Application of Technology ■

Willow: A Uniform Search Interface

DEBRA S. KETCHELL, ML, MATTHEW M. FREEDMAN, MS, WILLIAM E. JORDAN, MA, ML, EDWARD M. LIGHTFOOT, SCOTT HEYANO, PETER A. LIBBEY

Abstract The objective of the Willow Project is to develop a uniform search interface that allows a diverse community of users to retrieve information from heterogeneous network-based information resources. Willow separates the user interface from the database management or information retrieval system. It provides a graphic user interface to a variety of information resources residing on diverse hosts, and using different search engines and idiomatic query languages through networked-based client–server and Transmission Control Protocol/Internet Protocol (TCP/IP) protocols. It is based on a "database driver" model, which allows new database hosts to be added without altering Willow itself. Willow employs a multimedia extension mechanism to launch external viewers to handle data in almost any form. Drivers are currently available for a local BRS/SEARCH system and the Z39.50 protocol. Students, faculty, clinicians, and researchers at the University of Washington are currently offered 30 local and remote databases via Willow. They conduct more than 250,000 sessions a month in libraries, medical centers and clinics, laboratories, and offices, and from home. The Massachusetts Institute of Technology is implementing Willow as its uniform search interface to Z39.50 hosts.

■ **JAMIA**. 1996;3:27–37.

In 1989, information retrieval was neither open nor unified. It was cumbersome and complicated. Information resources were available to users through a variety of heterogeneous interfaces from a mixed profusion of local stand-alone and remote vendor systems using terminal access. Information resource vendors such as the National Library of Medicine (NLM), DIALOG,[1] OCLC,[2] SilverPlatter,[3] OVID,[4] and RLG[5] were not accessible over the Internet. The Z39.50 protocol, an early attempt to provide interoperability between information retrieval systems, had just been adopted. Client–server Internet navigation and retrieval tools, such as Gopher, World Wide Web, and Wide Area Information Services (WAIS),[6] were an unrealized potential.

That was the environment when a collaborative team of computing and library staff at the University of Washington (UW) began to develop a general-purpose retrieval tool to help users deal with the burgeoning collection of disparate information systems using client–server architecture, Transmission Control Project/Internet Protocol (TCP/IP) communication standards, and a graphic user interface. This vision of a uniform search interface became the Willow (Washington Information Looker-upper Layered Over Windows) Project. Now, Willow provides faculty, staff, and students ubiquitous access to more than 30 licensed or public domain databases from medicine to anthropology to engineering from a variety of sources.

In this article we describe design considerations, system design, and utilization of the Willow program. We then focus on lessons learned in the process of building a uniform search interface and promising future work.

Affiliations of the authors: Health Sciences Libraries and Information Center (DSK), University of Washington Libraries (WEJ), and Computing & Communications (MMF, EML, SH, PAL), University of Washington, Seattle, WA.

Presented in part at the Sixteenth Annual Symposium on Computer Applications in Medical Care, Baltimore, Maryland, 1992.

Willow prototype was supported in part by Digital Equipment Corporation. HealthTerms prototype was supported by the NLM (contract 467-MZ201782).

Correspondence and reprints: Debra S. Ketchell, ML, Health Sciences Libraries and Information Center, Box 357155, University of Washington, Seattle, WA 98195-7155. e-mail: ketchell@u.washington.edu

Received for publication: 5/25/95; accepted for publication: 9/22/95.

Background

Willow represents one of many possible approaches to facilitating access to the rapidly expanding set of network-based information resources. One possible approach to the problem is a "metaguide" or "connection manager" that helps users to find and then access information resources. Examples of this approach include the BioSYNTHESIS[7] system developed at Georgetown University and the Gopher[8] protocol developed at the University of Minnesota. OCLC's Spectrum[9] and the NLM's Unified Medical Language System's (UMLS's) Information Sources Map (ISM)[10] provide a method of building a database of information resources. Yale University's NetMenu[11] pilot of the ISM allows the user to identify and connect to biomedical resources based on locality, subject, cost, priority, availability, and other parameters. While the metaguide approach aids in discovery and connection, it forces the user to adapt to disparate search and retrieval interfaces.

A second approach centers on isolating the user from different search and retrieval idioms, either by "translating" multiple existing interfaces into a single interface or by using a standard protocol to exchange queries and results. Examples of the "translator" approach include the Intelligent Query Workstation[12] at the Massachusetts General Hospital and an increasing number of World Wide Web (Web) "gateways." Protocol solutions include SilverPlatter's proprietary DXP and the American National Standards Institute/National Information Standards Organization (ANSI/NISO) Z39.50 Information Retrieval Service and Protocol Standard.[6,13]

Z39.50 provides an interoperable method to search databases, retrieve records, and perform ancillary tasks related to information retrieval. It is platform- and operating-system-independent client–server protocol that addresses communication between an origin (client) and a target (server). It does not define user-interface behavior. Z39.50-1995 includes both version 2 and version 3 of the standard and is compatible with Z39.50-1992 (version 2). Many organizations have used the Z39.50 protocol to integrate local and network databases, including the University of California's MELVYL[14] system, Penn State University,[15] NOTIS,[16] and OCLC.[17]

Design Considerations

Willow[18] was envisioned as a tool to simplify information retrieval through a uniform interface to heterogeneous information resources available locally or via the Internet. The technical architecture for this uniform interface would be flexible and transportable rather than tied to a specific workstation or search engine. For the user, the interface would be a tool to simplify query formulation (including the use of terminology assistance to increase precision and recall) and then produce a summary of matches to either aid query refinement or allow selection of full text that could be saved for individual use.

