Guide to Network Resource Tools

EARN Association

May 3, 1993

Introduction

As the worldwide academic computer network grows and expands far beyond its previous confines, so the resources and services available on the network evolve and multiply at a dizzying rate. The typical user is hardpressed to keep up with this explosive growth. Fortunately, a number of tools are available to facilitate the task of locating and retrieving network resources, so that users anywhere can utilize texts, data, software and information for public access. Facilities to explore public domain software repositories, to consult mailing list archives and databases, to retrieve directory information and to participate in global group discussions are now available to all.

The key to exploiting these resources is a **server**, special software on a computer somewhere in the network which accepts requests (or queries or commands) and sends a response automatically. The requestor does not have to be working on the same computer (or even in the same part of the world) in order to use the server. Many servers accept requests via electronic mail, so that often the requestor need not even be on the same computer network as the server. In many cases, servers are interconnected so that once you have established contact with one server, you can easily communicate with other servers as well.

Today, many users have powerful computers on the desktop, with advanced graphical, audio and storage capabilities, which are connected to the network. This fact has given rise to what is known as the **client-server** model. Users can have special software on their local computer called a **client** which can utilize the capabilities of that computer and can also communicate with a server on the network. These clients provide an easy-to-use, intuitive user interface, allow use of pointing devices such as a mouse, and exploit other local features. The client sends the user's requests to a server using a standardised format (called a *protocol*) and the server sends its response in a condensed format which the client displays to the user in a more readable way.

The tools described in this guide have been divided into five functional areas. Actually, several of the tools have several different functions, and have been classified according to their main purpose. The first section, **Exploring the network**, covers two relatively new services, Gopher and World-Wide Web, which use the client-server model to provide a means of moving through a wide range of network sources and resources in a uniform and intuitive way. Two tools for searching in a wide range of different databases located throughout the network, WAIS and ASTRA, are documented in the section **Searching databases**. The problem of knowing where to find people and things in the network is addressed in the third section, **Finding network resources**, which deals with Archie, WHOIS and NETSERV. While just about all of these network tools can be used to get files of one sort or another, there are a few servers available for getting files easily and efficiently from various repositories in the network. Two of these servers, Trickle and BITFTP, are covered in section four on **Getting files**. The final section deals with what is perhaps the most popular of all the network resources, discussion groups on every imaginable topic. The two tools discussed in this section are LISTSERV and Netnews.

Acknowledgements

We would like to express our gratitude to the authors and developers of these tools for making their work available to the networking community. We have consulted their documentation, and the documentation of others, in preparing this guide, and for this we are also grateful. We have used their descriptions and explanations when suitable, but any errors or inaccuracies are the sole responsibility of the EARN staff. Consult the sources listed at the end of each section for specific queries and original detailed information.

Notice

This document is available in electronic format from: **LISTSERV@EARNCC.BITNET**

Send the command: **GET filename** where the filename is either:

NETTOOLS PS	(Postscript)
NETTOOLS MEMO	(plain text)

Contents

Part 1
EXPLORING THE NETWORK

GOPHER

What is Gopher

The **Internet Gopher**, or simply **Gopher**, is a distributed document delivery service. It allows users to explore, search and retrieve information residing on different locations in a seamless fashion.

When browsing it, the information appears to the user as a series of nested menus. This kind of menu structure resembles the organization of a directory with many subdirectories and files. The subdirectories and the files may be located either on the local server site or on remote sites served by other Gopher servers. From the user point of view, all information items presented on the menus appear to come from the same place.

The information can be a text or binary file, directory information (loosely called *phone book*), image or sound. In addition, Gopher offers gateways to other information systems (World-Wide Web, WAIS, archie, WHOIS) and network services (Telnet, FTP). Gopher is often a more convenient way to navigate in a FTP directory and to download files.

A Gopher server holds the information and handles the users' queries. In addition, links to other Gopher servers create a network wide cooperation to form the global Gopher web (*Gopher-space*).

Who can use Gopher

Gopher uses the client-server model to provide access to the Gopher web. You must be on the international TCP/IP network (the Internet) in order to use a client on your computer to access Gopher.

How to get to Gopher

Users explore the Gopher menus using various local clients or accessing a remote client via an interactive Telnet session.

Local clients

Public domain clients for accessing a Gopher server are available for: Macintosh, MS/DOS, OS/2, VM/CMS, VMS, NeXT, Unix, X-Window.

The clients are available for anonymous FTP from many FTP sites (eg, boombox.micro.umn.edu in the directory */pub/gopher*).

Remote clients

Some sites allow public access via Telnet to a client. To access such a remote client, telnet to one of these sites:

info.anu.edu.au	Australia (login: info)
tolten.puc.cl	Columbia
ecnet.ec	Ecuador
gopher.chalmers.se	Sweden
consultant.micro.umn.edu	USA
gopher.uiuc.edu	USA
panda.uiowa.edu	USA (login: panda)

At the login prompt type **gopher** (unless specified otherwise) and the top-level Gopher menu for that site will be displayed.

Users are requested to use the site closest to them.

Using Gopher

The implementations of the Gopher clients on various platforms are slightly different to take advantage of the platforms' capabilities (mouse, graphic functions, X-Window server) and to offer the popular look and feel. Even with different implementations, the same set of functions and commands is available.

When issuing the **gopher** command, you will be connected automatically to the default Gopher server specified at the installation. The format of the command is:

gopher <hostname>

where *hostname* is an optional alternative Gopher server you want to talk to.

When connected to a Gopher server, it is still possible to access another server by exploring the *Other Gopher servers in the rest of the world* branch. To locate them more easily, the Gopher servers are distributed in geographical regions:

- Africa
- Europe
- Middle East
- North America
- Pacific
- South America

and then by countries.

Access to a Gopher server is identical whether using a local or a remote client: a simple menu-driven interface which doesn't require any special training or knowledge from the user.

Here is a sample menu:

```
                  Internet Gopher Information Client v1.1

                          Information About Gopher

    1.  About Gopher.
    2.  Search Gopher News <?>
    3.  Gopher News Archive/
    4.  comp.infosystems.gopher (USENET newsgroup)/
    5.  Gopher Software Distribution/
    6.  Gopher Protocol Information/
    7.  University of Minnesota Gopher software licensing policy.
    8.  Frequently Asked Questions about Gopher.
    9.  gopher93/
    10. Gopher| example server/
    11. How to get your information into Gopher.
--> 12. New Stuff in Gopher.
    13. Reporting Problems or Feedback.
    14. big Ann Arbor gopher conference picture.gif <Picture>


Press ? for Help, q to Quit, u to go up a menu          Page: 1/1
```

In the example above, any item can be selected by typing its line number or by moving the cursor (-->) next to it.

An item could be:

- a subdirectory
- a text File
- a binary File
- a sound file
- an image file
- a phone book (directory information)
- an index-search
- a Telnet session

Items are displayed with an identifying symbol next to them. In the example above, "<?>" means a full text index search, "/" means a subdirectory, "<Picture>" means an image file and no symbol means a text file.

Some Gopher clients are not able to handle certain file types (eg, sound files). Some clients display only files of types they can handle or files they suppose you are interested in. Others display all types of files.

Most Gopher clients allow you to create, view and select *bookmarks*. A bookmark keeps track of the exact location of a Gopher item, regardless of where it resides. It is useful when you often need to reach a file or a service located far from the top-level directory. A collection of bookmarks is like a customized Gopher menu.

Some capabilities of a local Gopher client are bound to the capabilities of your own computer. In fact, for sound files, image files and Telnet sessions, the Gopher client looks for the appropriate software on your computer and passes control to it to perform the requested task. When the task is completed, control is returned to the Gopher client.

At any time, it is possible to terminate the session (quit command), to cancel the current processing or to get the on-line help (help command).

An item is processed according to its type:

a subdirectory its contents are displayed. To go up one level, use the up command.

a text file the file is displayed. Then you can browse it, search for a particular string, print it on a local printer or copy (save) it onto your local disk space in a user-specified file (the last 2 functions may not be available to you).

a binary file the remote file is simply copied onto your local disk space in a user-specified file. Binary files are binhexed Macintosh files, archives (.zip, .tar,...), compressed files, programs, etc.

a sound file the remote file is played through your local audio device if it exists, as well as the appropriate utility. Only one sound file can be active at a time; you will be warned if you try to play a sound before a previous one is done.

an image file the remote file is displayed on your computer screen if an image viewer exists on your computer.

a phone book you are prompted for a search string to look up people information through the selected phone book. Since different institutions have different directory services, the queries are not performed in the same fashion.

an index-search you are prompted for a search string which may be one or more words, plus the special operators **and, or**, and **not**. The search is case-insensitive. Usually, an index is created to help users locate the information in a set of documents quickly. Eg:

```
terminal and setting or tset
```

will find all documents which contain both the words *terminal* and *setting*, or the word *tset*. **or** is non-exclusive so the documents may contain all of the words.

The result of the index search looks like any Gopher menu, but each menu item is a file that contains the specified search string.

a Telnet session Telnet sessions are normally text-based information services, for example, access to library catalogs.

VERONICA

Veronica was designed as a solution to the problem of resource discovery in the rapidly-expanding Gopher web, providing a keyword search of more than 500 Gopher menus. Veronica helps you find Gopher-based information without doing a menu-by-menu, site-by-site search. It is to the Gopher information space, what archie is to the FTP archives

Veronica is accessible from most top-level Gopher menus or from the *Other Gopher servers...* branch. There is no need for opening another connection or another application.

There are two search methods: the simple Boolean search method and the partial Boolean search method. When you choose either search method, you will be prompted for a search string.

The searches are carried out on an index of Gopher menu titles. They are NOT full-text searches of data at Gopher sites. Veronica queries are case-insensitive.

Simple Boolean search method:

The search string may contain keywords optionally separated by **and** and **or**. **and** is assumed between 2 words. **or** takes precedence over **and**. Eg:

```
eudora
```

will give you a list of menu titles that contain *eudora*:

```
Electronic Mail: Eudora on Macintosh, Micro-08
Modem Setting Eudora Slip.
A UNIX-based Eudora reader for those that ...
Eudora:  Popmail for the Macintosh.
Eudora.
etc.
```

while

```
eudora and macintosh
```

will give you a list of menu titles that contain both *eudora* and *macintosh*:

```
Eudora:  Popmail for the Macintosh.
v4.1 EUDORA: E-MAIL FOR THE MACINTOSH.
Micro News:  Eudora - A Mailer for the Macintosh.
Eudora: Electronic Mail on Your Macintosh.
ACS News - Eudora Mail Reader for Macintosh.
etc.
```

Partial Boolean search method:

BOOLEANS
The search string contain keywords optionally separated **and**, **or** and **not**. The evaluation is done from left to right. **not** clauses are evaluated after all the other clauses are evaluated. Eg:

```
red and blue or yellow but not green and orange or black
but not white
```

will be interpreted as:

```
     (((((red and blue) or yellow) and orange) or black) not
     green) not white)
```

PARTIALWORD
'*' is the wildcard character, which can replace any other character. It causes a partial search on the substring it follows. Eg:

```
     desk*
```

will give you a list of menu titles, including:

```
     The Help Desk.
     Keene State College Press Release COMPUTER ON EVERY DESK.
     DESKQview/X... An alternative to Windows???.
     Ethernet at Your Desktop/
```

LITERAL
A string surrounded with a pair of quotes (') or double-quotes (") causes a exact match search (case-insensitive). **and**, **not**, and '*' may occur in the literal string, they have no special meaning. The first part of a literal string must be a word, rather than a delimiter symbol, to get a successful search. Eg:

```
     "desk"
```

will give you a list of menu titles, including:

```
     Mac on Your Desk.
     Information Desk.
     Available at the Help Desk ....
```

The result of both search methods is a Gopher menu composed of items (with an identifying symbol) whose titles match the search string. Like any Gopher items, you can process them: open subdirectories, display text files, etc.

