

The Internet for Anthropologists

(<http://www.lib.ncsu.edu/staff/morgan/eric-talks-to-anca.html>)

Eric Lease Morgan, NCSU Libraries

This text is a written version of the presentation given to the Association of North Carolina Anthropologists (ANCA) at the North Carolina State University Libraries April 22, 1995. Its purpose is to provide an overview of the Internet (specifically the World Wide Web) and what it can mean for anthropologists.

Client/Server Computing

To truly understand how much of Internet works, you must understand the concept of client/server computing. The client/server model is a form of distributed computing where one program, the client, communicates with another program, the server, for the purpose of exchanging information. The first piece of software is the client. Its responsibility is usually to:

- Handle the end-user interface
- Translate the end-user's requests into the desired protocol
- Send the requests to the server
- Wait for the server's response
- Translate the response into "human readable" results
- Present the results to the end-user

The server's functions include:

- Listening for client queries
- Processing those queries
- Returning the results back to the client

Atypical client/server interaction goes like this:

1. End-user runs client software to create a query
2. Client connects to the server
3. Client sends the query to the server
4. Server analyzes the query
5. Server computes the results of the query
6. Server sends the results to the client
7. Client presents the results to the

end-user
8. Repeat

This client/server interaction is a lot like going to a French restaurant. At the restaurant, you (the end-user) are presented a menu of choices by the waiter (the client). After making your selections, the waiter takes note of your choices, translates them into French, and presents them to the French chef (the server) in the kitchen. After the chef prepares your meal, the waiter returns with your dinner (the results). Hopefully, the waiter returns with the items you selected, but not always; sometimes things get "lost in the translation."

User-interface development is the most obvious advantage in client/server computing. Within this model it is possible to create an interface to data independent of the computing environment hosting the data. Therefore, the user interface of a client/server application can be written on a Macintosh and the server can be written on a mainframe. At the same time, clients could be written for DOS- or Unix-based computers and access the same data from the same mainframe. Since the user interface is now the responsibility of the client, the server has more computing resources to spend on analyzing any queries and disseminating information. Here lies another advantage of client/server computing; it tends to use the strengths of divergent computing platforms to create a more powerful applications. There is no reason why a Macintosh may not be used as a server except its computing and storage capabilities are dwarfed by the mainframe's. The client/server model also provides the opportunity to store information in a central location and disseminate that information regardless of the remote computer.

In short, client/server computing provides a mechanism for disparate computers to cooperating on a single computing task.

Common Internet protocols

Most of the software developments taking place on the Internet have done so only in the past few years. Namely, those being wide area information servers (WAIS), gopher, and the hypertext transfer protocol (HTTP, or more commonly called the World Wide Web or WWW for short). Before these newer protocols were developed, the standard telnet, file transfer protocol (FTP), and electronic mail protocols were tools used most often. The majority of these protocols are based on client/server computing and therefore require at least client and server software to make them useful.

Telnet is a way of logging on to a remote computer and using your computer as if it were attached to the remote computer. When you telnet to a remote computer, your local computer is acting like a "dumb" terminal. It is a lot like using your modem from home to dial into your campus computing centers. This tool was originally used to share computing resources of a central site (like super computer resources) with researchers at remote distances. Nowadays it is mostly used to connect to the electronic card catalogs of libraries.

FTP provides rudimentary file manipulation services on remote computers. This includes the ability to copy files from (get) remote computers as well as copying files to (put) remote computers. FTP also provides the ability to create and manipulate remote directory structures. Thus you can create and delete directories on remote computers.

Wide Area Information Servers (**WAIS**) are an indexing/searching/document-distribution technology. The indexing part of WAIS creates pointers to specific collections of data just as Sociological Abstracts points readers to specific journal articles. Unlike Sociological Abstracts, WAIS also provides the means of retrieving items from the collection using the client/server model. Thus, after a collection of files (text files, software, data files, graphics, et cetera) has been indexed, a client program can be used to search the index. The user's search strategy is then sent to the server who processes the query and returns the results. The client then has the opportunity to select items from the results and retrieve the actual files.

The **gopher** protocol, developed by a computing center at the University of Minnesota was the first Internet service to put a user-friendly front end on the protocols outlined above. Using the client/server model of computing, the gopher protocol allows a server administrator to construct a menu of Internet services including most of the items listed above as well a few others. Furthermore, it allows the administrator to present these services in such a way that the user simply selects items from a menu and the server does the rest. Previous to gopher, to telnet, FTP, read USENET news, or search a WAIS index the user had have client software for each of these services. With the advent of the gopher protocol and a well organized gopher server, these services were reduced to selecting options from a menu.