Specific design objectives have evolved over the course of the project, and include:

- Concentrate first on resource available locally using the BRS/SEARCH software (Dataware Technologies, Cambridge, MA) and remotely using the Z39.50 protocol.
- Keep the interface simple. The goal would be to use a search and retrieval paradigm that would work well for many, but not all, users—form-based, iterative search refinement.
- Use a graphic user interface to increase user productivity, reduce the need for training and support, and increase the use of advanced features.
- Provide modes suitable for both anonymous access in a public terminal environment and customization in an authenticated environment.
- Ensure accountability of database use for licensed resources.
- Capitalize on uniform services supported by a computing environment based on TCP/IP protocols and client–server architecture.
- Recognize the lack of a common user workstation. The Willow architecture would be UNIX-based, and the initial interface would be built for X-Windows (using X-Motif style guidelines) for access on X-terminals in central locations. Any user could potentially have access on his or her workstation through the use of X-emulation software.

The Willow design process has been an iterative cycle of prototype, test, and redesign based on user feedback.

System Design

Willow is based on a client–server architecture using TCP/IP communications protocols. Figure 1 illustrates the major components of this architecture: 1) the interface, 2) the database driver, and 3) the database server. The advantages of this design are extensibility (add new drivers for new protocols or database serv-

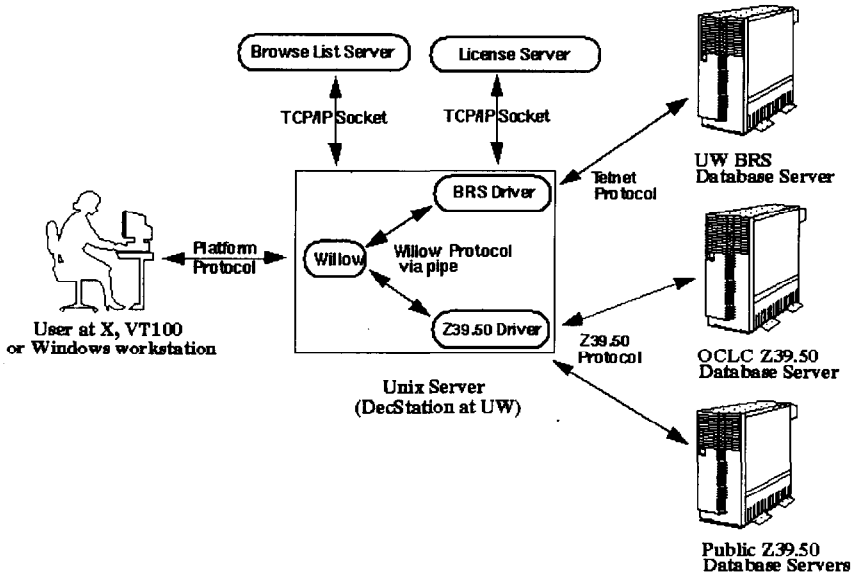

Figure 1 The Willow client–server architecture.

ers) and easier cross-platform support (one only has to port the interface). Originally developed on the Digital Equipment Corporation (DEC) Ultrix operating system, precompiled binaries are also available for Sun Solaris and SunOS, IBM RS6000, and Silicon Graphics. Willow has also been ported to DEC OSF/1 and HP/UX. The source code is copyrighted, but freely available for any noncommercial purpose. The components of the Willow architecture and a view from the user's perspective are described in this section. Detailed technical documentation[19,20] and a simulation of Willow in operation are also available.

The Database Driver

The Willow interface (X, VT-100, or Windows versions) itself does not know anything about heterogeneous database hosts. Host-specific routines are localized in a second program, the database driver, that communicates with a particular database host to make the connection, interpret command syntax, and parse the incoming data stream. Willow and the driver communicate via an internal protocol using standard interprocess communication mechanisms—UNIX pipes (X and VT-100 versions) and TCP/IP sockets.

A database driver can be written to communicate with any database host. Willow comes with a model driver, which is designed to be taken apart and rebuilt for this purpose. However, it is by no means a simple toolkit. Writing a custom database driver is not a trivial undertaking. The original driver was developed for the local BRS/SEARCH system implementation in 1989, and provides access to a set of UW databases, including MEDLINE and the Libraries Catalog. The plan for a Z39.50 protocol driver was deferred until 1993 due to a lack of Internet-accessible databases. The current Z39.50 version 2 driver is a result of a collaboration between the authors' group at UW and colleagues at the Massachusetts Institute of Technology (MIT). It is based on the Stanford Z39.50 API (Application Program Interface) library and is interoperable with version 2 and 3 servers.

At UW, the databases currently available to the public through the BRS driver are MEDLINE, Cumulative Index to Nursing & Allied Health Literature (CINAHL), PsycINFO, Libraries Catalog, University Handbook, UW Course Catalog, Legal Resources Index, ERIC, Grolier's Encyclopedia, INSPEC, Books in Print, Publisher's Directory, Business Index, Expanded Academic Index, Newspaper Index, and Company Profiles. Public databases through the Z39.50 driver are OCLC WorldCat, Avery Index to Architectural Periodicals, Anthropological Literature, PAIS, MLA Bibliography and public domain databases (e.g., CIA World FactBook and Library of Congress Catalog). Experimental databases include the NLM's Z39.50 MEDLINE, UW Photos, Elsevier's TULIP full-image articles, several OCLC databases, several small internal databases and a number of Z39.50 library catalogs.