Learning more about Gopher

The Internet Gopher is developed by the Computer and Information Services Department of the University of Minnesota. Bug reports, comments, suggestions, etc. should be mailed to the Gopher development team at: **gopher@boombox.micro.umn.edu.**

Mailing list: **gopher-news@boombox.micro.umn.edu**
To subscribe send a mail to: **gopher-news-request@boombox.micro.umn.edu**

Usenet newsgroup: **comp.infosystems.gopher**

A comprehensive description of veronica search methods is available from the veronica menus.

Veronica is being developed by Steve Foster and Fred Barrie at the University of Nevada. Bug reports, comments, suggestions, etc. should be addressed to: **gophadm@futique.scs.unr.edu**

WORLD-WIDE WEB

What is World-Wide Web

World-Wide Web (also called **WWW** or **W3**) is an information system based on hypertext, which offers a means of moving from document to document (usually called *to navigate*) within a network of information.

Hypertext documents are linked to each other through a selected set of words. For example, when a new word, or a new concept, is introduced in a text, hypertext makes it possible to point to another document which gives more details about it. The reader can open the second document by selecting the unknown word or concept and the relevant section is displayed. The second document may also contain links to further details. The reader need not know where the referenced document is, and there is no need to type a command to display it, or to browse it to find the right paragraph. Cross-references may be defined in the same document. A collection of documents is a database.

If you were reading this document on a hypertext system, instead of this all too short explanation about hypertext, you would have a selectable pointer to a complete hypertext information web with examples and more pointers to other definitions.

For instance, in the first document you might read:

```
The WorldWideWeb (W3) is a wide-area "hypermedia" information
retrieval initiative aiming to give universal access to a large
universe of documents.
```

Selecting *hypermedia* will display the following explanation for you:

```
                  WHAT IS HYPERTEXT
Hypertext is text without the constraint of linearity.

Hypertext is text which contains "links" to other texts. The
term was coined by Ted "Nelson" around 1965.

Hypermedia is a term used for hypertext which is not
constrained to be text.
```

Then you can learn more about *links* and *Nelson*.

Also, special documents (indexes) in the WWW information space can be search for given keyword(s). The result is a document which contains links to the documents found.

World-Wide Web uses hypertext over the network: the linked documents may be located at various sites. WWW can handle different text formats and various information organizations. WWW also provides access to many of the other tools described in this guide.

Who can use World-Wide Web

WWW uses the client-server model to provide access to the information universe. You must be on the international TCP/IP network (the Internet) in order to use a client on your computer to access WWW.

How to get to World-Wide Web

Users access the World-Wide Web facilities via a client called a *browser*. This interface provides transparent access to the WWW servers. If a local WWW client is not available on your computer, you may use a client at a remote site. Thus, an easy way to start with WWW is to access a remote client.

Local clients

Usage of a local client is encouraged since it provides better performance and better response time than a remote client.

Public domain clients for accessing WWW servers are available for: Macintosh, MS/DOS, VMS, VM/CMS, MVS, NeXT, Unix, X-Window.

The clients are available for anonymous FTP from info.cern.ch in the directory *pub/www*. All these platforms support a simple line mode browser. In addition, graphical clients are available for: Macintosh, Windows, X-Window, NeXT and Unix.

Remote clients

To access a remote WWW client, telnet to one of the server sites. At the *login:*prompt enter **www**, no password is needed. There are many WWW servers already available throughout the network, and the number is growing, so it is not possible to include a comprehensive list here. A server may provide databases on specific topics. The following servers act as demonstration sites:

info.funet.fi	Finland
vms.huji.ac.il	Israel
info.cern.ch	Switzerland
eies2.njit.edu	USA

CERN is the entry point to find information about WWW itself and to have an overview of the Web with a catalogue of the databases sorted by subject.

Using World-Wide Web

When using a graphical interface, you access the WWW functions by pressing mouse buttons. In particular, references are highlighted or underlined words. To follow a link, double-click on the associated reference.

The line mode browser is a more simple user-interface: references are numbers in square brackets next to words. Type the number and hit the RETURN key to follow a reference. For example, here is the beginning of the *Subject Catalogue* you get on the CERN server:

```
           The World-Wide Web Virtual Library: Subject Catalogue
                     INFORMATION BY SUBJECT

See also arrangements by organization[1] or by  service type[2].
Mail www-request@info.cern.ch if you know of online information
not in these lists....

Aeronautics     Mailing list archive index"3" .

Astronomy and Astrophysics
                Abstract Indexes"4" (unavailable).   Astrophysics
                work at FNAL"5" .

Bio Sciences[6] separate list .

Computing[7]    Seperate list.

Geography       CIA World Fact Book"8" , India: Miscellaneous
                information"9", Thai-Yunnan: Davis collection"10",

Law             US Copyright law"11" .

Libraries[12]   Lists of online catalogues etc.

Literature & Art[13]  separate list.

Mathematics     CIRM library"14" (french). The  International
                Journal  of   Analytical   and Experimental Modal
1-30, Back, <RETURN> for more, Quit, or Help:
```

The following commands are available within WWW. Some are disabled when not applicable (eg, FIND is enabled only when the current document is an index). Angle brackets (<>) indicate an optional parameter.

Help gives a list of available commands depending on the context, and the hypertext address of the current document.

Manual displays the on-line manual.

Quit exits www.

Up, Down scrolls up or down one page in the current document.

Top, Bottom goes to the top or the bottom of the current document.

Back goes back to the document you were reading before.

Home goes back to the first document you were reading.

Next, Previous
goes to the next or previous document in the list of pointers from the document that led to the current one.

List gives a numbered list of the links from the current document. To follow a link, type in the number.

Recall <number>
if number is omitted, gives a numbered list of the documents you have visited.

To display one specific document, re-issue the command with number.

<Find> keyword
queries the current index with the supplied keyword(s). A list of matching entries is displayed with possibly links to further details. *Find* can be omitted if the first keyword does not conflict with another WWW command. Multiple keywords are separated by blanks.

Go address goes to the document represented by the given hypertext address, which is interpreted relatively to the current document.

Extra command available on Unix versions only:

Print prints the current document, without the numbered document references. The default print command is *lpr*, but it may be defined in your local working environment by the variable WWW_PRINT_COMMAND.

To access WWW with the line mode browser, type: **www**. The default first document will appear on your screen. From this point, you should be able to navigate through the WWW universe by reading the text and following the instructions at the bottom of the screen. If you want to start with a first document other than the default, or if you want to change some other aspect of the usual interaction, there are a number of command line parameters and options available. The full format of the WWW command to invoke the line mode browser is:

www	<options> <docaddress <keyword>>

where:

docaddress is the hypertext address of the document at which you want to start browsing.

keyword queries the index specified by docaddress with the supplied keyword(s). A list of matching entries is displayed. Multiple keywords are separated by blanks.

Options are:

-n non-interactive mode. The document is formatted and displayed to the screen. Pages are delimited with form feed (FF) characters.

-listrefs adds a list of the addresses of all documents references to the end. Non-interactive mode only.

-pn sets the page length to n lines. Without a number, makes the page length infinite. Default is 24.

-wn sets the page width to n columns. The default is 78, 79 or 80 depending on the system.

-na hides references in the text. Useful, when printing out the document.

-version displays the version number of the software.

The options listed above should be available in all clients. They may be abbreviated. Case is not significant. Special characteristics of the line mode browser interface are:

number type in a number given in [] and hit the RETURN key to follow the link associated to the reference.

RETURN hit the RETURN key to display the next page of the current document (without a reference number).

Learning more about World-Wide Web

World-Wide Web is being developed at CERN (European Particle Physics Laboratory) by the World-Wide Web team leaded by Tim Berners-Lee. Bug reports, comments, suggestions, etc. should be mailed to: **www-bug@info.cern.ch**

On-line documentation is available from info.cern.ch, for anonymous FTP or using the remote WWW client.

Mailing lists: **www-talk@info.cern.ch**
To subscribe to this list send a mail to **www-talk-request@info.cern.ch**

Usenet newsgroup: **comp.infosystems.www**

Part 2
SEARCHING DATABASES

WAIS

What is WAIS

WAIS, Wide Area Information Server, is a distributed information retrieval system. It helps users search databases over networks using an easy-to-use interface. The databases (called sources) are mostly collections of text-based documents, but they may also contain sound, pictures or video as well. More than 400 databases on topics ranging from *Agriculture* to *Social Science* can be searched with WAIS.

The databases may be organized in different ways, using various database systems, but the user isn't required to learn the query languages of the different databases. WAIS uses natural language queries to find relevant documents. The result of the query is a set of documents which contain the words of the query: no semantic information is extracted from the query.

Who can use WAIS

WAIS uses the client-server model to provide access to databases. You must be on the international TCP/IP network (the Internet) in order to use WAIS.

How to get to WAIS

There are many WAIS servers throughout the network. A directory-of-servers database is available at several sites. You can address a query to it.

quake.think.com acts as a demonstration site, you can telnet to it. At the *login:* prompt enter **wais**, no password is needed.

A complete listing of the advertised databases is available via anonymous FTP from the University of North Carolina (SunSITE.unc.edu) in the directory */pub/wais* as file *wais-sources.tar.Z.*

Public domain clients for accessing WAIS are available for: Macintosh, MS/DOS, Windows 3.x, VMS, NeXT, Unix, X-Window.

These clients are available for anonymous FTP from the University of North Carolina (SunSITE.unc.edu) in the directory */pub/wais*.

The client interface differs slightly on different platforms. However, the queries are performed in the same way, whatever the interface.

- Step 1: The user selects a set of databases to be searched from among the available databases.
- Step 2: The user formulates a query by giving keywords to be searched for.
- Step 3: When the query is run, WAIS asks for information from each selected database.
- Step 4: Headlines of documents satisfying the query are displayed. The selected documents contain the requested words and phrases. Selected documents are ranked according to the number of matches.
- Step 5: To retrieve a document, the user simply selects it from the resulting list.
- Step 6: If the response is incomplete, the user can state the question differently or feed back to the system any one or more of the selected documents he finds relevant.
- Step 7: When the search is run again, the results will be updated to include documents which are *similar* to the ones selected, meaning documents which share a large number of common words.

Learning more about WAIS

Bug reports, comments, suggestions, etc., should be addressed to:
George Brett **George.Brett@cnidr.org**

Mailing list: **wais-discussion@think.com**
To subscribe send a mail to **wais-discussion-request@think.com**

Usenet newsgroup: **comp.infosystems.wais**

ASTRA

What is ASTRA

The **ASTRA** service allows users to retrieve documents from databases known by ASTRA throughout the network. Users can send their queries to the ASTRA server which in turn forwards the query to the related database servers. This provides an easy-to-use uniform access method to a large number of databases.

ASTRA provides the same user interface for all databases it can access, even if the database servers have different access languages, such as STAIRS, ISIS or SQL.

Each database defined in ASTRA has an abstract which holds information about the database: title, name of the maintainers, a brief description of the database, the main topics of the database and its language. Users are advised to look at the abstracts before sending requests, to avoid sending requests to the wrong databases.

Some databases actually combine several different databases that deal with the same topics. When a user sends a request for such a database, the request is forwarded to all related databases.