The **WWW** began in 1989 as the brain-child of Tim Berners-Lee, and first realized in 1991 while working at CERN, a particle physics laboratory in Geneva, Switzerland. The WWW protocol, formally called the hypertext transfer protocol (HTTP) was first intended as a means to share information between members of the high energy physics community. The operative word describing HTTP is "hypertext" as originally described by Vannevar Bush and coined by Theodor H. Nelson. In this system, text is presented to a reader with "links" to other texts intended to provide more explanation of the original text. Scholarly journal articles represent an excellent application of this technology.

For example, scholarly articles usually include multiple footnotes. If hypertext is applied to a journal article, then the reader could select footnotes from the article and be "transported" to the footnote. The footnote, in turn could contain links and the process could go on indefinitely. Within this same system the reader has the opportunity to return to where they originated. Since its inception, the hypertext concept as embodied by HTTP has to include descriptions of college and university departments, collections of Internet resources, newspapers, other items of a non-scholarly nature, and just about anything else you can conceive.

It wasn't until early 1993 when Bob McCool and Marc Andreessen, then of the National Center for Super Computing Applications (NCSA) wrote both HTTP client and server applications that HTTP really started to become more popular. Since the server application (httpd) was available for many flavors of Unix, not just NextStep, the server application was easily put to use by many people. Since the client application (Mosaic for X Windows) supported graphics, as well as the WAIS, gopher, and FTP protocols, the client application was head and shoulders over the original CERN client application in terms of aesthetic appeal as well as functionality. Later, a more functional terminal-based client (Lynx) was developed by Lou Montulli then of the University of Kansas and made HTTP accessible to the lowest common denominator, vt100 based terminals. Lastly, since NCSA later released Macintosh and Windows versions of Mosaic, HTTP became even more popular because an even wider audience now had access to the 'Web. Since then other client and server HTTP applications have been developed, but the real momentum was created by the developers at NCSA.

Client Software

Five examples of WWW client software are described here: MacWeb, Mosaic for Microsoft Windows, Lynx, Mosaic for X Windows, and Netscape. These particular pieces of software are described because they presently represent the best clients for the most common operating systems: Macintosh, Microsoft Windows, terminal-

based VMS or Unix computers, and computers running X Windows.

The real power of these WWW clients (usually referred to as "browsers") is their ability to understand multiple Internet protocols. Specifically, each of the browsers described here understand how to FTP files, act as gopher clients, as well as read and interpret the output of HTTP servers. Additionally, each of these pieces of software understand "forms", an HTML extension allowing the end-user to complete electronic forms similar to gopher+ ASK blocks. While none of these clients directly understand the telnet protocol, each one of these browsers can be configured to load and run your telnet software. Since WWW browsers take URLs as input, and since URLs uniquely describe the location of various Internet resources, then WWW browsers are the tools that really turn the World Wide Web into a "world wide web".

As the name implies, **MacWeb** (<http://galaxy.einet.net/EINet/MacWeb/MacWebHome.html>) is a WWW browser for the Macintosh. Written at Microelectronics and Computer Technology Corporation (MCC), MacWeb is distributed via the Enterprise Integration Network (EINet). MacWeb requires System 7 and at least MacTCP version 2.0.2. Just about anybody using a Macintosh is using System 7. MacTCP is a operating system extension available from Apple Computer enabling your Macintosh to understand the Transport Control Protocol (TCP) necessary for Internet communications. A very important piece of software called "StuffIt Expander", is strongly recommended when using MacWeb as well as MacMosaic. StuffIt Expander is a utility program used to translate and uncompress files usually retrieved via FTP archives.

Besides its speed, elegant and easily customizable interface, the automatic creation of HTML documents from its hotlists, MacWeb indirectly supports the WAIS protocol by launching MCC's WAIS client, MacWAIS.

Mosaic for Microsoft Windows (<http://www.ncsa.uiuc.edu/SDG/Software/WinMosaic/HomePage.html>) is bound to be one of the more popular WWW

browsers since most people have or will have Microsoft Windows-based computers. [11] Mosaic for Microsoft Windows requires a winsock DLL. Like MacTCP, a winsock DLL is software allowing your computer to understand TCP. Common winsock packages include LANWorkplace and Trumpet Winsock. Additionally, Mosaic for Microsoft Windows requires Windows extensions (Win32s) in order to take advantage of 32-bit applications. Since almost all computers running Windows contain '386 microprocessors, and since the Win32 software is available via FTP from NCSA, then you will be able to use Mosaic for Microsoft Windows if you are willing to install the necessary software.