Database Configuration File

Beyond the database driver, each individual database has an external configuration file containing the specific details available to the user such as field labels,

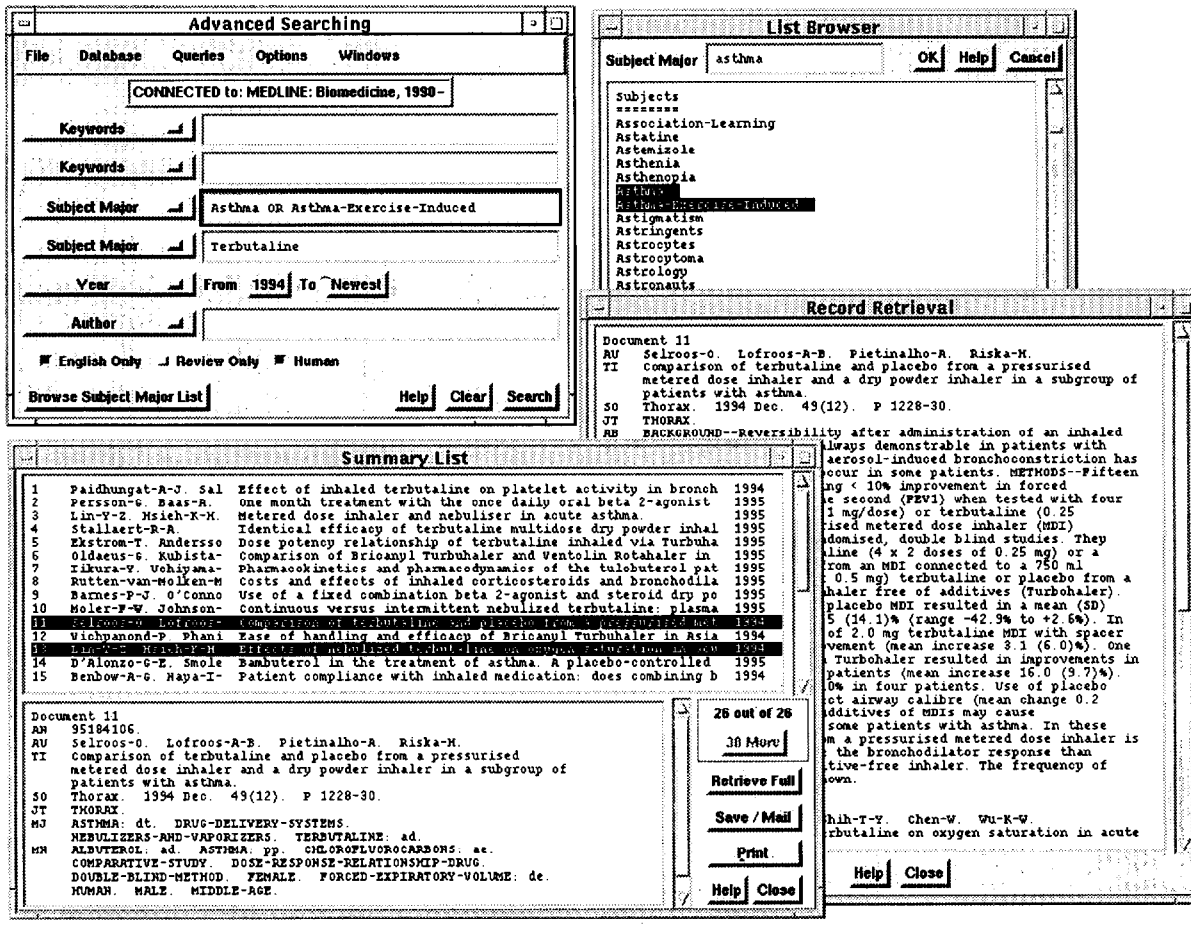

Figure 2 The four windows of the X-Motif version of Willow.

unique search window attributes, and browse lists. These configuration files may be located on the same server as the database drivers or may be retrieved from a remote configuration server via the Hypertext Transfer Protocol[21] (HTTP) of the Web. Any Web server can be a database configuration server. The Z39.50 Explain service is expected to eventually eliminate the need for manually prepared configuration files. Using Explain, Willow will ask the database driver for meta-information about its databases and, based on that information, Willow will dynamically configure itself. Integration of Explain capabilities into Willow is in its early stages at MIT.

Browse Lists

Willow includes a list-browsing mechanism to permit users to browse an ordered list of terms for more precise query formulation. Users navigate the list browser by typing the first few characters of a term or by using the scrollbar to move back and forth in the list. Selected terms are automatically transferred to the Search window. Originally developed for browsing controlled-vocabulary subject terms, the list browser has been extended to sort lists in a variety of formats, including a multifield display.

For UW databases, Willow typically offers alphabetic lists to select subject terms, publication types, languages, journal titles, and authors. Some lists display the number of matches for a term in a database. The Libraries Catalog call number list presents title, location, and call number data ordered by call number, which provides the user with a "virtual" shelf of materials in more than 20 library locations. This local browse list service presents the user with a formatted list containing punctuation while returning a normalized form to the search system. This allows Libraries Catalog searchers to select from an easy-to-read display of authors and titles instead of a list of prebound phrases optimized for machine performance.

Because the native BRS index browse was deemed inadequate, the browse lists for databases loaded locally are maintained on a separate list server, essentially duplicating each of the indexes for which browse

service is desired. The list server has been optimized for very fast response in order to support the scrollable, incremental browse interface in Willow. Further, the query terms returned to Willow have been optimized for the BRS/SEARCH engine by inserting hyphens between words in an index phrase.

The Z39.50 protocol provides a generalized browse facility through its Scan service. Scan performance varies considerably from host to host. With some database hosts, scanning terms in specific fields is a relatively expensive operation. Some hosts are designed in such a way that implementing a scrollable browse list is almost impossible, and few are fast enough to support incremental browse on each keystroke. The Z39.50 protocol now supports both "display string" and "search string" like our local list server, but no host has yet implemented this feature.