Who can use ASTRA

Anyone who can send electronic mail to EARN/Bitnet can access ASTRA. Interactive user interfaces (clients) to ASTRA are available for VM and VMS systems on the EARN/Bitnet network. For all other users, there is a batch language that permits batch queries using e-mail.

How to get to ASTRA

Currently there are five ASTRA servers installed at the following addresses:

ASTRADB@ICNUCEVM.BITNET
ASTRASQL@ICNUCEVM.BITNET
ASTRADB@IFIBDP.BITNET
ASTRADB@IFIIDG.BITNET
ASTRADB@IRMKANT.BITNET

There are clients available for IBM VM and DEC VMS systems running on the NJE network. To order either client, send the following command to ASTRADB@ICNUCEVM.BITNET:

```
SUBSCRIBE SYS=sysname Firstname Lastname
```

where *sysname* is either VM or VAX, depending on your system, and *Firstname Lastname* is your real name (not your e-mail address). The ASTRADB server will send you a copy of the executable files and documentation files necessary to run the client.

Communication between the user interface and the server can be performed either by interactive messages, or via sendfile. In sendfile mode, the query entered by the user is placed in a file and delivered to the server.

Using ASTRA

The commands accepted by the ASTRADB server can be divided into two sections: general commands, such as HELP, and the commands that are part of the ASTRA Batch Query Language.

Parameters in angle brackets (<>) are optional.

General Commands:

GET	fn ft <fm> <password>

where:

fn is the CMS filename of the file

ft is the CMS filetype of the file

fm is the CMS filemode of the file

password a password is required to access certain files.

This command allows the user to order any file from an ASTRA database server. SENDME is a synonym for GET command.

NEWS

The NEWS command delivers a news bulletin from the ASTRA service. A separate file called ASTRA LOGNEWS with a complete log of old news is also available by sending the command: GET ASTRA LOGNEWS to the server.

HELP	<fn>

where:

fn is the optional filename of the helpfile to fetch.

This command provides the requested helpfile. If *fn* is omitted, the ASTRA INFO file is delivered. If a parameter is given, the helpfile with that filename is sent. Helpfiles on several databases are available.

Batch Query Language Commands:

Although the full-screen user interfaces to ASTRA are easier to use, the server also accepts batch-mode commands, which can be sent to the server by e-mail. The following batch-mode commands may be specified. Note that only the first character of each command is required.

StartRequest:	Username: username Host: host

where:

username is the userid to which the results should be sent.

host is the host (or node) name to which the results should be sent.

This command starts the transaction, and specifies the address *username@host* to which the command results should be sent. For example, if the command comes from toot@hubble.circe.fr, the command should be:

```
S: U: toot H: hubble.circe.fr
```

Dbname:	dbname TypeResult: resulttype

where:

dbname is the name of the database

resulttype is the type of result desired.

This command selects a database name to use, and the type of results desired. Here the *dbname* can be any database name valid to ASTRA service. For an up-to-date listing of the available databases, send the command

```
GET ASTRA HELPDBTB
```

to ASTRADB@ICNUCEVM.BITNET.

resulttype indicates how the server should report the results. The choices are NUMERIC, DOCUMENT or SOFTWARE. NUMERIC indicates that only the number of hits should be returned, DOCUMENT specifies that the user should receive all documents matching the criteria, and finally SOFTWARE indicates that all related reports, software and pictures should also be sent to the user. For example, to search the STAR database and get the hit count, enter:

```
D: STAR T: N
```

= querytext

This command starts a keyword search on the selected database. The **querytext** format can be either in STAIRS/CMS query language, or in the query language of the host (SQL stored query, or ISIS query language). The format to be used is specified with the Printformat option. Consult the ASTRA help file for a complete list of search keywords.

Printformat:	format

where:

format is the print format name defined for this database.

This command tells the server which report format to use to display the results of a query. A default printformat called *D* is defined for each database, and specifies that the whole document is to be displayed. For examples and details, refer to the meta-database GEOL. Some other keywords that can be specified for the printformat are as follows:

```
H=Y/N: Heading Required Yes/No
F=T/C: Tabular or Comma-Delimited format.
```

EndRequest:

Use the EndRequest command to signal the end of a job. The Dbname, TypeResult, =, and PrintFormat commands may be repeated as many times as necessary within a batch job.

Examples

If you work on a VAX/VMS system in EARN/Bitnet and your name is Hilary Clinton, to subscribe to the ASTRA service and get the files for the interactive interface, use the command:

```
$ SEND ASTRADB@ICNUCEVM "SUBSCRIBE SYS=VAX Hilary Clinton"
```

To get help on ASTRA's meta-database, send the command:

```
HELP META
```

to: ASTRADB@ICNUCEVM.BITNET

Here is a sample batch file:

```
startrequest: username: TURGUT host: FRORS12.CIRCE.FR
dbname: star typeresult: numeric
= campus and network
printformat:
dbname: soft typeresult: document
= campus and network
printformat: all
endrequest:
```

Learning more about ASTRA

An extensive help file is available by sending the command HELP to ASTRADB@ICNUCEVM.BITNET. The following three files are also available at the same address using the GET command:

META DOCUMENT	the list and description of the available databases
ASTRAIMM DOCUMENT	the installation and maintenance guide
ASTRAFS DOCUMENT	the system specification manual

There are also LISTSERV lists for ASTRA users:

ASTRA-UG Used for the distribution of a newsletter about new databases or new versions of the current databases.

ASTRA-OG Used to report problems about the server. Users are encouraged to submit their questions or problems to this list.

To subscribe, send the command:

```
SUB listname Your Name
```

where *listname* can be either ASTRA-UG or ASTRA-OG to:

LISTSERV@ICNUCEVM.BITNET

An ASTRA newsletter is also available by sending the command:

```
NEWS
```

to ASTRADB@ICNUCEVM.BITNET.

The developers of the ASTRA service may be contacted at:
ASTRA@ICNUCEVM.CNUCE.CNR.IT

Part 3
FINDING NETWORK RESOURCES

ARCHIE

What is ARCHIE

Archie is an information system. It offers an electronic directory service for locating information in the Internet network.

The best known use of archie is for scanning a database of the contents of more than 1000 anonymous FTP sites around the world. Currently, this database contains more than 2,100,000 file names from anonymous FTP sites. This database is known as the archie database.

The files made available at anonymous FTP sites are software packages for various systems (Windows, MSDOS, Macintoshes, Unix, etc.), utilities, information or documentation files, mailing list or usenet group discussion archives. At most FTP sites, the resources are organized hierarchically in directories and subdirectories. The database tracks both the directory path and the file names.

The archie database is automatically updated, thereby ensuring that the information is accurate. Using this database, users can easily find the the location of files they need without logging onto several machines.

Who can use ARCHIE

Users on any network can access the archie database by electronic mail. Other means of access are available to users on the Internet (see the section **Using ARCHIE** below for details).

You are requested to respect a few basic rules when you request information from an archie server:

- avoid connecting during working hours; most of the archie servers are not dedicated machines, they have local functions as well.
- make your queries as specific as possible; the reponse will be quicker and shorter.
- user interfaces installed on your computer contribute to reduce the load on the server sites, please use them.
- use the archie server closest to you and, in particularly, don't overload the transatlantic lines.

How to get to ARCHIE

The archie database is maintained in 15 different locations.

archie.au	Australia
archie.funet.fi	Finland
archie.th-darmstadt.de	Germany
archie.doc.ic.ac.uk	Great Britain
archie.cs.huji.ac.il	Israel
archie.wide.ad.jp	Japan
archie.kuis.kyoto-u.ac.jp	Japan
archie.sogang.ac.kr	Korea
archie.nz	New Zealand
archie.luth.se	Sweden
archie.ncu.edu.tw	Taiwan
archie.ans.net	USA
archie.rutgers.edu	USA
archie.sura.net	USA
archie.unl.net	USA

There are three ways to access the archie database: via a local client, interactive Telnet session or electronic mail. Each type of access is described below in the **Using ARCHIE** section.

Using ARCHIE

The format of the parameters is given at the end of this section. Angle brackets (<>) indicate an optional parameter; a vertical bar (|) indicates a choice of parameters.

Using a local client:

Usage of these clients is encouraged since they provide quick and easy non-interactive access to the archie servers, and thus, better performance of the servers and better response time for the user.

Public domain clients for accessing archie servers are available for: Macintosh, MS/DOS, OS/2, VMS, NeXT, Unix and X-Window.

The clients are available for anonymous FTP from the archie sites in the directory *archie/clients*.

Archie client command and parameters

The archie client is a command with parameters that you enter on your local machine. With most versions of the archie client, if you type **archie** with no parameters, you will get a list of the possible parameters and a short description of each.

The format of the command is:

archie	<-options> string | pattern

where the options are:

o specifies an output file name to store the results (not available with all clients).

l lists the result one match per line. This form is suitable for parsing by programs.

t sorts the result inverted by date

m# specifies maximum number of matches to return (# within the range 0 to 1000). The default value is 95.

h archie_server
specifies an archie server to send the query to; if this parameter is not given, then the query will be sent to the default archie server, if one is defined.

L lists known servers and current default

The following group of options determines the kind of search performed on the database. They are mutually exclusive.

s a match occurs if the file/directory name contains *string*. The search is case insensitive.

c as above, but the search is case sensitive.

e *string* must EXACTLY match (including case) the file/directory name in the database. This is the DEFAULT search method.

r searches the database using *pattern*. It contains special characters which must be interpreted before performing the search.

There may be some slight differences in the options available with different clients on different platforms.

The result is a list of FTP site addresses with entries matching the argument, the size of the resource, its last modification date and its directory. By default, the list is sorted by host address. See the **Examples** section below for an example of archie output.

Using Telnet:

To access an archie server interactively, telnet to one of the existing servers (see the list of servers in the section **How to get to ARCHIE** above). At the *login:* prompt enter **archie**, the login procedure end s leaving the user at a *archie>* prompt. The server is ready for the user requests and the following commands are available:

exit, quit, bye

exits archie.

list <pattern>

provides a list of the sites in the database and the time at which they were last updated. The optional parameter limits the list to specific sites.

The result is a list of site names, sites IP address and date of the last update in the database.

The command **list** with no *pattern* will list all sites in the database (more than 1000 sites!). The command: list \.de$ will lists all German sites.

site site-name

lists the directories and, recursively, the subdirectories, of site-name in the database. The result may be very long.

whatis string

searches the database of software package descriptions for *string*. The search is case-insensitive.

prog string | pattern

searches the database for *string* or *pattern* which represents the name of the resource to be found in the database. Searches may be performed in a number of different ways specified in the variable *search* (**set** command) which also decides the interpretation of the parameter as *string* or *pattern*.

The result is a list of FTP site addresses with matching entries, the size of the resource, its last modification date and the directory to find it. The number of hits is limited by the *maxhits* variable (**set** command).

The result of **prog** can be sorted in different ways, dependin g on the value of the *sortby* variable (**set** command).

By default, the variables *search*, *maxhits* and *sortby* are set to, respectively, exact match search on *string*, 1000 hits and unsorted resulting list.

Typing the keyboard interrupt character during a search will abort it. The results up to that time are displayed.

See the **Examples** section below for an example of the prog command and its results.

mail <email> <,email2...>

sends the result of the last command in a mail message to the specified e-mail address(es). If issued with no argument, the result is sent to the address specified in the variable *mailto* (**set** command).

show variable

displays the value of the given variable name. If issued with no argument, it displays all variables. See the **set** command below for the possible variables. set variable value sets one of the archie's variables. Values of these variables affect how archie interacts with the user.

