case workers. Notice that a Benefits list is available instead of a Medications list.

transactions that are committed to the server in real time to avoid inconsistencies in the medical record.[19] Data that are likely to have immediate use to concurrent users are treated as short transactions; for instance, the entry of lab results, e-mail messages, and the creation of a new patient record. In our client-server architecture, there is no provision for automatically posting new results on the server to all connected clients. Initially, the distribution of short transactions will occur through a simple polling mechanism. The connected client program will periodically inspect a data field on the server by using a simple query with minimal overhead. When flagged, the data field will indicate the presence of new short transaction information since logging on. If the data are high-priority and pertinent (e.g., e-mail for the provider, or information on the patient whose record the provider is browsing), they can be retrieved automatically and the user will be alerted. Otherwise, the system will simply note the presence of new data for the user to retrieve. We are exploring the possibility of using database triggers and network messaging to alert clients directly without polling. Short transactions are particularly critical in areas such as new patient identification. If providers were allowed to create a new patient record locally and to delay updating the central server, duplicate patient identifications could occur in the central database. Hammond et al.[20] and others have shown that reconciling duplicate patient entries can be error-prone and incomplete. On the other hand, a patient encounter is treated as a long transaction, since partial information about an encounter is not likely to be useful. Only after all encounter data have been entered (changes in the problem and medication lists, physical examination results, lab tests, etc.) is the encounter posted to the central server as a transaction.

Results and Discussion

The pilot stand-alone system has been installed at the BHCHP medical respite unit, and has been cap-