Multimedia Extensions

Although Willow handles data only in the form of plain ASCII text, a multimedia extensions mechanism allows an external viewer to be launched to handle image, sound, movie, Standardized General Markup Language (SGML), non-Roman language character sets, and other types of data. One example is the ability to search the Libraries Catalog to locate holdings with record variants in Chinese, Japanese, or Korean scripts, and then view the non-Roman version by pressing a botton that launches a custom viewer. As a participant in the Elsevier TULIP Project, we prototyped retrieval of a full-image journal article. The user locates a record in a database linked to these images, and then retrieves the full-image article by pressing a button that launches a TIFF image viewer. The user can then navigate through the pages and zoom in for a finer display as well as send the image to a postscript printer or a fax machine.

Online Help

Help is available at each stage of search and retrieval. Currently, the X version offers a hierarchical menu of bitmap images and the VT-100 version a flat list of topics for general interface help and database-specific assistance for formulating queries. A new model to provide help is discussed in Lessons Learned.

License Manager and Authentication

To meet the objective of "open" access (no userid/password required) at terminals in the libraries and medical centers, we implemented a general-purpose network license server. This server created pools of accounts for walk-up use. When the user selects a database, the database driver transparently contacts the license server and requests and receives an account and password for that database or search system. The license server regulates a specific number of users for a particular search system host (e.g., OCLC) or a specific database (e.g., the local Books in Print database, which is restricted to a limited number of users) according to current licensing contracts. For increased security, passwords are created on the fly by the license server and are valid for only 30 seconds.

More generally, the license server allows definition of several classes of IP addresses and restricts access based on the incoming address. This provides open access for UW addresses to all databases, and a limited number of sessions over the Internet to the Libraries Catalog. To avoid lockout of UW faculty, staff, and students connecting from addresses beyond the UW, the user is given an opportunity to override a rejection message with a valid UW userid/password.

As has been stated, users connecting with valid UW IP addresses are not required to login with a userid and password. However, anonymous use means that preferences cannot be saved across sessions. All versions of Willow allow the user to save preferences, search strategies, and retrieval to their account or computer. Willow can also be configured to prompt the user for a personal userid for each database or search system, although this feature is not currently in service at UW.

Willow in Operation

To the user, Willow is composed of four "windows," as illustrated in Figure 2. The Search window is the anchor window with the menu bar. A database is chosen from the Database menu and then the query is built by entering terms on the form lines. Field labels (e.g., subject, journal title) for each line are changeable using the pull-down menu on each button. The availability of a browse list for selecting appropriate terminology for a particular field is highlighted when a form line is selected. One or more terms may be selected from a browse list and will be automatically transferred in the correct format to the form line. Simple limits (e.g., English, human) appropriate for each database are highlighted for selection at the bottom of the Search window. Pressing the Return key instructs Willow to translate the query into language understood by a particular database host. A one-line description (e.g., author, title, date) of matched records is displayed in the Summary window. Clicking or scrolling through the title list dis-

plays a brief record (e.g., full author, title, and source information) to enhance the review process. Text from the brief record may be copied back to the Search window to refine the query. Search strategies may be saved for later use.

The user may select one or many titles and retrieve the full record results. Records may be printed or saved. Save options include sending to an e-mail address, saving to an e-mail folder, or transferring a file (e.g., Zmodem or FTP). The e-mail function allows users to send results to special addresses where the text is either faxed to a specified phone number or printed and mailed to a specified campus box number. The user may select and save options such as field label type, field selection, inclusion of search strategy, and print and save defaults. Willow also provides version-specific customization—fonts, colors, etc.

Utilization

Interface Evaluation

We solicit user feedback through field testing, formal usability evaluations, library staff observations, and several e-mail comment options. A comment button is always displayed when searching Willow in the libraries, and the e-mail address is prominently publicized. This mail is tracked to identify specific problems and provide a means to see trends. Many users have become directly involved in the redesign by describing their workflow or approach to search and retrieval.

An evaluation of the prototype by user observation of nine faculty and six librarians in 1990 determined that the design was viable. In 1992 the beta version was evaluated to determine usability for "walk-up" users at public stations in the library. A think-aloud protocol with a postsearch questionnaire was used with 120 students at six sites over a four-week period. Sixty percent of the subjects felt the interface was easy to use, and 54% felt they could determine what to do by looking at the screen. Questionnaires and observations highlighted preferred features and usability trends. The most significant problem was start-up assistance. A default setup for the library X-terminals was developed to provide a desktop that cannot be randomly changed, easy database selection, start-up help, and a permanent button bar to reset the terminal, print any selected text, or send a comment.

Later in 1992, a usability study was conducted to determine Willow's effectiveness as a Libraries Catalog interface, particularly for simple author or title search (the most common use). The think-aloud protocol test found that the form-based Search window was too complicated and required too much time for quick catalog look-ups. A new look-up mode was designed, as illustrated in Figure 3. After implementation, library staff observed that users required less assistance and asked fewer questions.