One of the nicest features of Mosaic for Microsoft Windows is its menubar customizability. By editing the mosaic.ini file, you can delete or add menu items to the menu bar. Consequently you can configure Mosaic and have it display commonly used Internet resources for you and your clientele.

Lynx

(<ftp://ftp2.cc.ukans.edu/pub/WWW/lynx/>) is a character-cell WWW browser, meaning it is intended to be used on DOS computers or "dumb" terminals running with the Unix or VMS operating systems.

The flavors of Lynx are ideal clients in two cases. First, they are wonderful when your only Internet connection is located on a remote computer, ie. most dial-in access. Secondly, these clients are excellent when you need to provide a lowest common denominator interface, ie. vt100 terminals.

You won't get pictures with Lynx clients. Nor will you hear sounds. But the Lynx clients do support the "mailto" URL. Mailto URLs are URLs specifying the Simple Mail Transfer Protocol (SMTP) or Internet mail. Thus, when an end-user using a Lynx client selects a mailto URL, then the end-user will be presented with a "form" to complete and the resulting text from the form will be delivered via Internet mail to the person or computer specified in the URL.

Mosaic for X Windows

(<http://www.ncsa.uiuc.edu:80/SDG/Software/XMosaic/>), coupled with NCSA's HTTP

server (httpd), really gave the WWW the momentum and visibility it has today. This browser supports copy and paste from the display. Direct WAIS support, and therefore URLs like <wais://wais.lib.ncsu.edu/alawon?nren> are valid. At the present time, just about the only thing it doesn't support is the mailto URL. One of the nicest features of Mosaic for X Windows is its ability to directly deliver a displayed document to somebody via email. The downside of Mosaic for X Windows is its requirement of a relatively high powered computer. While a Macintosh equipped with MacX or a Windows-based computer with HummingBird can run X Windows terminal sessions, Mosaic for X Windows really requires direct access to a Unix or VMS machine running the X Windows software.

Netscape (<http://www.mcom.com/>) is the most full-featured WWW browser available to date. It smoothly integrates the FTP, gopher, and usenet news protocols into one application. It has implemented many proposed extensions to the hypertext markup language (HTML). This means it understands markup tags like `<center></center>` as well as table-creation tags. Unlike some of the other browsers mentioned here, Netscape opens up multiple connections to remote servers at a single time. This makes the Netscape browser seem faster than other browsers. Netscape also implements a security standard known as the secure sockets layer (SSL). By using this standard, the use of Netscape, in conjunction with an SSL compliant server application, and send and receive confidential information over the Internet with no possible breach in security.

Uniform Resource Locators

Uniform Resource Locators (URL) are a fundamental part of the WWW. They are used to concisely describe and identify the protocol and location of Internet resources. Presently, the most definitive document describing URLs is called "WWW Names and Addresses, URIs, URLs, URNs".

In general, a URL has the following form:

`protocol://host/path/file`

"Protocol" denotes the type of Internet resource. The most common are: gopher, wais, ftp, telnet, http (WWW), file, and mailto (electronic mail). "Host" denotes the name or Internet Protocol number of the remote computer. Examples include: www.lib.ncsu.edu or 152.1.39.42. "Path" is a directory or subdirectories on the remote computer. "File" is the name of the file you want to access.

Using variations of this general form, you can use URLs and your WWW browsers to access just about any Internet resource. Simple examples of URL's include:

1. <ftp://ftp.lib.ncsu.edu/pub/stacks/alawon/alawon-v1n04>
2. <http://www.lib.ncsu.edu/stacks/alawon-index.html>

The first example illustrates how your WWW browser can be used to copy a file from a remote FTP server. It translates to:

1. FTP to [ftp.lib.ncsu.edu](ftp://ftp.lib.ncsu.edu)
2. Log on as anonymous
3. Change directories to [/pub/stacks/alawon/](ftp://pub/stacks/alawon/)
4. Get the file [alawon-v1n04](ftp://alawon-v1n04)

Since WWW browsers understand and implement the File Transfer Protocol (FTP), you do not have to remember all the commands necessary to do FTP. All you have to remember is how to create a FTP-style URL.

Similarly, the second example opens up a HTTP connection to www.lib.ncsu.edu, changes directories to stacks, and retrieves the file alawon-index.html.