Variables and values are:

mailto email <,email2 ...>
specifies the e-mail address(es) to mail the result of the last command when *mail* is issued with no arguments.

maxhits number
specifies the maximum number of matches **prog** will generate (within the range 0 to 1000). The default value is 1000.

search search-value
determines the kind of search performed on the database by the command: *prog string | pattern*. search-values are:

sub a partial and case insensitive search is performed with *"string"* on the database, eg:

```
"is" will match "islington" and "this"
and "poison"
```

subcase as above but the search is case sensitive, eg:

```
"TeX" will match "LaTeX" but not "Latex"
```

exact the parameter of **prog** (*string*) must EXACTLY match (including case) the string in the database. The fastest search method of all, and the default.

regex *pattern* is interpreted before performing a search on the database.

sortby sort-value
describes how to sort the result of **prog**. sort-values are:

hostname on the FTP site address in lexical order.

time by the modification date, most recent first.

size by the size of the found files or directories, largest first.

filename on file or directory name lexical order.

none unsorted (default)

The reverse sorting orders from those described here are obtained by prepending *r* to the *sortby* value given. (eg, reverse hostname order *hostname* is *rhostname*).

term terminal-type <number-of-rows <number-of-columns>>
tells the archie server what type of terminal you are using, and optionally its size in rows and columns, eg:

```
set term xterm 24 100
```

Using Electronic mail:

Users limited to electronic mail connectivity can access the archie servers by sending mail to the domain address of one of the servers listed in the section **How to get to ARCHIE**. The commands are sent in the body part of the mail.

An archie server recognizes eight commands: help, path, prog, whatis, list, site, compress and quit. Particularities to the electronic mail interface are described below. If an empty message, or a message containing no valid requests is received, it will be considered to be a *help* request.

Command lines begin in the first column. All lines that do not match a valid command are ignored. The *Subject:* line is processed as if it were part of the message body.

help
sends you the help file. The **help** command is exclusive, ie, other commands in the same message are ignored.

path return-address
specifies a return e-mail address different from that which is extracted from the message header. If you do not receive a reply from the archie server within several hours, you might need to add a path command to your message request.

prog pattern <pattern2 ...>
searches the database for each interpretation of *pattern* which represents the name of a resource to be found in the database.

If multiple *pattern*s are placed on one line, in that case, the results will be mailed back in one message. If multiple **prog** lines appear, then multiple messages will be returned, one for each **prog** line. Results are sorted by FTP site address in lexical order. If *pattern* contains spaces, it must be quoted with single (') or double (") quotes.
The search is case-sensitive.

whatis string <string2 ...>
searches the database of software packages descriptions for each *string*. The search is case insensitive.

list pattern <pattern2 ...>
provides a list of the sites in the database that match *pattern* and the time at which they were last updated.

The result is a list with site names, sites IP address and date of the last update in the database. lists the directories and, recursively, the subdirectories, of *site-name* in the database.

site site-name

compress
causes the result of the current request to be *compressed* and *uuencoded*. When you receive the reply, you should run it through *uudecode*. This will produce a .Z file. You can then run *uncompress* on this file and get the result of your request

quit
nothing past this point is interpreted. Useful when a signature is automatically appended at the end of your mail messages.

Description of *pattern*

A *pattern* describes a character string including characters which take a special meaning. The special meaning is lost when "\" is put before the character. The special characters are:

. (period) this is the wildcard character that replaces any other character, eg, "...." will match any 4 character string.

if " " appears at the beginning of the pattern, then the searched string must start with the substring following the " ". If it occurs anywhere else in the pattern it is regarded as non-special, eg:

```
" efghi" will match "efghi" or "efghijlk" but not
"abcefghi"
```

**$** If "$" appears at the end of the pattern, then the searched string must end with the substring preceding the "$". If occurring anywhere else in the pattern, it is regarded as non-special, eg:

```
"efghi$" will match "efghi" or "abcdefghi" but not
"efghijkl"
```

Examples

If you are using an archie client, and enter the command:

```
archie -s eudora
```

or if you send, by e-mail or during a Telnet session, the command:

```
prog eudora
```

then archie will send you the following results:

```
Host dorm.rutgers.edu

 Location: /pub/Mac/comm
   DIRECTORY drwxr-xr-x     512  Oct  2 1990  Eudora
 Location: /pub/Mac/comm/unsorted
        FILE -rwxr-xr-x  634517  Sep  1 1990  eudora.tar.Z

Host extro.ucc.su.oz.au

 Location: /pub
   DIRECTORY drwxr-xr-x     512  May  6 1992  eudora
 Location: /pub/eudora
        FILE -rw-r--r-- 1140708  Mar  6 1991  eudora1.1ctb1.i.hqx
        FILE -rw-r--r-- 1138727  Mar  6 1991  eudora1.1ctb2.i.hqx
        FILE -rw-r--r-- 1140546  Mar  6 1991  eudora1.1tcp.i.hqx
```

etc.

If you send the command "list \.de$" by e-mail or in a Telnet session, then you will get the following results:

```
41 currently stored site names match '\.de$'

alice.fmi.uni-passau.de            132.231.1.180  17:54 3 Mar 1993
askhp.ask.uni-karlsruhe.de         192.67.194.33  19:20 3 Mar 1993
athene.uni-paderborn.de             131.234.2.32  19:26 3 Mar 1993
charly.bl.physik.tu-muenchen.de  129.187.160.10  20:26 3 Mar 1993
coma.cs.tu-berlin.de               130.149.29.10  21:17 3 Mar 1993
dsrbg2.informatik.tu-muenchen.de  131.159.0.110  21:57 2 Mar 1993
```

etc.

If you send the command **whatis archie** by e-mail or in a Telnet session, then you will get the following results:

```
archie       A Prospero client for archie
awl          Layout language for widget hierarchies (X11)
perl_archie  An archie client in perl
xarchie      X11 client for the archie system
```

Learning more about ARCHIE

However you communicate with the archie server, on-line help is available.

If you have any questions about archie write to the Archie Group, Bunyip Information Systems Inc. at **info@bunyip.com**.

Mailing list: **archie-people@bunyip.com**
To subscribe send a mail to: **archie-people-request@bunyip.com**

Archie was developed by Alan Emtage, Peter Deutsch, and Bill Heelan from the McGill University Computing Center, Canada.

WHOIS

What is WHOIS

WHOIS provides directory service to network users. This service is a way of finding e-mail addresses, postal addresses and telephone numbers. It may also deliver information about networks, networking organizations, domains and sites.

The main database of networking-related names (organizations, sites, networks, people, etc.) is maintained by the Internet Registration Service (InterNIC). Actually, names in this database are automatically The names of the administrative and technical contacts for registered domains are automatically entered into the database when domain or IP number applications are processed by the Internet coordination authority. Each entry of the database has a *handle* (a unique identifier), a name, a record type, and various other fields depending on the type of record. This database will be used as an example in the descriptions below.

Before April 1, 1993, the Network Information Center (NIC) of the Defense Data Network (DDN) was the Internet coordination authority and, therefore, maintained the database (known as the NIC database). The NIC database is now restricted to information about the *.mil* domain. Many documents still refer to these names.

Many academic sites maintain their own database to offer information about their staff members and students.

Who can use WHOIS

WHOIS is available to users on the international TCP/IP network (the Internet).

A WHOIS server is accessible across the network from a user program running on local machines or via an interactive Telnet session to the site which hosts the server.

How to get to WHOIS

There are many WHOIS servers throughout the network and a comprehensive list would be too long to be included here. A WHOIS server offers information about the organization to which it belongs: it doesn't share a common directory with other WHOIS servers and doesn't know either where to find information about other institutions.

Using WHOIS

WHOIS has become the familiar name of the user program for accessing a WHOIS database, although NICNAME is the original name.

In the following, angle brackets (<>) indicate an optional parameter.

UNIX computers have a native whois command. On non-UNIX machines, ask your system administrator whether your computer has it or not. This command searches the database on the specified site for entry which contains identifier. The format is:

whois <-h sitename> identifier

where:

sitename the domain address of the site which hosts the database you want to query (eg, whois.internic.net). On some installations, the default value is set to the NIC database site (nic.ddn.mil).

identifier a name (person, host, domain or network), an IP number or a *handle*.

Special characters may be used to specify the search:

. before the identifier will cause a name-only search.

! before the identifier will cause a *handle*-only search.

... or . after the identifier will cause a partial search: everything starting with identifier will match.

@ in the identifier will cause a search on the e-mail addresses.

***** before the identifier will return the entire membership list of the entry that match identifier (eg, shows a site and its registered users).

% before the identifier will return only the entire membership list of the entry that match identifier (eg, shows the registered users of a site).

The special characters may be used together.

The results are displayed in one of 2 ways:

- a full detailed display for a single match,
- a list of summary lines for multiple matches.

In both cases, the *handle* is shown in parenthesis after the name to be used in further searches.

Telnet session

To access interactively the InterNIC database, telnet to the InterNIC site (whois.internic.net). There is no specific login. Other WHOIS databases have a Telnet access and offer most of the functions below (eg, whois.ripe.net which hosts the WHOIS database of the European IP Networks).

WHOIS invokes the information retrieval program.

? displays a short on-line help.

HElp accesses the full on-line help.

Q, QUIT, RETURN key
exits WHOIS

<keyword> identifier
searches the database for entry which contains identifier. The default action is to do a broad search, looking for matches in many fields: handle, name, nicknames, hostname, IP number, etc, and finding all record types. keyword may be used to narrow the search to a specific record type.

Special characters may be used in identifier to specify the search:

. before the identifier will cause a name-only search.

! before the identifier will cause a *handle*-only search.

... or . after the identifier will cause a partial search: everything starting with identifier will match.

@ in the identifier will cause a search on the e-mail addresses.

***** before the identifier will return the entire membership list of the entry that match identifier (eg, a host and its registered users).

% before the identifier will return only the entire membership list of the entry that match identifier (eg, registered users)

~ before the identifier will return the entry that match identifier only, no membership list.

The special characters may be used together.

keyword specifies the record type to limit the search. It may be one of (CAPITAL letters indicate acceptable abbreviation):

PErson limits the search to persons

DOmain limits the search to domains (eg, DO EARN.NET)

HOst limits the search to hosts (eg, HO PRINCETON)

NEtwork limits the search to networks (eg, NE EBONE)

Organization
limits the search to organizations (eg, O CREN)

NAme same as leading '.' in identifier.

HAndle same as '!' in identifier.

PArtial same as trailing '.' in identifier.

Mailbox same as '@' in identifier.

EXPand same as '*' in identifier.

SUBdisplay same as '%' in identifier.

Full or '=' shows detailed display for EACH match

SUMmary or '$'
shows summary always, even if just one match.

Except if Full or SUMmary are specified, the results are displayed in one of 2 ways,:

- a full detailed display for a single match,
- a list of summary lines for multiple matches.

In all cases, the *handle* is shown in parenthesis after the name to be used in further searches.

Etiquette

In general, WHOIS servers should only be used for isolated queries about specific information. Typically, it is not acceptable to make an extended series of queries to obtain large sections of the directory. Such a strategy is unfair both because of excessive consumption of server resources, and because the directory information belongs to individuals. In particular, extracting lists of people for commercial purposes is strictly prohibited.