A usability study and general field testing in 1993 found that the VT-100 version was easy to use for a nongraphic interface. However, these tests did not include users who had been searching MEDLINE through the original BRS command interface since 1989 and were unfamiliar with the Willow search model. Librarians worked directly with these users (primarily health sciences faculty) to translate from the old to the new model. Since many of these users imported references into personal reference managers, the record format and FTP options were simplified. Despite a small but vocal negative reaction, more than 60% of the MEDLINE and more than 95%

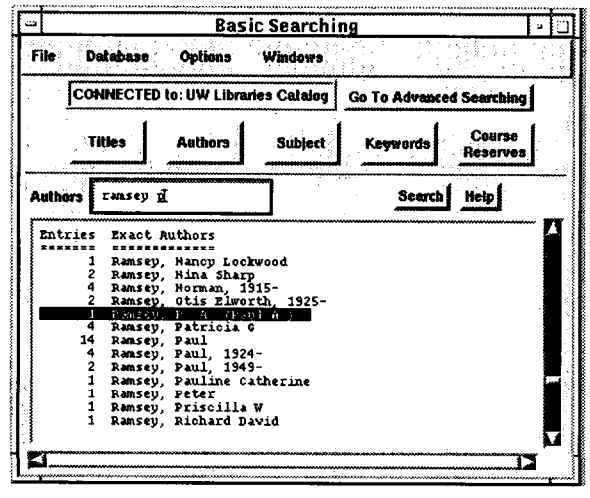

Figure 3 Willow has a basic look-up mode designed specifically for searching library catalogs. While all three stages are displayed here, the user interacts with an expanding window as the look-up progresses. Initially, the user sees only the top row of buttons (Titles, Authors, Subject, Keywords, and Course Reserves) and selects the type of search desired. The window then expands to provide the user with a text box in which to enter the first few characters of the item desired. The window expands again to display a browse list, which highlights the closest match to the user's input. The user may choose to scroll up or down the list (from beginning to end) or enter a new string of characters in the text box if the term desired is not displayed in the window. To retrieve one or more items, the user selects from the list and clicks the Search button. If a single item was selected, the full record is then displayed. If more than one term was selected, the standard Summary List window is displayed.

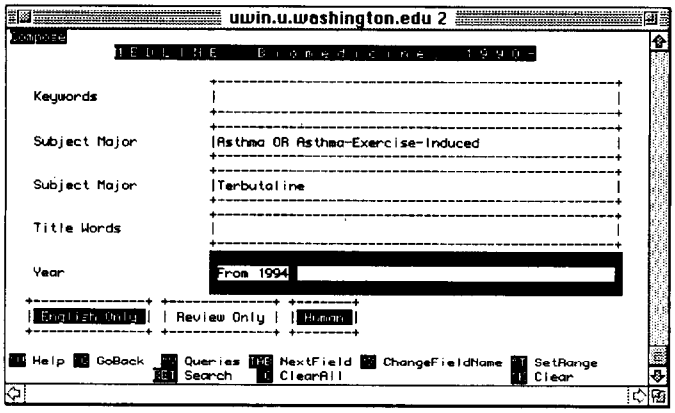

Figure 4 The Search window of the VT-100 interface.

of other database users moved to the VT-100 Willow within two months.

The Windows version of Willow is currently in beta test with faculty. Since a platform-specific version is desired by users and the Willow search model is the same, the majority of testing has focused on the ability of users to retrieve and install the software on their desktops.

Current Availability

Willow deployment is ubiquitous at the UW. Centrally administered server clusters support more than 300 X-terminals in more than 20 local and remote library sites; more than 600 X-terminals in the two medical centers and affiliated clinics; and campus-wide access for individuals using X-emulation software (e.g., MacX or Reflection). The VT-100 version, illustrated in Figure 4, supports low-end modem connections. The Windows version, illustrated in Figure 5, and MacX provide a graphic interface for "dial-ip" users with TCP/IP software and high-speed modems.

Usage

Usage has increased in surges followed by plateaus over the last six years. From July 1994 to June 1995, there was an average of 212 simultaneous users. During July 1995, the figure climbed to 237. The highest maximum number of simultaneous users reached 478 in February 1995. The average search session lasts approximately 14 minutes based on discrete connections. Usage between versions was split 56% X and 44% VT-100 during 1994–95. The higher use on X is not surprising considering that more than 900 terminals are available in the libraries and medical centers. Within the health sciences there has been an increase in the use of the VT-100 connections as the large number of affiliated clinical faculty take advantage of these databases. We anticipate a significant increase in the use of the Windows version from both office and home. MEDLINE and the Libraries Catalog databases are the most heavily used. During July 1995 the simultaneous users by database totaled 34% UW Catalog, 23% MEDLINE, 12% WorldCat, and 31% other databases.

MIT and the California Institute of Technology (CalTech) are among other institutions using or preparing to use Willow as part of their campus-wide information strategies. MIT deploys Willow on about 1000 DECStation/Ultrix, Sun/Solaris, SGI/Irix, and IBM/AIX workstations across campus. These machines are part of a campus infrastructure that supports student, faculty, and staff computing needs. MEDLINE, Avery Index, and TULIP are production databases, and 20 more OCLC First Search/Z39.50 databases are under development. CalTech is testing Willow for access to journal table of content indexing.