URLs are more complicated than the general form illustrated above; URLs can also provide the means to present the logon name for telnet connections, a communications port, an index/search query, and/or an HTML anchor.

For example, a complete URL identifying a telnet session would look something like:

<telnet://library@library.ncsu.edu:23/>

where "library" denotes the logon name and 23 denotes the communications port. (Port 23 is the standard telnet communications port.) Thus you can use your WWW browser to initiate a telnet session. Specifically, the above example opens up a telnet connection to "library.ncsu.edu" and depending on your browser, you may be reminded to log on as "library". This URL does not use the "path" nor "file" parameters because they are meaningless for telnet sessions.

On the other hand, if you wanted to manually query the Geographic Name Server, then the URL would be:

telnet://martini.eecs.umich.edu:3000/

Since the Geographic Name Server requires no password, no password is specified, but since the Geographic Name Server "listens" on port 3000, a non-standard port number must be specified.

WAIS searches can be specified using URLs. Unfortunately, at the present time, only Mosaic for X Windows directly implements the WAIS protocol. WAIS URLs have the following form:

wais://host:port/database?query

where "port" is assumed to be 210, the standard WAIS/Z39.50 port, "database" is the source file to search, "?" delimits the database from the query, and "query" is the your search strategy. For example:

wais://vega.lib.ncsu.edu/alawon.src?nren

Gopher servers and files can be specified with URLs as well. Since gopher resource specifications require "Type" identifiers, and since paths to gopher resources often include spaces, gopher URLs usually deviate from the norm. For example, here is a URL describing the location of a subdirectory in Gopher at the NCSU Libraries:

gopher://gopher.lib.ncsu.edu/11/library/

Notice the pair of 1's after the Internet name of the computer. These 1's specify the resource as a directory. On the other hand, the following URL specifies a specific text file within that directory:

gopher://gopher.lib.ncsu.edu/00/library/abou

The 0's denote text files.

Anasty wrench gets thrown into the business when the path and/or file names of the Internet resources contain special characters like spaces or colons. In these cases, escape codes must be used to denote the special characters. For example,

gopher://gopher.lib.ncsu.edu/0ftp%3amrcnext.cso.uiuc.edu%40/pub/etext/etext91/aesop11.txt

This long URL first specifies a gopher server (gopher.lib.ncsu.edu) to FTP a file from mrcnext.cso.uiuc.edu, and get aesop11.txt. Notice the "%3a" and "%40" in the URL. They are used to denote a colon and at-sign (@), respectfully. Furthermore, notice the zero proceeding the "ftp". Here again, this is used to identify the remote file as a text file. In short, gopher URLs are particularly difficult to decipher. If you must communicate the location of a gopher resource via a URL,

then it is much safer to first visit the resource in question and copy resulting URL from your client's display.

In summary, URLs unambiguously describe the location of Internet resources. Using URLs as a standard, Internet-client programs like WWW browsers can interpret URLs and retrieve the desired information. URLs describe the protocols and locations of Internet resources independently of Internet-capable client software.

Anthropology-related Internet resources

There are quite a number of anthropology-related Internet resources available. They run the gamut of FTP sites to WWW servers. Some of the more extensive collections include:

- Anthropology - <http://www.lib.ncsu.edu/disciplines/anthropology.html>
- Anthropology and Culture - [gopher://riceinfo.rice.edu:70/11/Subject/Ant](http://riceinfo.rice.edu:70/11/Subject/Ant)
- Anthropology on the Internet - <http://www.umanitoba.ca:80/anthropology/aaa-revue.html>
- Institute of Social and Cultural Anthropology - <http://www.rsl.ox.ac.uk/isca/>
- World Wide Web Virtual Library: Anthropology - <http://www.usc.edu/dept/v-lib/anthropology.html>

Sometimes these pages may not contain all the sorts of information you need and therefore you may have to go searching for it. Try these resources as examples:

- Lycos - <http://lycos.cs.cmu.edu/lycos-form.html>
- Searching the Web - http://www.yahoo.com/Reference/Searching_the_Web/
- WebCrawler - <http://webcrawler.cs.washington.edu/WebCrawler/WebQuery.html>
- Yahoo - <http://www.yahoo.com/>

Finally, there was recently published an article entitled "Internet resources for anthropology" by Anita Cohen-Williams and Julia A. Henderson in *College and Research Libraries News* volume 56, number 2, February 1995 pages 87-90, 113. This article lists many Internet resources for anthropologists including more than a few discussion groups.