Examples

whois \!EARN... (remark: "\" is an escape character)

```
EARN (EARN-HST)  SEINE.EARN.NET
130.84.8.204
European Academic Research Network (EARN-DOM)
EARN.NET
```

whois EARN-DOM

```
European Academic Research Network (EARN-DOM)
   EARN Office
   CIRCE BP 167
   91403 ORSAY CEDEX, France

   Domain Name: EARN.NET

   Administrative Contact:
      Deckers, Hans  (HD35)  deck@FRORS12.CIRCE.FR
      +33 1 6982 3973
   Technical Contact, Zone Contact:
      Bovio, Daniele  (DB355)  hi@FRORS12.CIRCE.FR
      +33 1 6982 3973

   Record last updated on 23-Oct-92.

   Domain servers in listed order:

   LUMIERE.CIRCE.FR  130.84.8.14
   SEINE.EARN.NET  130.84.8.204
```

whois hi@f...

```
Bovio, Daniele (DB355)  hi@FRORS12.CIRCE.FR
   EARN
   EARN Office
   CIRCE BP 167
   91403 ORSAY CEDEX, France

   +33 1 6982 3973

   Record last updated on 23-Oct-92.
```

Learning more about WHOIS

The WHOIS service is documented in an Internet Request For Comments (RFC 1400).

If you have any questions about whois write to **nic@nic.ddn.mil**.

Bug reports, comments, suggestions, etc. should be mailed to **suggestions@nic.ddn.mil**.

NETSERV

What is NETSERV

NETSERV is a server, which allows fast access to data files and programs of general interest. The main functions of the server can be divided into three general categories: File Server, Node Management and User Directory services.

As a file server, NETSERV provides a file repository consisting of information files, and programs. It allows users to retrieve files, to store files and to subscribe to the files of their choice. The latter two functions however, require the user to be a *privileged user* meaning, a user who has a password for NETSERV.

As a node management tool, NETSERV provides data files and programs for node administrators and network coordinators, and allows the registration of new nodes or changes in the network.

Finally, each NETSERV keeps a separate user directory of the users who have chosen to register themselves to the server. Users can specify a wide variety of information on themselves, permitting queries to be performed by others on any of the available fields.

In order to achieve a balanced load on the network and a faster response time to users, NETSERV uses a distributed server concept: this is achieved by the installation of a large number of servers on the network to ensure that the user can locate a nearby server. All servers communicate with each other to distribute updated information and make it available from each copy of the server.

NETSERV's file server functions include retrieving any file present in its filelists, storing new versions of a file, and subscribing to files stored on the server. Its file directories are arranged in an hierarchical method, with NETSERV FILELIST being on **top** or at the **root** of the filelists. This filelist can be obtained by sending a **GET NETSERV FILELIST** command to any NETSERV. Filelists contain short descriptions of the files, and two access codes for each file. These codes represent the GET and PUT privileged required for that file. These codes are explained at the beginning of the NETSERV FILELIST file.

NETSERV's user directory allows users to search for registered users on any of the available fields, such as name, network address, profession, phone number, etc.

Who can use NETSERV

The server is available in almost every country. To find the closest server for your area send a QUERY SERVICE command to a server. The following are examples of NETSERV server addresses:

```
NETSERV@FRMOP11.BITNET
NETSERV@HEARN.BITNET
NETSERV@BITNIC.BITNET
```

In general, there is only one NETSERV permitted for one country. However, in some limited cases, such as when the country has a large number of nodes, additional servers may be installed. In any case, the user is not required to run NETSERV to be able to access and use the server.

How to get to NETSERV

NETSERV accepts e-mail access from users on any network. Commands to NETSERV should be placed in the body of the mail file, and not in the *Subject:* line.

For users in the EARN/Bitnet network, NETSERV is accessible via interactive message. Commands from privileged users requiring a password must be sent this way.

NETSERV does not have delivery limitations, except that a file ordered from NETSERV cannot be ordered again on the same day.

Using NETSERV

NETSERV offers a complete set of commands for every task. In particular, its HELP feature is thorough: a short help info about a single command or message can be requested by the command **HELP** *helptype* where *helptype* is either a command, a sub-command or the message number that was displayed by the server.

Commands For General Users:

Use the **GET** command to retrieve a file, or a filelist stored on NETSERV. The format of the command is:

GET<xx> filename filetype <prologtext>

where:

XX (optional)
defines the transfer format of the file, and can be one of: ND for NETDATA, DD for DiskDump, PP for Punch or Print,LP for LISTSERV-Punch, and 80 to split records at 80 columns, in Punch format. If XX is omitted then NETSERV will use the format best suited to your site.

filename is the filename of the file.

filetype is the filetype of the file.

prologtext is any text that you wish placed at the top of the file.

Use the **GET NODENTRY** command to extract an entry for a node in the Master Nodes file, which contains complete network and contact information about all nodes in EARN, BITNET and cooperating networks. The format of the command is:

GET NODENTRY entryid

where:

entryid is the name of a node in the BITEARN NODES file.

Use the **QUERY** command to display various information about the server. The format of the command is:

QUERY keyword

where *keyword* determines the behaviour of the command. Acceptable keywords are:

CMD to see the available NETSERV commands,

CONTROLLER
to display the network address of this NETSERV's Controller,

FILE filename filetype
to display information about a file stored on NETSERV

SERVICE <nodeid>
to provide information about which countries or sub-networks this NETSERV provides service for, and which NETSERV is responsible for your node area, or for another node specified by **nodeid**.

Use the **SERVE** command to have service reinstated, if your access to NETSERV has been suspended. This command must be issued from another user address. The format of this command is:

SERVE user@node

where:

user@node is your address.

Use the **UDS** command to add information about yourself to the server, or to update or delete it. The format of the command is:

UDS	option keywords

where:

option can be ADD, REP, FIND, GET or DEL.

keywords are the keywords that may be required for that option.

The format of the ADD option is:

```
UDS ADD
:USERID.xxxxxxxx  :NODE.xxxxxxxx
:NAME.firstname middle lastname
:PHONE.phone number
:ADDR.institute name;city;country
:DESCR.job title;activities;interests;etc
```

The GET and FIND options can be used to locate other users on the network. FIND only returns one line per entry, whereas GET returns entire entries for the matching users. These options must be followed first by the maximum desired matching and also by the search criteria. The criteria must be specified as:

:fieldname searchvalue <:fieldname searchvalue ...>

The fieldname can be any of the ones listed in the above example, and the search value can be a subset of the actual sought value, such as *SM* instead of *Smith*. Example:

```
UDS FIND 30 :name Harry :addr New York
```

The search is case-insensitive and additional blanks are ignored.

NETSERV has a wide variety of clients that provide a local interface to the service. The most widely used one is called **NETSRV EXEC** and runs on IBM/VM systems. It features online help and facilitates sending commands to NETSERV. It is available via **GET NETSRV EXEC** command from any NETSERV.

The **NODES EXEC** program provides a full screen interface to information within the BITEARN NODES file.

Examples

If you want to find out what NETSERV server provides service for your node, then send the command:

```
QUERY SERVICE
```

to: NETSERV@HEARN.BITNET (or to any other NETSERV server).

If you want to get details on the node ROEARN, then send the command:

```
GET NODENTRY ROEARN
```

to your nearest NETSERV.

If you want to get an up-to-date listing of all nodes in the international NJE network, then send the command:

```
GET BITNET NODELST
```

to your nearest NETSERV.

Learning more about NETSERV

The server provides a large helpfile which can be obtained by sending a **GET NETSERV HELPFILE** command to any NETSERV. A list for NETSERV maintainers is available as NETSRV-M@HEARN.BITNET. Additional information can be obtained from the NETSERV maintainer, Ulrich Giese <U001212@HEARN.BITNET>.

Part 4
GETTING FILES

TRICKLE

What is TRICKLE

TRICKLE is a service which will send you files on request, or by subscription. TRICKLE works with various anonymous FTP sites, computers in the Internet network that allow public access and retrieval of software and files. It provides a quick and easy alternative to FTP, whether or not you have access to the Internet.

There are several TRICKLE servers throughout the world that cooperate to distribute the files efficiently. To request files the user issues commands to the nearest TRICKLE server, which delivers the software either from its local cache disk, from the cache of another TRICKLE server, or from an FTP site that holds the software.

Who can use TRICKLE

There are currently TRICKLE servers at the following addresses:

Location	EARN/BITNET	Internet
In Turkey:	TRICKLE@TREARN	TRICKLE@ege.edu.tr
In Italy:	TRICKLE@IMIPOLI	
In Belgium:	TRICKLE@BANUFS11	TRICKLE@ccs.ufsia.ac.be
In Austria:	TRICKLE@AWIWUW11	
In Germany:	TRICKLE@DEARN	TRICKLE@vm.gmd.de
In Israel:	TRICKLE@TAUNIVM	TRICKLE@vm.tau.ac.il
In Netherlands:	TRICKLE@HEARN	TRICKLE@hearn.nic.surfnet.nl
In France:	TRICKLE@FRMOP11	TRICKLE@frmop11.cnusc.fr
In Colombia:	TRICKLE@UNALCOL	TRICKLE@unalcol.unal.edu.co
In Sweden:	TRICKLE@SEARN	TRICKLE@searn.sunet.se
In Turkey:	TRICKLE@TRMETU	TRICKLE@3090.cc.metu.edu.tr

When you send a command to a TRICKLE server, it either executes the command or sends you a message with the address of the TRICKLE server for your area.

The files which are available from TRICKLE are organized in main directories which contain many subdirectories. The main directories which are currently available are:

Directory	Source FTP Site	Contents
MSDOS	simtel20.army.mil	Large MS-DOS software archive
MISC	simtel20.army.mil	Software for VM,VMS,UNIX
SIGM	simtel20.army.mil	SIG/M CP/M archive
PC-BLUE	simtel20.army.mil	PC-BLUE MS-DOS archive
CPM	simtel20.army.mil	CP/M Software Archive
ARCHIVES	simtel20.army.mil	Various discussion group archives
UNIX-C	simtel20.army.mil	Unix and C code software archive
MACINTOS	simtel20.army.mil	ApLple Macintosh software archive
OS2	ftp-os2.nmsu.edu	Large archive of OS/2 software
AMIGA	nic.funet.fi	Large Amiga collection
KERMIT	watsun.cc.columbia.edu	Kermit network software
TEX	rusinfo.rus-uni-stuttgart.de	TeX software and fonts
WUARCHIVE	wuarchive.wustl.edu	MS-DOS and others
EXPO-MIT	export.lcs.mit.edu	Unix and others
UUNET	ftp.uu.net	Unix and others
SUMEX-AIM	sumex-aim.stanford.edu	Macintosh and others

How to get to TRICKLE

You send commands to TRICKLE by electronic mail. The commands should be in the body of the mail message, one command per line. Any number of commands (up to your daily command limit) may be placed in one message. Note that all TRICKLE commands begin with a slash (/). Users on the EARN/Bitnet network may also send commands to TRICKLE by interactive message.

Using TRICKLE

All commands begin with a slash (/). Note that the angle brackets (<>) are part of the command, NOT an indication of an optional parameter.

/PDDIR	obtain a directory listing
/PDGET	request a file
/SUB	subscribe to automatically get new files
/UNSUB	to cancel a subscription

Important Commands:

Use the **/PDDIR** command to obtain directory listings.

/PDDIR	
/PDDIR	<dirname>
/PDDIR	<dirname.subdirname>pattern

where:

dirname is the name of a main directory,

subdirname is the name of a subdirectory,

pattern is the first characters of a filename, to limit the directory listing.