We expect an increase in Willow usage and deployment as public-domain Z39.50 server packages become available. A promising development is the appearance of the Clearinghouse for Networked Information Discovery and Retrieval's (CNIDR) ISite[22] software. ISite includes a Z39.50 protocol engine, an indexing tool, and an integrated search engine that potentially will allow smaller (e.g., departmental) information resources to be searched using Willow, essentially integrating personal/departmental, campus-wide, and remote databases under a single interface. At the UW, several small internal databases are being tested (e.g., indexes to environmental impact statements and microform sets). CNIDR's Internet-accessible AIDS Patents[23] database demon-

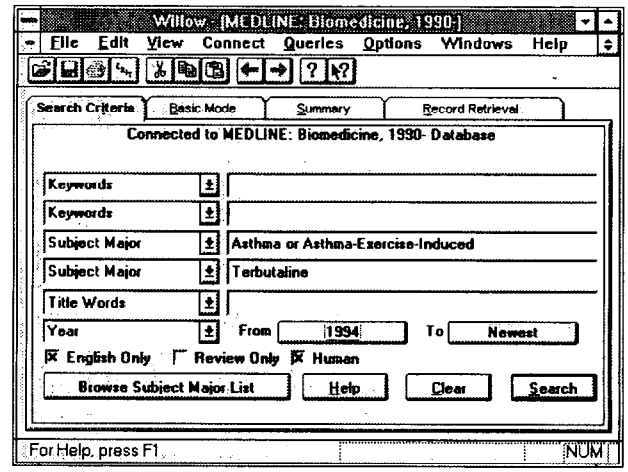

Figure 5 The Search window of the Windows interface.

strates that even moderately large files can be made accessible in this way.

Lessions Learned

Designing for Performance

We made design compromises in favor of overall system response as the number of simultaneous sessions grew over the course of the project. For example, Willow was designed to retrieve all results for a query for user browsing. This design placed a heavy burden on the back-end search engine (in this case UW BRS system), which affected overall user response rates. We then redesigned Willow to retrieve only the first 30 results, with the option of retrieving more in blocks of 30. This limit has minimal impact for users doing quick look-ups or more advanced searchers working through an optimum strategy. It does hamper the effectiveness of researchers' downloading large retrieval sets by reducing their efficiency, particularly reducing the effectiveness of those who intend to review the information later using a word processor or to import the data into a personal reference manager. Limiting the quantity of initial retrieval is frequently designed into heavily used Internet search tools such as Lycos[24] and WebCrawler.[25] The rapid advances and lower costs in raw computing power and new search engine technology should eliminate performance handicaps that restrict design optimization.

Designing for the Majority

We adopted two design rules: keep it simple and the 80% solution (i.e., satisfaction of 80% of the users). We did not want to become feature-driven, nor did we want to design for a small group of power users. For example, the use of form-based query formulation was selected in 1989 based on experience in teaching health sciences faculty and students to use the NLM's Grateful Med software. Grateful Med had proven to be a good interface for novice and intermediate MEDLINE use compared with the local command-based search system. As noted earlier, a small group of users who prefer the direct command approach (e.g., raw DOS and UNIX) have responded negatively to this query model. New graphic desktops and the emergence of form-based searching on the Web have strengthened the familiarity of our form-based model. Experience in teaching users and observation of users' independent use at public terminals confirmed our design expectations. However, simultaneous searching across databases and a simple approach to search syntax (e.g., Boolean and positional operators) while maintaining overall simplicity is still an unrealized goal.

Initially prototyped for "abstract and index" databases, Willow was redesigned to meet the needs of library catalog users. The form-based search did not match the simple, routine look-ups that most users expect when approaching a library catalog. We designed a catalog look-up mode for the three most common searches (Fig. 3). This mode also includes a simple approach to finding reserve materials by course name or instructor. The catalog data are searched using the local BRS system; the real-time circulation and serial check-in data are retrieved on demand from our Innovative Interfaces library system and integrated seamlessly into Willow's retrieval display for the user.

Optimizing across Information Resources

One of the benefits of our separation of interface, database driver, and database host is the ability to move to a new search engine without a major impact on users. The burden of reorientation is placed on the developers and not on the users. However, there is always tension between the quest for performance (and attendant target-specific optimizations) and a desire to insulate the user from the peculiarities of individual systems. Integrating heterogeneous information resources into a uniform interface means that we cannot always provide the optimum search model for each resource.

Aggravating the tension is a lack of normalization rules for search terms. The user is required to provide terms in a syntax that may vary from database to database as well as between search engines. For example, an author search may need to be expressed as:

UW MEDLINE	smith-j
UW ERIC	smith-john
OCLC'sWorldCat	smith, john bernhard 1858–1912
RLG's Avery Index	smith, john

Z39.50 provides a "normalized name" attribute, but essentially relies on out-of-band discovery of the normalizing conventions in use for a particular host. Our UW BRS implementation will accept unhyphenated authors in some databases, but the search runs much more slowly. In RLG's Avery Index, author terms without commas are interpreted as "firstname lastname." In OCLC's WorldCat, two spaces are required between the author's name and dates in the author phrase index. It is difficult to insulate the user

from this type of variability without crippling the user's options. It also makes implementation of enhancements such as simultaneous searching of multiple databases problematic.

For databases maintained at UW, we can minimize this impact by providing both word and phrase indexing and supporting positional operators. We do not control indexing practices or proximity searching of remote databases and must rely on term lists and the Z39.50 Scan service. Simple search strategies also decrease the amount of system-specific knowledge the user must have, but many users have information needs that are poorly served by simple strategies.

Building Cross-platform Access

In 1989 the X-Motif interface was selected to distribute Willow in libraries and medical centers on X-terminals to keep maintenance and support costs low. We anticipated that Macintosh and Windows users would employ X-emulation software to access this interface. However, this interface did not address the needs of users with slow dial-in connections from home. We investigated cross-platform development toolkits at that time and found that the least common denominator approach would not meet user expectations, and in 1993 began developing a VT-100 interface. This interface remains a core product as a safety net to support a variety of end-user devices.

