LISTSERV User Guide

EARN Association

August 23, 1993

LISTSERV Guide for GENERAL USERS

Notice

We would like to acknowledge the help and guidence given to us by Eric Thomas while writing the LISTSERV User's Guide. Please send any comments you may have about this guide to EARNDOC@EARNCC.BITNET.

Document Version Number : 2.0

Written for Listserv 1.7f

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

Send the following command to the above LISTSERV server: **GET filename**, where the file-name is either:

LSVGUIDE PS	(Postscript)
LSVGUIDE MEMO	(plain text)

ISBN 2-910286-02-9

Contents

The LISTSERV Server

Introduction

LISTSERV was originally devised by the BITNET Information Centre (BITNIC) and was designed to act as a *mailing list server* whose function was to distribute electronic mail (also called e-mail) to users appearing in a mailing list. The server managed any number of these mailing lists, each one addressing a specific area of interest for network users and each having an independent set of list members. For example, para-psychologists who were interested in the topic of near-death-experiences could establish their own mailing list. Those users from across the network who wished to subscribe to this list could then do so by having their e-mail address added to it. The LISTSERV server would subsequently distribute a copy of any e-mail sent to this list to each of the list members. This service provided an extremely convenient means for the exchange of ideas and information between the members of any list since it was LISTSERV (and not individual users) who managed the distribution of e-mail to all final recipients. Users had only to send mail to the e-mail address of a list in order to communicate with any number of list members. This greatly simplified the task of sending e-mail to potentially very large audiences. The concept of mail distribution through a centralized server was termed *mail explosion* since a single piece of e-mail posted to a list resulted in multiple copies subsequently being sent out by the LISTSERV server, one to every list member.

The LISTSERV concept was later adopted and modified by Eric Thomas in a new version of the server called *Revised* LISTSERV. This LISTSERV server retained its primary function as a mail distribution agent and also added many new and powerful features. A major enhancement was the addition of a command processor to the Revised LISTSERV. With the original LISTSERV, all requests for subscription to or removal from mailing lists had to be processed on a user's behalf by the LISTSERV administrator. As the popularity of the server increased, this inevitably led to delays. The new server, however, allowed these and other commands to be sent directly to LISTSERV by the users themselves. This not only reduced the administrative overhead of the server but also ensured that the members of a list would be an up-to-date and interested audience. A more fundamental enhancement to Revised LISTSERV over its predecessor came in its operational design. This was made to accommodate the *decentralization* of the service. Decentralization meant that instead of there being one centralized LISTSERV server controlling all of the mailing lists, many of these servers were located across the network and managed their own, independent set of lists. This improved both the efficiency and robustness of the service by allowing users of LISTSERV to be subscribed to mailing lists managed by a number of different servers, all located at different computer sites. Even so, a high degree of *inter-server cooperation* was maintained which preserved the benefits of a centralized service. For example, a global *sign-me-off-all-lists* command sent to one LISTSERV would be automatically propagated to all the other LISTSERV servers on the network. Likewise, mailing lists at different LISTSERV servers could be *peered* together. Lists such as these would have the same name but each one would have a different membership. Mail sent to any one of these lists would automatically be distributed to the membership at each of its *peers*. This meant that a large mailing list could be split over one or more LISTSERV servers. Using a decentralized topology for LISTSERV also gave rise to the development of powerful mail distribution algorithms that significantly reduced the overall electronic mail traffic load on the EARN/BITNET network and which provided a fast and efficient service to all LISTSERV users.

In addition to its mailing list functions, LISTSERV was enhanced to maintain a database for each mailing list. These *notebook* databases (also called *list archives*) held a copy of every mail message that was distributed on a mailing list. Users could search for and retrieve old mail messages by using the database functions of LISTSERV. Besides notebook and other standard databases, LISTSERV could have any number of databases defined and maintained locally. Another function added to LISTSERV was that of a file server. This meant that the server could be used as a central file repository which could be interrogated and respond to retrieval requests from any user. LISTSERV users could also *subscribe* to files so that they would be informed of file updates or would automatically receive an updated copy of a file. The final function added to LISTSERV was that of an information server. This allowed users to receive up-to-date network information such as details of individual EARN/BITNET computer sites and the overall configuration of the network.