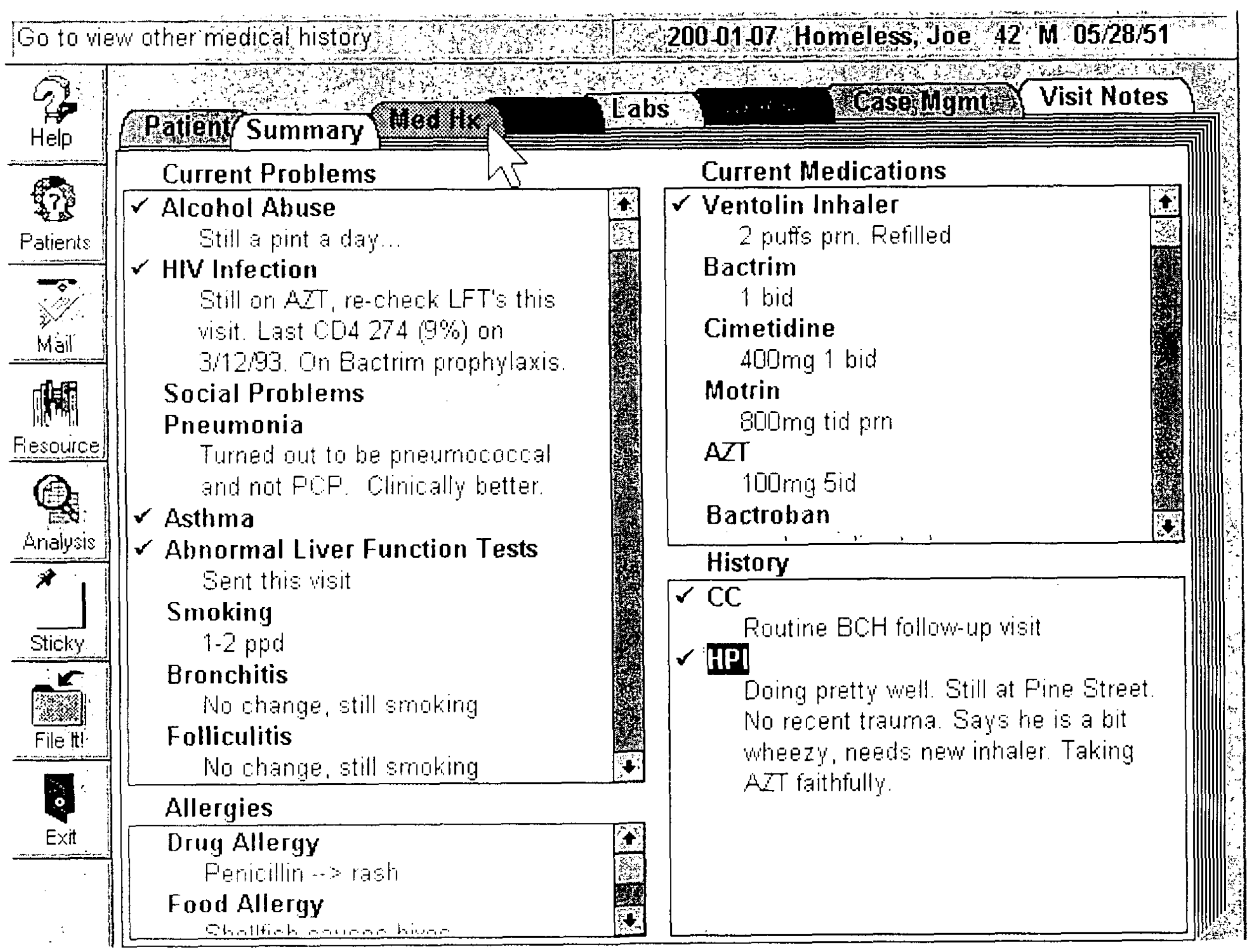

Figure 5 The Summary section showing the result of annotating "HIV Infection," as well as a number of other annotations. The check marks in the margin are created by the system as a reminder that an annotation has been made.