Similar to the early dissemination of Mosaic (early availability for X-Motif only), the X interface diffused very slowly into offices. Macintosh users could use site-licensed Apple software (MacTCP and MacX) to put Willow on their desktops. A Windows X-emulation solution was both more expensive and complex due to the number of possible software configurations and the need for a fast processor (e.g., 386, preferably 486) and ample memory (e.g., 8Mb, preferably 16Mb). The availability of uniform access to dial-ip services [Serial Line Interface Protocol (SLIP) and Point-to-Point Protocol (PPP)], more powerful home and office desktop computers, and the Web have intensified Macintosh and Windows users' expectations. By 1994 we began developing a Windows version that would be compatible with Windows95 and WindowsNT desktops. We ported the Windows code to a Macintosh version in early 1994 with unsatisfactory results and are now investigating a second conversion based on new code.

In 1995 we also prototyped a Web version of Willow and determined that user functionality would be sacrificed. Although the Hypertext Markup Language[21] (HTML) syntax allows the creation of forms for various types of user input, it has considerable limitations. Only a small number of interface objects are currently supported, and no type-checking is possible within a particular field. There is also no mechanism for creating dependencies between interface objects. For example, if a user chooses a particular database, there is no way to change other interface objects to reflect those options available only for that particular database. Other limitations include the lack of a robust method for maintaining state information and problems with user authentication and security. Emerging technologies such as Sun's HotJava[26] may provide the mobile Web code to recreate Willow as a browser-based application. In the meantime, we are imbedding HTML widgets into the Willow code to format Web-based retrieval display and help information, as well as integrating multimedia extensions.

Managing Support Demands

Early in our collaboration, the Libraries and Computing & Communications (C&C) divided major interface responsibilities. C&C staff provide the search software and hardware, load and maintain the databases, program Willow, and support the uniform access campus and dial-in services for ubiquitous faculty, staff, and student access. The Libraries staff license and profile databases, create and distribute documentation, and provide training. The Willow application itself is developed as a collaborative project. Central e-mail addresses and referrals between the collaborators are essential as users go to the most comfortable source for assistance.

Significant resources are invested in creating and distributing printed and online help as well as providing personal assistance and training. Printed documentation is still needed for some first-time users, particularly remote users. Several models have been used to reduce staff and printing costs. We create getting-started, basic search techniques, and database-specific documentation in printed (single, color-coded $8\frac{1}{2} \times 11$ sheets) and HTML formats. Within the health sciences there has been a particular demand for information about how to import retrieval into reference managers. A new model for online help across interface versions is under development, as it is too labor-intensive to create and maintain a distinct set of help files for each platform. We have developed an experimental design that launches an external Web browser to display help files in HTML format based on the existing multimedia extension mechanism. Another promising discussion calls for imbedding HTML display widgets within the interface itself.

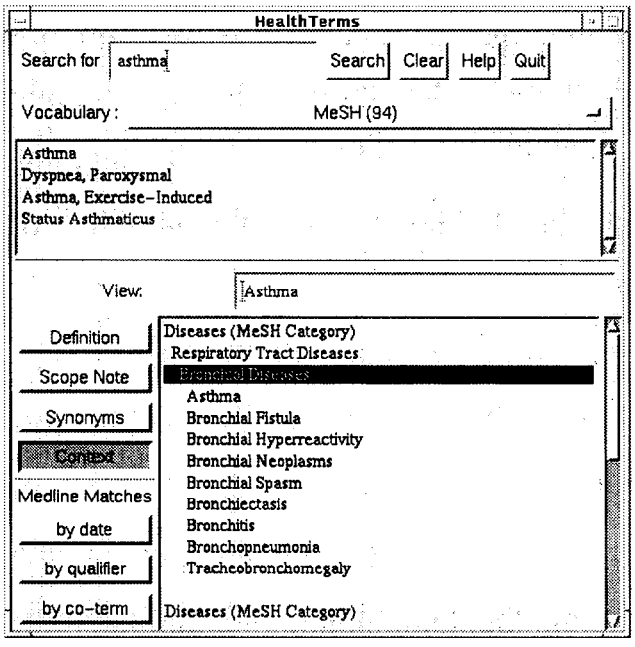

Figure 6 The X-Motif prototype for a Unified Medical Language System (UMLS)-based subject browse list for MEDLINE.

Staffing models have also changed to meet the growing demand for user assistance: 1) e-mail assistance; 2) information/help desks staffed by students and nonprofessional staff; and 3) professional staff available as consultants by appointment. Librarians are making office calls and expanding a departmental liaison program for more personalized service.

Promising Future Work

Willow currently provides a generic, alphabetic browse list for the selection of controlled vocabulary terms. This is a good lowest-common-denominator model, but it is not intended to be a thesaurus. In 1994, we prototyped a browser called HealthTerms[27] using data from the UMLS MetaThesaurus, as illustrated in Figure 6. The prototype's design objectives were: 1) keep it simple; 2) include only the most useful data; 3) use an open architecture; and 4) retain stand-alone usability. HealthTerms is an X-Motif interface to the NLM's Internet-accessible Knowledge Sources Server[28] (KSS). While most data received from the KSS are displayed as received, some programming was required to limit what the user sees and to provide optimal navigation. A usability study was employed in 1995 to test the prototype. The nine clinicians in the study concurred that our prototype filled a need for subject heading discovery and allowed them to create more sophisticated queries. We plan to pursue a Web-based HealthTerms module that will be integrated into the Willow architecture in 1996.

Willow currently provides a simple categorization of available databases. As the number of accessible databases grows, it is clear that users will need assistance in finding the right set of information resources. We will investigate the UMLS ISM and other resource discovery tools for possible integration.

Conclusion

Willow is a general-purpose, extensible search-and-retrieval tool. It uses database drivers to translate user queries and actions into the idiom expected by a remote database host. Drivers are available for a local UW search host as well as the Z39.50 protocol. Willow insulates users from the idiosyncratic query commands and provides a list browser mechanism to assist in query formulation. Multimedia extensions allow users to access data such as sound, images, or tagged SGML text using external viewers.