This guide describes how the LISTSERV server may be utilized by any network user. The four major functions of LISTSERV are summarized along with their corresponding commands. These commands are available for everyone and require no special LISTSERV command privileges. Commands which do require special privileges (that is, commands for list owners and LISTSERV administrators) are not included here. For a more detailed introduction to LISTSERV, send the command GET LISTPRES MEMO to your nearest (or any) LISTSERV server.

Sending Commands to LISTSERV

LISTSERV does not need to be on your own computer in order for you to be able to use it. All the servers reside on the EARN/BITNET network and are available to any users that can reach them. Once you are able to do this, you may then send them commands such as signing on or off various lists or any of the other commands described in this guide. There are three ways to communicate with LISTSERV servers: via e-mail, by interactive message and by file transfer. For detailed information about sending electronic mail, interactive messages and file transfers, see the **DOC FILELIST**, available from LISTSERV at node EARNCC. These three methods of communicating with LISTSERV are described below.

By E-mail

This is the primary, most common method of communicating with LISTSERV and can be employed by any user from any network. EARN/BITNET users will be able to send e-mail to LISTSERV servers since they are situated on the same network. Users on other networks must first ensure that they can reach the EARN/BITNET network where the servers reside. Contact your local support personnel if you do not know how to accomplish this. For LISTSERV to be able to read and understand any mail message, it must comply to a mailing standard called RFC822. This is an Internet document that defines the format and syntax of an electronic mail message. Contact your local support personnel to confirm that your e-mail complies to this standard if you are not sure.

Sending commands to LISTSERV via e-mail is not the same as sending mail to a mailing list. Command mail must be sent to the server itself (for example, LISTSERV at DEARN) and should contain a series of LISTSERV commands in its text. On the other hand, a mail message intended for distribution to the members of a mailing list should be sent to the name of that list (for example, the I-KERMIT list at DEARN). When a LISTSERV server receives e-mail, it will assume that it is command mail and will sequentially execute the commands found on each line in the mail. The results of these commands will then be e-mailed back to the sender. The address the server uses to return the command results will normally be the *From:* addressee in the command mail (that is, the original sender of the mail). Note that LISTSERV will ignore the contents of the *Subject:* mail header so make sure that your commands are in the body of the mail message and that you have only one command per line. Any blank lines that may be present in the body of the mail will be ignored. For example, if you wanted LISTSERV at EARNCC to execute the commands **LIST** and **GET DOC FILELIST**, the body of your mail would look like this:

```
LIST
GET DOC FILELIST
```

and would be addressed to the userid LISTSERV at the node EARNCC. Besides returning command results, LISTSERV will also inform you of any unknown commands or syntax errors that were found in the command mail. A number of these errors will be tolerated and processing will be passed to the next command in sequence. However, should the error count pass a certain threshold, processing of the command mail will be stopped and any remaining commands will not be executed.

By Interactive Messages

In addition to sending commands to a LISTSERV server via e-mail, they can also be sent through interactive messages. This method is only available to EARN/BITNET users and not to users from other networks. Interactive messages provide perhaps the most convenient method for communicating with a LISTSERV server. Commands can be sent (one at a time) to LISTSERV which will then execute them and send a response to your screen. Note that for some LISTSERV commands (usually those that generate large quantities of output), results will be sent to you in a file. If this occurs, you will be informed by a message on your screen. Note that commands sent to LISTSERV via interactive messages must be addressed to the server itself and not to any mailing lists.

By File Transfer

For EARN/BITNET users, it is also possible to send commands to LISTSERV via NJE file transfer. It is not possible to use FTP or the file transfer systems of other networks to achieve this. Commands should be entered into a file or dataset using the same rules as described above for e-mail. The file may then be transferred to the LISTSERV server where you want the commands executed. The results will normally be returned to you by e-mail. Note that when you transfer command files to LISTSERV, they must be sent to the server itself and not to any mailing lists.