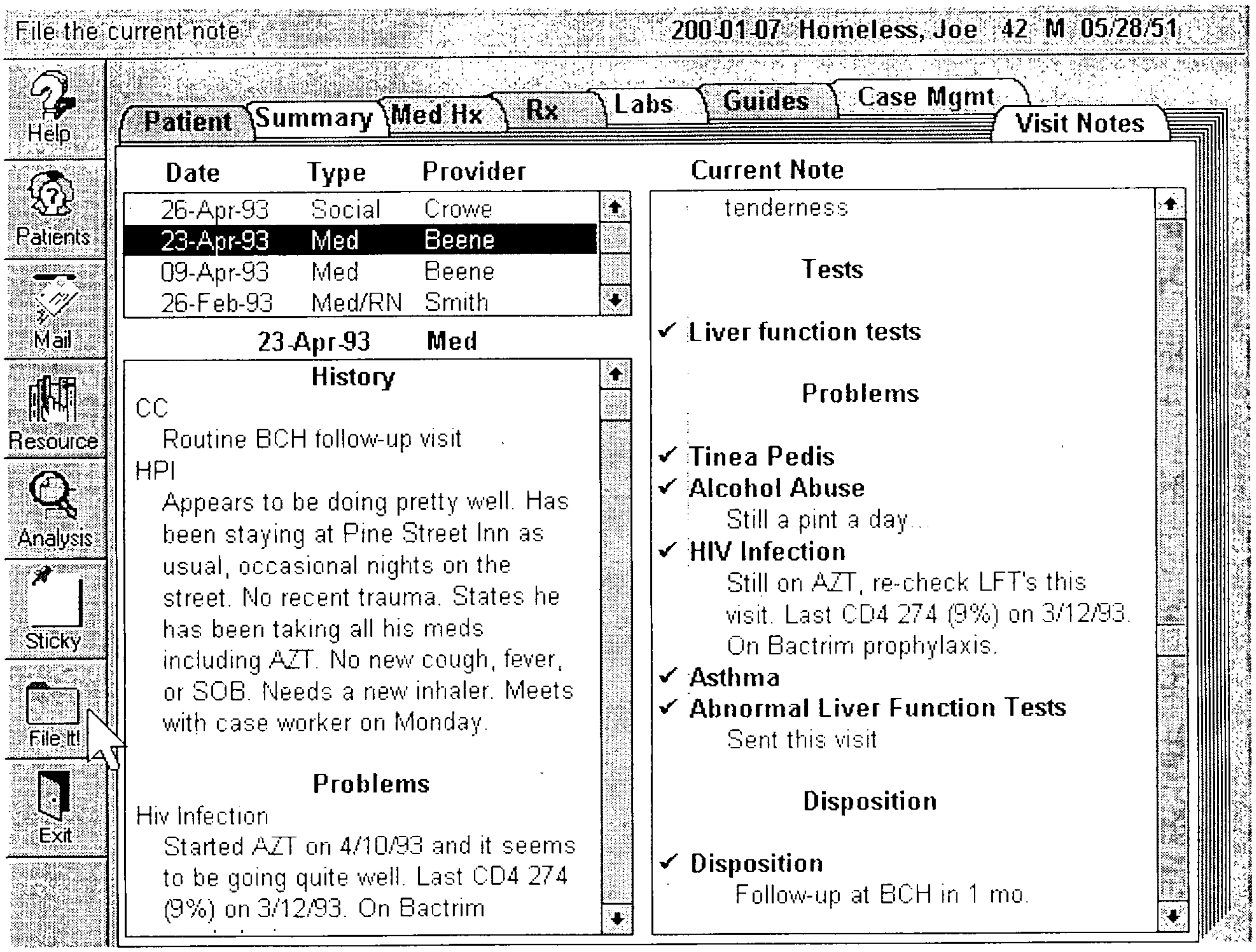

Figure 6 The Visit Note section of the record shows the annotations collected into a new, current note on the right (the note has been scrolled down to the Problems section). The most recent note is retrieved from the central database and is displayed on the left.

turing data for homeless patients. This system is available for use around the clock at the site. Both administrators and care providers can use the system, although the utility to the care providers is limited because the system does not support encounter notes. Information entered about the first 129 patients generated a half a megabyte of data. The majority of the data are entered by one person who works at the respite unit in a non-clinical capacity. The first 129 patient referrals came from 37 unique sites. Tables 1 to 3 provide details about the features of this population as captured by this system.

The more extensive client-server, remote solution is currently under active development with major components of the database schema, user interface, and distributed database strategies completed. The design of these components has been discussed in detail above. The client-server architecture over network modem links has been implemented and tested. This technology has been found to be reliable once a connection is established. Dropped connections are rare, probably because of the error-correcting feature of the modems. However, the establishment of a connection over the occasional noisy phone line can require more than one attempt.

A key issue in the design of this system is maximizing performance over the relatively low-bandwidth phone-line links between the remote clients and the central database. The use of SQL, distributed databases, and local client computing power as described above have been important for maximizing performance. However, the major factor in determining performance is the limitation of the phone-line bandwidth. Optimizing data retrieval to minimize both the number of SQL queries and the actual amount of information retrieved has been critical, even when it requires running complex queries on the database server.

Table 3 ■

Demographic Data for the 129 Homeless Patients in the Respite-unit System Database, Shown as Percentages of the Total

Description	%
Male	80
Alcohol use	47
Substance-abuse history	20
Veteran	17
Psychiatric diagnosis or history	15
Intravenous drug abuse	9
PPD +	8
HIV +	6

Figure 7 illustrates the difference in performances between a remote modem connection and a direct network connection. The performance numbers for the direct network connection were obtained over unshielded twisted-pair (UTP) Ethernet. A simple query such as a patient name lookup from an identification number is shown on the figure at the y-intercept (0 topics). These types of simple queries execute almost instantaneously on the direct link. Even when these simple queries return minimal data (in the case of the identification number lookup, only a single patient name), the remote link requires up to three seconds to complete the query. The relative performance of the remote link can be represented by the ratio: time of direct retrieval/time for remote retrieval. The higher this number, the better the relative performance—for the simple query it is 0.12. The additional two seconds is network overhead related to the low-bandwidth link and the added complexity of translating network and modem protocols for each query.