Use /PDDIR without any parameters to get a listing of the main directories. With: PDDIR <dirname> you will get a listing of the subdirectories under that directory. If you specify both directory and subdirectory, it will list the files that are available in that subdirectory. A pattern may be used to reduce the size of the listing.

Use the **/PDGET** command to get files.

/PDGET	<dirname.subdirname>filename (delivery_option

where:

dirname is the name of a main directory

subdirname is the name of a subdirectory

filename is the name of a file

delivery_option
specifies the format to which the file(s) should be translated before being sent to you. The possible values are:

```
EBC80  UUE  XXE  HEX  BTOA
```

The option EBC80 should be used to get text files if you work on an IBM mainframe system. The other options are formats for translating binary files so that they can be sent via electronic mail. You will need a program to translate the file back to its original form. The default for EARN/Bitnet users is to send the file as-is. The default for others is UUE.

You can use an asterisk (*) as a wildcard character in the *subdirname* or *filename* fields. This will cause TRICKLE to send the file(s) matching your specification.

The **/SUB** command is used to subscribe both to directories and to individual files.

/SUB	<dirname>
/SUB	<dirname.subdirname>pattern
/SUB	QUERY

where:

dirname is the name of a main directory,

subdirname is the name of a subdirectory,

pattern is the first characters of a filename

If you subscribe to a directory, you will get a summary of added files about once a week, depending on how active the FTP site is. This listing shows the names, sizes and dates of each file added. If you subscribe to files, a new version of the file will be sent to you as soon as TRICKLE is informed that it has been stored at FTP site.

The *pattern* consists of the first few letters of a filename. Since filenames usually reflect the version number of the file, it is a good idea to omit the number when specifying a pattern. For example, it is better to send a

```
/SUB <MSDOS.VIRUS>SCANV
```

command rather than

```
/SUB <MSDOS.VIRUS>SCANV102
```

since, as new versions of the file are stored, the name might change to SCANV103, SCANV104, etc.

The /SUB QUERY command allows you to get a list of the files you are subscribed to.

The **/UNSUB** command may be used to cancel a subscription.

/UNSUB <dirname>

/UNSUB <dirname.subdirname>pattern

where:

dirname is the name of a main directory,

subdirname is the name of a subdirectory,

pattern is the first characters of a filename

Optionally, you can put a '*' instead of the pattern to remove all entries for that subdirectory.

Examples

For a listing of the files in the VIRUS subdirectory of SIMTEL20 (the MSDOS directory), send the command:

```
/PDDIR <MSDOS.VIRUS>
```

To get the file 00-INDEX.TXT from <MSDOS.VIRUS> in EBCDIC format, send the command:

```
/PDGET <MSDOS.VIRUS>00-INDEX.TXT ( EBC80
```

To subscribe to automatically get new versions of the SCANV software from <MSDOS.VIRUS>, send the command:

```
/SUB <MSDOS.VIRUS>SCANV
```

To unsubscribe from all files in the VIRUS subdirectory, send the command:

```
/UNSUB <MSDOS.VIRUS>*
```

Learning more about TRICKLE

The /HELP command may be sent to any TRICKLE server to obtain a very detailed help file from the server. The LISTSERV@EARNCC.BITNET also holds a file describing this server. This file can be obtained by sending a *GET TRICKLE MEMO* command to **LISTSERV@EARNCC.BITNET**.

BITFTP

What is BITFTP

BITFTP, BITNET FTP Server, allows users of EARN, Bitnet and associated networks to access FTP sites on the Internet.

The BITFTP server provides a mail interface between a user on the EARN/Bitnet network to FTP sites on the Internet. BITFTP handles this task by passing the commands specified in the mail message from the user to another server that actually makes the connection to the FTP sites. When the server finishes the interaction with the FTP site, or fails due to an error, a transcript of the result is sent back to the user, as well as the requested file(s), if any.

The format of the files sent to the user can be defined within the mail message. BITFTP can deliver files in NETDATA and UUENCODE formats.

Who can use BITFTP

BITFTP is currently available only to users on EARN, Bitnet and other regional NJE networks. It currently does not accept access to European FTP archive sites from European users. Further, it does not support putting files onto FTP sites or the mget (*multiple get*) command.

How to get to BITFTP

There are BITFTP currently running at:

```
BITFTP@PUCC.BITNET
BITFTP@DEARN.BITNET
```

Servers at other locations are planned.

BITFTP accepts requests via electronic mail, including IBM NOTE and PROFS-format messages, as well as by file transfer.

Using BITFTP

BITFTP implements a large subset of the FTP commands of the IBM's TCP/IP for VM, using the same syntax. This software is documented in the IBM manual **TCP/IP for VM User's Guide**. The important commands are:

FTP	connect to an FTP site
USER	give user-id and password
CD	change directory
DIR	get a detailed listing of the directory
LS	get a listing of the directory
GET	get a file from the FTP site
BINARY	set file transfers to binary mode
QUIT	end the connection to the FTP site

Command description:

Use the **FTP** command to specify which host to connect to. This command must be the first command in your mail file. You can also specify the file format that you wish BITFTP to use to deliver files to you. Angle brackets (<>) indicate that a parameter is optional.

FTP	hostname <fileformat>

where:

hostname is either the IP address or the domain name of the host to connect to.

fileformat is the format in which BITFTP should deliver files to you. It can be either NETDATA or UUENCODE.

Use the **USER** command to tell the host the username and the password to be used for the FTP connection. Note that on Unix systems, both username and password are case-sensitive.

USER	username password

where:

username is the user-id to use for the FTP connection.

password is the password for the username you specified. It can be omitted if you specified *anonymous* for the userid.

Use the **CD** command to select a particular directory as *current* directory.

CD	directory-name

where:

directory-name
is the name of the directory to be selected.

Use the **DIR** command to display a list of the files in the current, or specified, directory. The file names, and depending on the site, the file size, file creation date and other information will be listed.

DIR	<pattern>

where:

pattern is a pattern that defines which file names should be displayed. Many FTP sites are case-sensitive, thus care must be used with the pattern. The pattern may contain any number of characters, and the wildcard character asterisk (*) may be used to represent any characters.

The **LS** command is similar to the DIR command, except that with most FTP sites, it only displays the filenames, without any other information. The pattern specification is identical to DIR.

LS	<pattern>

where:

pattern is a pattern that defines which file names should be displayed. Many FTP sites are case-sensitive, thus care must be used with the pattern. The pattern may contain any number of characters, and the wildcard character asterisk (*) may be used to represent any characters.

Use the **GET** command to obtain a file from the current host.

GET foreignname <localname>

where:

foreignname is the filename of the file, as stored at the FTP site. With many sites, the case of the filename must be respected.

localname is an optional local filename. If you specify a localname, it must be in the form: filename.filetype where neither part is longer than eight characters.

Use the **BINARY** command to set the FTP transfer mode. If this option is set then no EBCDIC-ASCII translation will take place. This should be used for non-text files.

BINARY

Use the **QUIT** command to close the connection to the host, and to terminate the BITFTP session.

QUIT

Here is a sample BITFTP command file:

```
FTP nis.nsf.net
USER anonymous
cd introducing.the.internet
get intro.to.ip
get network.gold
get where.to.start
get zen.ps
get zen.txt
QUIT
```

Learning more about BITFTP

A four-page instruction to the BITFTP service can be obtained by sending a HELP command in the body of a mail message to the **BITFTP@PUCC.BITNET** address.

Additional information on BITFTP may be obtained from EARN (NETHELP@EARNCC.BITNET). More information about TCP/IP and FTP in general can be obtained from a wide variety of sources, including the IBM **TCP/IP for VM User's Guide**, and the FTP site **nis.nsf.net** which provides good introductory documents on the subject.

Part 5
NETWORKED INTEREST GROUPS

NETNEWS (USENET)

What is NETNEWS

NETNEWS, or USENET as it is more commonly called, is a collection of *newsgroups* distributed electronically around the world. Netnews provides a means for local users to communicate with each other, and with other users nationally and internationally.

Usenet was developed for Unix systems in 1979 by two graduate students at Duke University. Within a year, fifty Unix sites, including Bell Labs, were participating. Now, there are thousands of sites running a number of operating systems on a variety of hardware platforms communicating via Usenet around the globe.
Site administrators control their own sites. No one has any real control over any site but his own.

Sites are not entirely without influence on their neighbors, however. There is a vague notion of *upstream* and *downstream* related to the direction of high-volume news flow. To the extent that *upstream* sites decide what traffic they will carry for their *downstream* neighbors, those *upstream* sites have some influence on their neighbors' participation in Usenet.

There are many misconceptions about Usenet. Despite the myths: Usenet is not an organization. Usenet is not a democracy. There is no person or group in charge of Usenet as a whole. Usenet is not an academic network. Although many Usenet sites are universities, research labs or other academic institutions, by plain count, most Usenet sites are commercial entities. Usenet is not the Internet. The Internet is only one of the various networks carrying Usenet traffic. For example, EARN carries Usenet traffic too. Usenet is not a UUCP network. UUCP is a protocol for sending data over point-to-point connections, typically using dialup modems. Sites use UUCP to carry many kinds of traffic, of which Usenet is only one. Usenet is not a UNIX network. Usenet can be found on many operating systems. Usenet is not software. There are dozens of software packages used at various sites to transport and read Usenet articles. So no one program or package can be called *the Usenet software.*

Who can use NETNEWS

Usenet newsgroups can be read at thousands of sites around the world. In addition, there are several sites that provide public dial-up service, so that people who are not at a Usenet site can have access to newsgroups as well.

How to get to NETNEWS

If your site provides Usenet access, then there are a large number of software packages available for reading news (at least one is probably available on your computer). These packages either access a local news spool, or use the NNTP protocol to access the news spool on some other computer in the network.

Within EARN, a network of Netnews distribution has been developed, providing efficient distribution of Usenet traffic while minimising the load on the network for the participating countries.

Using NETNEWS

There are many software packages available for reading and distributing Netnews on a variety of platforms. Most, if not all, of these packages provide the same basic abilities:

- *subscription* to news groups, so that you can choose to read the postings of groups that interest you quickly and easily.
- keeping records of which postings you have already read.
- *threads* of discussion, so that you can follow groups of postings that deal with the same subject.
- posting to news groups, so that you can easily partiipate in group discussions.

The following, excerpted mostly from the article **USENET Software: History and Sources**, by Gene Spafford, is not an exhaustive list of the existing news packages:

Unix:

Several popular screen-oriented news reading interfaces have been developed in the last few years to replace the traditional **readnews** interface. The first of these was **vnews** and it was written by Kenneth Almquist. **vnews** provides a **readnews**-like command interface, but displays articles using direct screen positioning. It appears to have been inspired, to some extent, by the **notes** system (described below). **vnews** is currently distributed with the standard 2.11 news source.

A second, more versatile interface, **rn**, was developed by Larry Wall (the author of Perl) and released in 1984. This interface also uses full-screen display with direct positioning, but it includes many other useful features and is very popular with many regular net readers. The interface includes reading, discarding, and/or processing of articles based on user-definable patterns, and the ability of the user to develop customized macros for display and keyboard interaction. **rn** is currently at release 4.4. It is being maintained by Stan Barber of the Baylor College of Medicine. **rn** is not provided with the standard news software release, but is very widely available because of its popularity. The software can be obtained from its official archive site, lib.tmc.edu, using FTP, and via mail from **archive-server@bcm.tmc.edu**

A variant of rn is **trn** by Wayne Davison. Trn adds the ability to follow **threads of discussions** in newsgroups; its latest version 2.2 is based on rn 4.4. It uses a Reference-line database to allow the user to take advantage of the **discussion tree** formed by an article and its replies. This results in a true reply-ordered reading of the articles, complete with a small ascii representation of the current article's position in the discussion tree. Trn can be obtained from ftp.coe.montana.edu in the /pub/trn directory, from uunet in the news subdirectory, and from many other archive servers world-wide.