The Special LISTSERV Address

To facilitate communications with the LISTSERV servers on the EARN/BITNET network, a special LISTSERV computer node has been defined. This means that any EARN/BITNET user can address their commands to the userid LISTSERV at the special node LISTSERV. For example, VM/CMS and VAX/VMS users may send a LIST GLOBAL command to receive details on all LISTSERV mailing lists across the network by issuing the appropriate interactive message command. For VM/CMS users:

```
TELL LISTSERV AT LISTSERV LIST GLOBAL
```

For VAX/VMS users:

```
$ SEND LISTSERV@LISTSERV LIST GLOBAL
```

All non-EARN/BITNET users may address their command mail to the userid LISTSERV at the special node LISTSERV.EARN.NET. Note that if this node is not yet defined on your network, you may also try LISTSERV%LISTSERV.BITNET@CUNYVM.CUNY.EDU. For example, if you wanted to join the 3D-L mailing list that discusses 3D computer graphics but are not sure of the address of the LISTSERV server hosting this list. You could send a subscription request to either of the following addresses:

```
LISTSERV@LISTSERV.EARN.NET
LISTSERV%LISTSERV.BITNET@CUNYVM.CUNY.EDU
```

Your subscription request will be automatically forwarded on to the LISTSERV server hosting the 3D-L mailing list (in this case, at the computer node ARIZVM1).

You may also utilize the special LISTSERV address to send e-mail to any LISTSERV mailing list if you are unsure of its address. For example, if you wanted to send mail to the BITFTP-L mailing list to request a copy of the BITFTP product, you could do so by addressing your e-mail to BITFTP-L@LISTSERV.EARN.NET. It will automatically be forwarded to the list's real address (in this case, BITFTP-L@EARNCC.BITNET) when it reaches the LISTSERV node. The special LISTSERV address is particularly useful if you have signed off a list for a vacation period and you are unsure of its address when you come to resubscribe or if a list has been moved to a new location. Note that any *backbone* LISTSERV server can provide this service and if one is located conveniently to you, it should be used in preference to the the special LISTSERV address. Once a list's address has been established, it should be used for all future mail sent to that list.

LISTSERV Functions and Command Descriptions

LISTSERV can be broadly categorized into four functional areas. These are the mailing list, file server, database and informational functions. Each of these areas is described in detail in separate chapters. Each chapter contains a brief introduction and then goes on to describe the LISTSERV commands relevant to that function. The syntax of each command has been written using a particular format. Command names come first and are presented in upper- and lower-case characters. The upper-case portion of a command shows its valid abbreviation. Either this or the entire command name may be used. The command name is followed by its parameters which are used to supply additional information in the command. All parameters are fully explained in each command description. Those parameters that are surrounded by '< ... >' characters are optional and do not always have to be included with each command (although there will be times when they are needed). The '|' character shows a choice between one or more different parameters. Where it appears, you can select only one of the parameters listed at a time.

LISTSERV Command Keywords

This section describes the set of standard command keywords used in LISTSERV commands. Note, do not confuse these with database keywords, which are explained in the **LISTSERV DATABASE Functions** chapter. When LISTSERV receives any command, it first selects and verifies all the command keywords it contains. These are identified as the text immediately preceding an '=' (equal) symbol in a command. If a standard command keyword is found, it is checked and any errors cause the immediate termination of processing for that command. If a keyword is found but not recognised (that is, it is a non-standard keyword), LISTSERV assumes that it belongs to the overall command and it is passed over. Once standard keyword checking is complete, LISTSERV then checks the overall command for any errors. The standard LISTSERV command keywords are as follows:

PW= password This keyword is used to specify a password in a command. If you have registered a personal password on a LISTSERV server, you will then have to validate certain commands by using the *PW=* command keyword in the command text. This feature is designed to prevent impostors from issuing commands while using your e-mail address. The use of passwords is strongly recommended for this reason. See the **PW** command in the section **Commands for FILE SERVER Functions** for details on registering personal passwords. Once you have registered a personal password at a LISTSERV server, you will be obliged to include the *PW=* command keyword in all commands where it appears as an option.