Therefore, it is preferable to use the fewest possible SQL queries to obtain the desired data. As queries become more complex, the fraction of the total query time used by the server increases. Because this time is independent of network speed, the performance of the remote link for more complicated queries relative to the direct link improves slightly. An example of such a query is the one that retrieves the current problem list for a patient. Figure 7 shows that the relative performance of the remote link for a problem list query of two topics is 0.15.

As increasing amounts of data are retrieved (using queries of equivalent complexity), the time needed for remote retrieval increases proportionally and the performance of the remote link relative to the network link degrades steadily (to an index of 0.07 at 16 topics). This occurs because of the network overhead associated with each packet of information sent over the remote link is multiplied many times over. This behavior is confirmed in Figure 7 by the diverging time points as the query retrieves four or more topics. The times for retrieving two, four, or even 16 topics using the direct network link are only negligibly different.

These benchmark figures illustrate our experience with client-server performance over low-bandwidth links: there are complex interactions between query com-

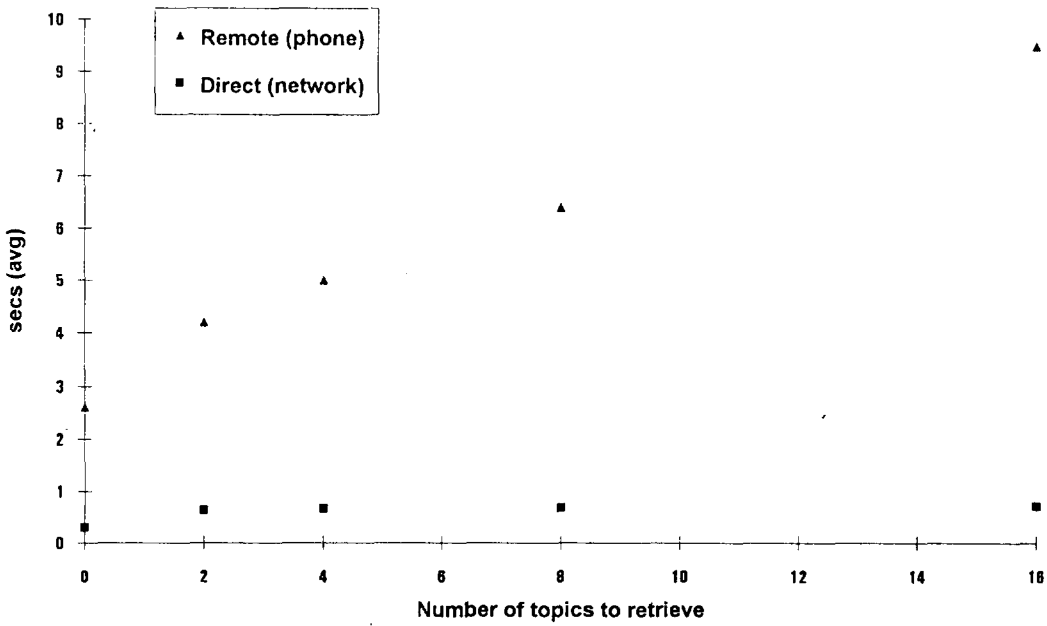

Figure 7 Benchmark data showing the performance of the remote model link compared with the direct network link. The set of points at the y-intercept (0 topics) represents a simple query (patient lookup by ID code) returning minimal data. The other points represent a more complex query (retrieving a problem list) returning more data as the number of topics increases from two to 16.