The Willow development effort demonstrates the complexity required to implement a uniform search interface in a large academic environment. The timeline of such a development project is measured in years, and frequent design modifications are required to meet user expectations in a rapidly changing information technology environment. The ubiquitous and rapidly changing nature of new technologies has changed the fundamentals on which we had based our original vision of a uniform search interface.

That original vision anticipated but could not implement the as-yet-unrealized potential of Z39.50 servers, multimedia browsers, and protocols such as HTTP/HTML and MIME attached mail. Now many users expect an information retrieval tool to provide rich text, images, sound, video, and hyperlinks. The lesson learned from this is that if the tool is to avoid obsolescence, it has to adapt to and assimilate emerging technologies. The vision of a flexible common interface for text searching has been metamorphosed due to Web developments. We now need a flexible common interface that conforms to the emerging Web standards, recognizes different sorts of information objects, and retrieves those objects in a palatable format. It will take continual investment and creativity to sustain Willow as an information retrieval tool as new technologies emerge.

■

The authors recognize the other members of the current UW Willow Development Team: Ellen Jensen and Jill McKinstry. Programmers for the project include: Matthew Freedman (X-Willow and overall Willow architecture); Scott Heyano (VT-100 version and HealthTerms prototype X-UMLS browser); Pete Libbey (Windows version); Jim Fox (UW Libraries Session Manager and Help); Leman Chung and Tom Unger (Browse Lists); Steve Jones (UW License Server); Dan Groves and Leman Chung (BRS customization for the Willow driver); Bill Cattey at MIT (the Z39.50 database driver based on the Stanford University Z39.50 API by Harold Finkbeiner, which incorporates the code from OCLC); and William Jordan (Libraries Catalog browse lists and the UW Z39.50 implementation).

References ■

1. DIALOG. Mountain View, CA: Knight-Ridder Information (http://www.dialog.com).
2. OCLC. Dublin, OH: Online Computer Library Center (http://www.oclc,org).
3. SilverPlatter. Norwood, MA: SilverPlatter Information (http://www.silverplatter,com).
4. OVID. New York: OVID Technologies.
5. RLG. Mountain View, CA: Research Libraries Group (http://www-rlg.stanford.edu).
6. Waldstein R. A pointer page about Z39.50 resources (http://ds.internic.net/z3950/z3950.html).
7. Broering NC, Bagdoyan HE. The impact of IAIMS at Georgetown: strategies and outcomes. Bull Med Libr Assoc. 1992;80:263–75.
8. Wiggins R. The University of Minnesota's Internet Gopher system: a tool for accessing network-based electronic information. Public Access Comput Syst Rev. 1993;4:4–60 (ftp://ftp.lib.ncsu.edu/pub/stacks/pacsr/pr-v4n02-wiggins).
9. Vizine-Goetz D, Godby J, Bendig M. Spectrum: a Web-based tool for describing electronic resources. Comput Network ISDN Syst. 1995;27:985–1001.
10. Lindberg DA, Humphreys BA. The UMLS Knowledge Sources: tools for building better user interfaces. Proc Annu Symp Appl Med Care. 1990:121–5.
11. Miller PL, Frawley SJ, Wright L, Roderer NK, Powsner SM. Lessons learned from a pilot implementation of the UMLS Information Sources Map. JAMIA 1995;2:102–15.
12. Cimino C, Barnett GO, Hassan L, Blewett DR, Piggins JL. Interactive query workstation: standardizing access to computer-based medical resources. Comput Methods Programs Biomed 1991;35:293–9.
13. Lynch CA. The client–server model in information retrieval. In: Dillon M, ed. Interfaces for Informatin Retrieval and Online Systems: a State of the Art. Westport, CT: Greenwood Press, 1991;301–18.
14. Needleman MH. The Z39.50 protocol: an implementor's perspective. Resource Sharing Info Network. 1992;8:89–103.
15. Carson SM, Freivalds DI. Z39.50 and LIAS: Penn State's experience. Info Tech Libr. 1993;12:230–7.
16. Corey JF. A grant for Z39.50. Libr Hi Tech. 1994;12:37–47.
17. Planning for the FirstSearch Z39.50 access option (http://www.oclc.org/oclc/promo/7405z39/7405z39.htm).
18. Ketchell DS, Fuller SS, Freedman MM, Lightfoot EM. Collaborative development of a uniform graphical interface. Proc Annu Symp Appl Med Care. 1992:251–5.
19. Freedman M. Willow: the Washington Information Looker-upper Layered Over Windows. X Resource. 1995;14:13–31.
20. Freedman M. Willow information center (http://www.washington,edu/willow).
21. Berners-Lee T, Cailliau R, Luotonen A, Frystyk-Nielsen H, Secret A. The World-Wide Web. Communications ACM. 1994;37:76–82.
22. Isite. Center for Networked Information Discovery and Retrieval (http://vinca.cnidr.org/software/Isite/Isite.html).
23. http://patents.cnidr.org/pto/bool.html
24. http://www.lycos.com
25. http://www.webcrawler.com
26. O'Connell M. Java: the inside story. SunWorld Online. 1995; Jul (http://www.sun.com/sunworldonline/swol-07-1995/swol-07-java.html).
27. Ketchell DS, Radow DP. Designing and testing a Meta-Thesaurus browser. Washington, DC: Presented at the Medical Library Association Meeing, May 1994.
28. McCray AT, Razi A. The UMLS Knowledge Source Server. MEDINFO 95 (in publication).