There are two popular macro packages named **GNUS** and "Gnews" that can be used with the GNU Emacs text editor. These allow reading, replying, and posting interaction with the news from inside the Emacs text editor. Client code exists to get the articles using NNTP rather than from a local disk. Copies can be found on most archive sites that carry the GNU archives.

nn is yet another reader interface, developed by Kim F. Storm of Texas Instruments A/S, Denmark, and released in 1989. **nn** differs from the traditional readnews and vnews by presenting a menu of article subject and sender-name lines, allowing you to preselect articles to read. **nn** is also a very fast newsreader, as it keeps a database of article headers on-line. (I.e. it trades space for time. A good rule of thumb is that the **nn** database size is 5%-10% of your news spool. So up to 110% of your news spool is the amount of space news and the **nn** database will take.) The current version of **nn** is 6.4.16. **nn** can be obtained via anonymous FTP from dkuug.dk, uop.uop.edu, or various other sites.

Yet another newsreader is the **tin** reader. It operates with threads, has different article organization methods, and is full-screen oriented. **tin** works on a local news spool or over an NNTP connection. It has been posted to alt.sources, and further information is available from Iain Lea (iain%anl433.uucp@Germany.EU.net). The current release of **tin** is 1.1 PL5. **Tin** is based more on the Notes and tass systems than **rn**. There is an extensive list of features, including interfaces to batch modes and auto unpacking mechanisms.

X-window:

xrn is an X11-based interface to NNTP that was written by Rick Spickelmier and Ellen Sentovich (UC Berkeley). The current version is 6.17. **xrn** supports many features, including sorting by subject, user-settable key bindings, graceful handling of NNTP server crashes, and many of the features of **rn** (including KILL files and key bindings similar to **rn**). **xrn** is actively supported by the authors with bug fixing and feature addition support from many of the users. **xrn** can be retrieved from most of the popular FTP sites (gatekeeper.dec.com, ftp.uu.net, export.lcs.mit.edu).

Another X11-based newsreader is **xvnews**. This is a news reader designed primarily for Sun workstations running OpenWindows. It runs with NNTP and is compatible with rn style commands. It is available from export.lcs.mit.edu in the contrib directory.

VMS:

ANU-NEWS is a news package written by Geoff Huston of Australia for VMS systems. **ANU-NEWS** is complete news system that allows reading, posting, direct replies, moderated newsgroups, etc. in a fashion closely related to regular news. The implementation includes the RFC 1036 news propagation algorithms and integrated use of the NNTP protocols (see below) to support remote news servers, implemented as a VAX/VMS Decnet object. An RFC 977 server implemented as a Decnet object is also included. The **ANU-NEWS** interface is similar to standard DEC screen oriented systems. The license for the software is free, and there are no restrictions on the re-distribution. For more info, contact gih900@fac.anu.oz.au (Geoff Huston). **ANU-NEWS** is available for FTP from kuhub.cc.ukans.edu. Contact SLOANE@KUHUB.CC.UKANS.EDU for more info.

A screen-oriented NEWS client for VMS is also available via FTP from ftpvms.ira.uka.de (contact Bernd Onasch <ONASCH@iravcl.ira.uka.de> for details).

Reader NNTP clients for VMS are also available, including VMS/VNEWS (current release 1.4) and an upcoming reader only version of **ANU-NEWS**. **VMS/VNEWS** is available for anonymous FTP from arizona.edu (contact jms@arizona.edu for more information) or at any site which archives vmsnet.source. Although the current release of **ANU-NEWS** is usable as a reader it can be difficult when used with a UNIX server.

VM/CMS:

There is an IBM VM/CMS version of the Usenet software which is widely available. It is known as **PSU NetNews**, and it is maintained by Linda Littleton (lrl@psuvm.bitnet/psuvm.psu.edu). Version 2.4 of the software is available from LISTSERV@PSUVM. **PSU NetNews** supports only 3270 terminals, and uses XEDIT as its screen driver. Most major VM sites appear to use this package. NetNews supports locally-stored news, not NNTP reading.

There is NNTP support for **PSU NetNews** for bulk news receipt: NNTPXFER will poll another site for news, and NNTPRCVR will receive news sent from a Unix NNTPXMIT process. Either program sends the news articles to NetNews for processing. Contact Andy Hooper (hooper@qucdn.bitnet or qucdn.queensu.ca) for more information, or obtain them from listserv@qucdn in PUBLIC FILELIST. These programs are provided with source, and require IBM's FAL TCP/IP and Pascal. An NNTPXMIT sender that works in cooperation with **PSU NetNews** is available from Herman Van Uytven (SYSTHVU@cc1.kuleuven.ac.be).

There is at least one NNTP news-reader for VM using XEDIT as its screen manager: **NNR**. Contact Paul Campbell (pc@mbunix.mitre.org) for information. The program requires IBM's FAL TCP/IP. The software is available for anonymous FTP from rusmv1.rus.uni-stuttgart.de 129.69.1.12] in the directory: **soft/kommunikation/news/beginner/software/vm-cms/***.

MVS:

An NNTP news reader is available for TSO/ISPF, called **NNMVS**. **NNMVS** is written by Stephen Bacher <seb1525@mvs.draper.com> at Draper Laboratory. It requires C/370 V1R2 (though V1R1 will work) and uses TCP/IP sockets. It is now available for anonymous FTP at ftp.uni-stuttgart.de under the directory **/soft/kommunikation/news/beginner/software/nnmvs**. The current release is Version 2 Release 3 Modification Level 1 (in pseudo-IBM parlance).

Macintosh:

An NNTP newsreader for Macintoshs is available called **News**. It is implemented as a HyperCard stack and depends on MacTCP. It is available from many Mac archives, including **ftp.apple.com** and **sumex-aim.stanford.edu**

MSDOS:

Trumpet is an NNTP news reader for MSDOS machines by Peter R. Tattam (peter@psychnet.psychol.utas.edu.au). It requires the use of a packet driver. It provides a user-friendly environment for news-reading and also supports the use of a mouse for point-and-click use. It is available as shareware from **tasman.cc.utas.edu.au.**

Note that the number of software packages available to run news, especially on PCs, is increasing. We have mentioned only a few of the many news packages available, and the presence or absence of any particular software package should not be construed as indicating anything about its suitability or usefulness.

In addition to the packages for Usenet access mentioned above, many other communications programs provide the possibility for Usenet access as well as their main function.

Learning more about NETNEWS

News programs communicate with each other according to standard protocols, some of which are described by RFCs. Copies of RFCs are often posted to the net and obtainable from archive sites. Current news-related RFCs include the following:

RFC 977 specifies NNTP, the Network News Transfer Protocol.
RFC 1036 specifies the format of Usenet articles.

Several articles on USENET are posted periodically to the newsgroup:
news.announce.newusers.

LISTSERV

What is LISTSERV

LISTSERV is a distribution list management package. It runs on IBM VM/CMS systems in the international NJE network (Bitnet/EARN). It allows groups of computer users with a common interest to communicate among themselves, while making efficient use of computer and network resources. It makes it easy for even novice users to discover, join, and participate in these interest group mailing lists. LISTSERV also provides facilities for logging and archiving of mail traffic, file server functions, and database searches of archives and files.

There are LISTSERV lists on every subject under the sun, with names ranging from AARPUB-L (AAR Electronic Publication list) to Z3950IW (Z39.50 Implementors Workshop). LISTSERV lists are international and eclectic. You will find lists in every imaginable field, for every audience, in many different countries and in many different languages.

See the description of the LIST command below for instructions on getting an up-to-date list of lists.

Who can use LISTSERV

Anyone who can send electronic mail to a Bitnet/EARN address can participate in a mailing list and access other LISTSERV facilities, as long as the message format is valid (according to the RFC822 standard), and has a usable return address. Every day, people use LISTSERV from HEPNET, Internet, Compuserve, MCIMAIL and many other networks throughout the world.

Ask your local support people if you don't know how to send electronic mail to EARN/Bitnet.

How to get to LISTSERV

To use LISTSERV facilities, send electronic mail with your LISTSERV commands to: LISTSERV@host-id where *host-id* is the host computer's NJE address (eg. TAUNIVM.BITNET) or its Internet domain name (eg. VM.TAU.AC.IL). There may be some local variation on the format needed to send mail to Bitnet or Internet addresses. Check with your local support people.

More than one command can be sent to LISTSERV in the same mail message. Each command must be on a separate line. LISTSERV will ignore the *Subject:* line of the mail header.

For Bitnet/EARN users, interactive messages are the fastest and most convenient way to send commands to LISTSERV, but bear in mind that interactive messages only work when the links between your computer and LISTSERV are up; if the message fails, you can always send your command via mail. To send interactive messages from an IBM computer running VM/CMS, the format is:

```
TELL LISTSERV AT node_id listserv_command
```

On a VAX VMS system, the format is:

```
$  SEND LISTSERV@node_id "listserv_command"
```

The LISTSERV command must be in quotes.

Note: To join a list, you send a message with your SUBscribe command to LISTSERV. To make an announcement to the members of a list, you send mail to the list address. For example, to join the EARN Users Group list (list name: EARN-UG, host site: IRLEARN), send a SUBscribe command to LISTSERV@IRLEARN.BITNET This can be done interactively, from VM:

```
TELL LISTSERV AT IRLEARN SUB EARN-UG Name Lastname
```

or from VMS:

```
$ SEND LISTSERV@IRLEARN "SUB EARN-UG Name Lastname"
```

or by mail, to LISTSERV@IRLEARN.BITNET, with the line:

```
SUB EARN-UG Name Lastname
```

in the text of the message.

To send a message to all list members, send mail to:

```
EARN-UG@IRLEARN.BITNET
```

There are more than 250 sites in over 30 countries throughout the world running LISTSERV as part of the world-wide network. Here is a list of some of the LISTSERV sites:

Host Computer	Site	Country
DEARN	GMD Bonn	Germany
BITNIC	BITNET Network Information Center	United States
HEARN	Katholieke Universiteit Nijmegen	Netherlands
CUNYVM	City University of New York	United States
PUCC	Princeton University, New Jersey	United States
ICNUCEVM	CNUCE Istituto del CNR, Pisa	Italy
UKACRL	Rutherford Appleton Laboratory	Great Britain
QUCDN	Queen's University Computing Services	Canada
FRMOP11	CNUSC, Montpellier	France
TREARN	Ege University Bornova,Izmir,	Turkey
UCHCECVM	Universidad de Chile, Santiago	Chile
TWNMOE10	Ministry of Education, Taipei	Taiwan
JPNSUT00	Science University of Tokyo	Japan
KRSNUCC1	Seoul National University	South Korea
UFRJ	Universidade Federal do Rio de Janeiro	Brazil
SEARN	Kungliga Tekniska Hoegskolan Stockholm	Sweden
ANDESCOL	Universidad de Los Andes, Bogota	Colombia
SAKFU00	King Faisal University, Hofuf	Saudi Arabia
HKUVM1	The University of Hong Kong	Hong Kong
PTEARN	Faculdade de Ciencias de Lisboa	Portugal
GREARN	Institute of Computer Science, Heraklion	Greece
PLEARN	Warsaw University	Poland
AEARN	Vienna University, Wien	Austria
CSEARN	Czech Technical University, Prague	Czech Republic
HUEARN	Hungarian Academy of Sciences	Hungary
TAUNIVM	Tel Aviv University	Israel
EARNCC	EARN, Paris Office	France

Using LISTSERV

The following summary of LISTSERV commands is divided into list subscription commands, informational commands, and file and filelist commands. This is followed by a brief description of the LISTSERV database facilities.