Table 4 ■

Common Retrieval-time Ranges for Different Types of Information in the Records of Homeless Patients*

Information	Typical Retrieval-time Range (sec)
Summary section	5–12
Case management section	5–12
Demographic information	3–5
Most recent visit note	5–15
History of a topic	4–10

*Real times will vary widely depending on the actual information in the records.

plexity, amount of data transferred, and actual number of queries required to complete a retrieval. Some typical performance data for common information retrieval functions are provided in Table 4. These data are provided as a broad range representing typical queries because the actual time for retrieval depends on a number of factors, including the actual amount of patient data, concurrent users, and database server caching of both queries and data.

Large-scale implementation of the complete client-server solution will proceed as the final elements of the application interface are completed, and vocabulary issues are settled. Under the current development schedule, deployment of this system is planned for the spring of 1994. The major remaining tasks include optimizing the flow of the user interface and settling important vocabulary issues, especially in the area of case management.

Reports will be created from the patient database to help BHCHP providers identify active patients for discussion at their monthly "rounds." The implementation of more dynamic "alerts" will be attempted to provide an additional level of quality assurance to providers. Currently the system uses a rule-based module for COSTAR health maintenance issues that is derived from other research with Clinical Workstation Guidelines at this laboratory.[21] These will need to be customized for the common health problems of the homeless population. The rules are documented as Arden Syntax Medical Logic Modules (MLM).[22] However, the system does not compile or interpret the MLMs directly, a technique that has been described in the design of decision support systems at Columbia–Presbyterian Medical Center.[23] Instead, the MLMs are manually recodified to operate using either a separate expert-system engine or SQL stored procedures. Other alerts will be designed to enhance communication between providers. For example, if one provider indicates in the record that a throat culture was ordered, and a predetermined number of days has passed, the next provider who retrieves that patient's record would see an alert to check the result of the culture. Such quality assurance reminders have proved useful at several different sites.[24,25]

This patient-oriented record for a homeless population may provide the potential to identify more clearly the health-care needs and problems of this population. Evaluation of the information captured through the use of the system, and the use of the system itself, will be an important next step. A study of the impact the system has on the homeless patients for which it is designed may also be enlightening. Will they find it alienating or comforting? Enhancing health-care delivery to the homeless has been a difficult task not only in Boston, but also in other urban centers around the nation. Although this system is being designed to support the needs of the BHCHP group and its patient population, we feel that the techniques (as well as the hardware and software) involved are generalizable. If the demand exists, dissemination or expansion of this system to help establish an expanded support network for these patients and their providers would be extremely appealing. Such an expanded effort would require a practical and exciting collaboration between many health-care sites.

Conclusion

Client-server and distributed-database strategies can be used to develop a computer-based health-care record system accessible from remote sites. The methods described above—use of a transaction-based central database server, controlled vocabularies on local databases, and remote networking with modems—allow remote access to information while maintaining the integrity and security of the data stored. While this project addresses the unusual demands of a homeless population, similar strategies may play an increasing role in the development of more patient-based record systems for the general population.

Dr. Chueh is supported by NLM training grant 1 T15 LM07092. The authors gratefully acknowledge support for this research by BHCHP and Hewlett-Packard Corp. Special thanks to Dyan Blewett, Judy Piggins, Mitch Feldman, and Laurie Hassan for their assistance, and to Dr. Robert Jenders for his editorial comments. Thanks also to the entire staff at the LCS, without whom this project would not be possible.

References ■

1. Stephens D, Dennis E, Toomer M, Holloway J. The diversity of case management needs for the care of homeless persons. Public Health Rep. 1991;106:15–9.
2. Gelberg L, Linn LS, Usatine RP, Smith MH. Health, homelessness, and poverty. A study of clinic users. Arch Intern Med. 1990;150:2325–30.
3. McCarty D, Argeriou M, Huebner RB, Lubran B. Alcoholism, drug abuse, and the homeless. Massachusetts Department of Public Health, Boston. Am Psych. 1991;46:1139–48.
4. Brudney K, Dobkin J. Resurgent tuberculosis in New York City. Human immunodeficiency virus, homelessness, and the decline of tuberculosis control programs. Am Rev Respir Dis. 1991;144:745–9.
5. Winkleby MA, Rockhill B, Jatulis D, Fortmann SP. The medical origins of homelessness. Am J Public Health. 1992;82:1394–8.
6. Date CJ. An Introduction to Database Systems, Volume I. Reading, MA: Addison-Wesley, 1986.
7. Friedman CF, Hripcsak G, Johnson SB, Cimino JJ, Clayton PD. A generalized relational schema for an integrated clinical patient database. Proc Fourteenth Annu Symp Computer Applications in Medical Care. 1990:335–9.
8. Johnson S, Friedman C, Cimino JJ. Conceptual data model for a central patient database. Proc Sixteenth Annu Symp Computer Applications in Medical Care. 1992:381–5.
9. Huff SM, Pryor TA, Tebbs RD. Pick from thousands: a collaborative processing model for coded data entry. Proc Sixteenth Annu Symp Computer Applications in Medical Care. 1992:104–108.
10. Wiederhold G. *Database Design*. New York: McGraw-Hill, 1983.
11. Miller S, Niedner C, London J. The organization engine: virtual data integration. Proc. Sixteenth Annu Symp Computer Applications in Medical Care. 1992:610–4.
12. Kohane IS, McCallie DP. A two-tiered database cache mechanism for a clinician's workstation. Proc Fourteenth Annu Symp Computer Applications in Medical Care. 1990:364–9.
13. Beeler W Jr, Gibbons PS, Chute CG. Development of a clinical data architecture. Proc Sixteenth Annu Symp Computer Applications in Medical Care. 1992:244–8.
14. Barnett GO. The application of computer-based medical record systems in ambulatory practice. N Engl J Med. 1984;310:1643–50.
15. Payne TH, Murphy GR, Salazar AA. How well does ICD9 represent phrases used in the medical record problem list? Proc Sixteenth Annu Symp Computer Applications in Medical Care. 1992:654–7.
16. Humphreys BL, Lindberg DA. The UMLS project: making the conceptual connection between users and the information they need. Bull Med Libr Assoc. Apr 1993;81:170–7.
17. Zielstorff RD, Cimino CC, Barnett GO. Representation of nursing terminology in the UMLS Metathesaurus: a pilot study. Proc Sixteenth Annu Symp Computer Applications in Medical Care. 1992;392–6.
18. Harbourt AM, Syed EJ, Hole WT, Kingsland LC III. The ranking algorithm of the Coach browser for the UMLS Metathesaurus. Proc Seventeenth Annu Symp Computer Applications in Medical Care. 1993:720–4.
19. Silberschatz A, Stonebraker M, Ullman J. Database systems: achievements and opportunities. Communications of the ACM. 1991;34(10):110–20.
20. Hammond WE, Straube MJ, Stead WW. The synchronization of distributed databases. Proc Sixteenth Annu Symp Computer Applications in Medical Care 1990:345–9.
21. Jenders RA, Barnett GO. Integration of health maintenance guidelines into a clinical workstation using expert system and relational database technology. Proc Seventeenth Annu Symp Computer Applications in Medical Care. 1993:885.
22. ASTM. Standard Specification for Defining and Sharing Modular Health Knowledge Bases (Arden Syntax for Medical Logic Modules). Philadelphia: American Society for Testing and Materials, 1991.
23. Hripcsak G, Cimino JJ, Johnson SB, Clayton PD. The Columbia–Presbyterian Medical Center decision-support system as a model for implementing the Arden Syntax. Proc Fifteenth Annu Symp Computer Applications in Medical Care. 1991:248–52.
24. Barnett GO, Winickoff RN, Dorsey JL, et al. Quality assurance through automated monitoring and concurrent feedback using a computer-based medical information system. Med Care. 1978;16:962–70.
25. McDonald CJ, Hui SL, Tierney WM. Effects of computer reminders for influenza vaccination on morbidity during influenza epidemics. MD Comput. 1992;9:304–12.