The list of commands included in this document is not exhaustive.

CAPITAL letters indicate acceptable abbreviation; angle brackets: < > indicate an optional parameter; vertical bar (|) indicates a choice of parameters. Format of parameters is given at the end. There are two basic versions of LISTSERV in use: **LISTEARN 1.3** and **Revised LISTSERV 1.7**. The two versions differ only in certain command parameters. Parameters marked with a (#) are for LISTEARN 1.3 only, those marked (+) are available only in **Revised LISTSERV 1.7**. To find out which version is installed at the server you wish to use, send the command: SHOW VERSion

List subscription commands:

Use the **SUBSCRIBE** command to join a list:

```
SUBscribe     listname  Your full_name
```

where *listname* is the name of a mailing list. Note that you do not send your electronic mail address. LISTSERV automatically determines your electronic mail address from your mail (or interactive message.)

Subscription to a list can be OPEN, CLOSED, or BY-OWNER. If it is OPEN, you will be automatically added to the list and sent notification. If it is CLOSED, you will not be added to the list, and LISTSERV will send you a message telling you that your request has been rejected. If it is BY-OWNER, your subscription request will be forwarded to the list owner(s), who will decide whether or not to add you to the list (LISTSERV will inform you to whom your request has been forwarded). To see what kind of subscription a list has, use the REView command. If you send a SUBscribe command for a list to which you already belong, then LISTSERV will interpret the command as a request to change your *full-name* on the list.

You can send your SUBscribe command to the LISTSERV site that manages the list you wish to join, or to any *backbone* LISTSERV site. All backbone sites have an up-to-date listing of all non-confidential lists at every LISTSERV site. If there are several sites with (unconnected) lists of the same name, then LISTSERV will send you a list of those sites and advise you to send your SUBscribe command direct. Otherwise, the backbone LISTSERV will automatically forward your request to the appropriate site.

When you join a list, LISTSERV at that site adds your name and e-mail address to its *signup* file. Once you have joined a list at a particular site, you need not include your name in subsequent SUBSCRIBE commands for lists at that site. LISTSERV will use the name it has for you in its *signup* file.

Use the **UNSUBSCRIBE** (or **SIGNOFF**) command to leave a mailing list:

```
UNSUBscribe     listname | *  <(NETWIDE>

SIGNOFF
```

where *listname* is the name of a mailing list. You can signoff all the lists you belong to at a particular LISTSERV by using the '*' (asterisk) character in place of a list-name. If you want your UNSUBscribe command to be propagated to all LISTSERV sites on the network, include the **(NETWIDE** option.

Use the **SET** command to change your personal options for a mailing list. The format of this command is:

SET	list-name | * < options >

where *list-name* is the name of the mailing list for which you are changing your options. You may change your options for a specific list or for all the lists you belong to at a particular LISTSERV by using the '*' (asterisk) character in place of a listname. The Query command can be used to dislay your current option settings for any list. The *options* you include with the SET command may be one or more of the following:

Mail/NOMail
: This indicates whether you want to receive mail from the list or not. The default is MAIL.

Files/NOFiles
: This indicates whether or not you want to receive files from the list. Use NOFiles if your system can not accept non-mail files. The default is FILES.

ACK/NOACK/MSGAck
: This indicates the kind of acknowledgement you want to receive from LISTSERV when you send mail to a list. ACK means that you want to receive a mail message from LISTSERV indicating if your message was successfully processed. NOACK specifies that no feedback should be sent to you about the processing of your message. MSGAck is meaningful for EARN/BITNET users only. With MSGAck, the server will send you the acknowledgement via interactive message. The default is ACK.

REPro/NOREPro
: Indicates whether or not you want to receive a copy of the mail which you send to the list. The default is NOREPro, meaning that mail which you send to the list will be distributed to all members of the list but you.

SHORThdr/FULLhdr/IETFhdr
: Indicates the type of headers you want to receive on mail from the list. SHORThdr means that the header will include only the *essential* fields: *Date:, To:,From:, Subject:,Sender:* and *Reply-to:*. FULLhdr means that all non-essential fields of the mail header will also be sent. IETFhdr is like FULLhdr except that a *Message-id:* field is added (if none was present in the mail sent to the list). The default is SHORThdr.

CONCEAL/NOCONCEAL
: Indicates whether or not you want your name and mail address to appear in the display of list members which is given in response to a REVIEW command. The default is NOCONCEAL.

After having issued a SET command, LISTSERV will confirm to you the successful alteration of your mailing list options. For lists which require validation of all commands, your SET request is forwarded to the list owner. For example, if you want to get a copy of your own messages sent to list EARN-UG at IRLEARN, send the command:

```
SET EARN-UG REPRO
```

to LISTSERV at IRLEARN. If you want to stop receiving mail from all the list at FRORS12 to which you belong, send the command:

```
SET * NOMAIL to LISTSERV at FRORS12.
```

Use the **CONFIRM** command to renew your subscription to a list. Some mailing lists require subscription renewal at regular intervals, usually once a year. A mail message is automatically sent to list members indicating that they must send a CONFIRM command within a given number of days or they will be removed from the list. The syntax of the CONFIRM command is:

CONFIRM	list-name | *

where *list-name* is the name of a distribution list to which you are confirming your subscription. You can confirm your subscription to all the lists you belong to at a particular LISTSERV by using the '*' (asterisk) character in place of a listname. LISTSERV will send a message that your subscription has been confirmed. For example, if you receive a message from LISTSERV@IRLEARN to confirm your subscription to the EARN-UG list, then send the command: CONFIRM EARN-UG to LISTSERV at IRLEARN.

Informational commands:

Use the **HELP** command to get a brief description of the most useful LISTSERV commands and also the names and network addresses os the server's postmaster(s). The format of the command is:

Help

Use the **INFO** command to get a brief description of the most useful LISTSERV commands and also the names and network addresses of the server's postmaster(s). The format of the command is:

Info	< ? | topic > < F= format > < CLASS= class >

where the '?' option will give you a list of topics for for which informational files can be requested. If you require the file in a specific file format or spoolfile class, this can be specified in the *F=* and *CLASS=* keywords

Use the **LIST** command to find out what LISTSERV lists are available.

List	
List	GLOBAL
List	GLOBAL string

Send the command LIST with no parameters to find out what lists are run locally. For example, to find out what lists are available at EARNCC, send mail (or an interactive message) to LISTSERV@EARNCC.BITNET with the word LIST in the message body. Send the command LIST GLOBAL to get a listing of all the lists in all the LISTSERVs throughout the world. This command should be sent to your closest LISTSERV (since all LISTSERVs have the same global list). Note that this file is rather large (over 3500 lines). To get a subset of the global file, send the command: LIST GLOBAL string. You will get a listing of all lists which contain *string* in the list name, host computer, or title.

Use the **QUERY** command to get information on the settings of your personal options for a list. The format of the command is:

Query	listname | *
	File fn ft

where *listname* is the name of a list to which you are subscribed. If you use an '*' (asterisk) instead of the name of a list then you will get information on all lists to which you belong at the LISTSERV where you send the command.

Use the **REVIEW** command to get the parameters of a list and the names and addresses of the list members. For some lists, the names and addresses of the members are not available to non-members.

REVIEW	listname
REVIEW	listname SHORT
REVIEW	listname NOHEADER

If you add the SHORT parameter, you will get only the list parameters without the list members. With the NOHEADER parameter, you will get the list of members without the list parameters.

The **SHOW** command can be used to display a diverse selection of information from a LISTSERV server. This ranges from displaying the names and electronic addresses of the administrators of a LISTSERV server to showing a list of all known LISTSERVs in a given country. The format of the SHOW command is as follows:

SHOW	show-option

where *show-option* can be:

HELP give a brief summary of all SHOW options.

Admin node-list(#) show the names and electronic addresses of the LISTSERV administrator(s) for the nodes given in *node-list*.

ALias node-list show a list of network defined aliases by which the nodes given in *node-list* are also known.

DISTribute get LISTSERV distribution job statistics

INcoming display statistics of incoming files and messages processed by the server.

NOde node-list show information regarding all the nodes given in *node-list*.

TRee(#) node < depth >
display a network tree map from the given *node* to a level of *depth* branches. If *depth* is not given, one level is assumed.

PEer(#) cntry-list show a list of all LISTSERVs for the countries given in *cntry-list*. These may be full names or country codes.

User(#) user-list display all names and electronic addresses that match (or partially match) the userids or real names given in *user-list*.

OWNer(#) list-name
show the name and electronic address of the owner(s) of the lists given in *list-name*.

VErsion show the version and release of the LISTSERV server and its operating system.

File and filelist commands:

Use the **GET** or **SENDME** command to get a file from LISTSERV.

GET filename filetype

SENDme

The names of the files follow the IBM VM/CMS convention: the two parts of the name are each one to eight characters in length, separated by a space. No matter how you send your GET command (e-mail or interactive message), LISTSERV will automatically send you the file in the most appropriate way.

Use the **INDEX** command to review a list of files or packages referenced in a particular filelist. The list is sent onto the requester as a file. The format of the command is:

INDex < filelist > < F= format > < CLASS= class >

where *filelist* specifies the name of the filelist you wish to index. If no name is specified, the index of the root filelist will be sent to you (called LISTSERV FILELIST). This command is functionally equivalent to a GET command with filename and filetype parameters of the filelist name and FILELIST respectively (qqv). If you require that the file or package be shipped in a specific file format or spoolfile class, you can include this information in the *F=* and *CLASS=* keywords.

Parameter description:

*	= all lists
userid	= any valid RFC822 network address not longer than 80 characters If omitted, the node defaults to the command sender's node
full_name	= first_name middle_initial surname (*not* "userid at node")

For each command that causes the server to send you a file, a file format option can also be specified by adding *F=format* to the end of the command. Acceptable formats are:

```
Netdata    Card      Disk        Punch      LPunch      UUe
XXE        HEX(#)    BTOa(#)     VMArc(#)   EBCdic(#)   Split(#)
VMSdump(+)           MIME/text(+)           MIME/Appl(+)
```

LISTSERV Database Facilities:

The archives of mailing lists, as well as certain other databases, can be searched using LISTSERV's database facilities. To find out what databases are accessible at a LISTSERV site send the command:

```
DATABASE LIST
```

To perform a database search, you send mail to LISTSERV containing a **batch command job** with your database query. In addition, EARN/Bitnet users on VM or VMS systems can access the database facilities interactively via the LDBASE program.

For more details on the LISTSERV database facilities, send the command:

```
INFO DATABASE
```

Examples

To subscribe to the EARNEWS list, which is located at the node FRMOP11, if your name is Ada Byron Lovelace, then send the command:

```
SUB EARNEWS Ada Byron Lovelace
```

to: LISTSERV@FRMOP11.BITNET

To get a listing of all distribution lists that have "europe" in the name or title, send the command:

```
LIST GLOBAL EUROPE
```

to your nearest LISTSERV.

To get the file PCPROG ZIP in XXE format the command is:

```
GET PCPROG ZIP F=XXE
```

Learning more about LISTSERV

Detailed documentation on LISTSERV, and related services, is available from: **LISTSERV@EARNCC.BITNET**

For a list of available documents send the command: INDex DOC