F= format This keyword controls the file format (or internal file structure) in which files will be sent to you. If you are not a member of the EARN/BITNET network, then LISTSERV will always use the default file format of MAIL. Otherwise, the default file format is dependent upon the information held in the BITEARN NODES file regarding your computer. The BITEARN NODES file is a special network definition file used in the EARN/BITNET network. Any user may ask for a file format other than their default by specifying it in the *F=* command keyword in the commands where it appears as an option. Note that this will not permanently change your default file format, its effect is only for those commands where it is specified. Following is a list of file formats that are valid for all users:

```
XXE  UUe  MIME/text  MIME/Appl  MAIL
```

In addition, EARN/BITNET users may specify:

```
Netdata  Card  Disk  Punch  LPunch  VMSdump
```

Contact your local support personnel for more information about the file formats supported on your computer.

CLASS= class This keyword controls the class of the spoolfile that will be sent to you. Valid file classes are composed of a single alphabetic character. Note that this option is relevant only to EARN/BITNET users but is of little relevance to the everyday LISTSERV user. It is used most commonly by client applications interfacing with LISTSERV to identify LISTSERV command output. This command keyword may be used with any command where it appears as an option.

LISTSERV MAILING LIST Functions

The primary function of LISTSERV is to operate mailing lists (also referred to as distribution lists). Mailing lists are used to distribute the e-mail sent to them to a list a recipients. They provide the means for a group of users to establish an e-mail forum on any topic or area of common interest. For example, quantum physicists who are interested in particle acceleration can request that a mailing list be created for them at one of the LISTSERV servers. This request must be directed to the LISTSERV administrator at a computer site operating a server. Once the list has been established, any user from across the EARN/BITNET network (or from any other network that can reach the EARN/BITNET network) can apply to join it. This may be done by sending a list subscription request to the LISTSERV server managing the list. Their name and e-mail address will subsequently be added to the list membership. LISTSERV will distribute copies of all e-mail sent to a mailing list to all of the members of that list. This service provides an extremely convenient means for the exchange of ideas and information between list members since LISTSERV (and not the mail sender) manages the distribution of e-mail to all of its final recipients. Users need only to remember one list address to which they send their mail in order to communicate to a potentially large number of users. Due to the efficiency with which LISTSERV distributes e-mail to list members, discussions or debates with a world-wide audience may be conducted.

Note that sending mail to a mailing list is not the same as sending mail to a LISTSERV server itself. To distribute mail to a mailing list, it should be sent to that list's address (for example, the Canadian adult education list ADLTED-L at node UREGINA1). Send e-mail to a LISTSERV server only if it contains a series of LISTSERV commands you want that server to execute (for example, command mail sent to LISTSERV at UREGINA1). See the section **Sending Commands to LISTSERV** for more information.

Locating LISTSERV LISTS

A LISTSERV server may manage any number of mailing lists. Each mailing list will have a list name of up to thirty two characters in length. However, most lists have only eight characters in their name and these commonly end with '-L' (to readily identify them as LISTSERV mailing lists). A list's name goes to make up its e-mail address. For example, the Biomechanics and Movement Science list has a list name of BIOMCH-L and is located at the node HEARN. The list therefore has an e-mail address of BIOMCH-L@HEARN, which should be used when sending mail to that list for distribution. The LISTSERV server at node HEARN will manage the list and it is to this server that all user commands related to the list (such as subscription requests) should be sent. Most LISTSERV servers can provide you with up-to-date information on all known mailing lists across the network in response to the following command:

```
LIST GLOBAL
```

You will receive a file called LISTSERV LISTS which contains the names, addresses and titles of all LISTSERV mailing lists. This is a very large file, so make sure you have the necessary resources to accommodate its size before you use this command. You may then browse this file to find the names and addresses of any lists that may interest you. This command is described in full in the section **Command for LISTS**.

Commands for LISTS

The following commands are designed for use with LISTSERV mailing lists. With them you may do things such as finding the names and addresses of lists, signing on to or off lists, reviewing lists or changing your personal options for any mailing list to which you are subscribed. Note that the *F=* and *CLASS=* command keywords may be used optionally with any of the commands where they appear (see the section **LISTSERV Command Keywords** for more information).

SUBscribe	Sign on to a mailing list
SIGNUP	A synonym for the SUBscribe command
UNSubscribe	Sign off a mailing list
SIGNOFF	A synonym for the UNSubscribe command
List	Receive a listing of all the mailing lists at a server
REView	Receive details of a mailing list
Query	Review your optional settings for a mailing list
SET	Change your optional setting for a mailing list
CONFIRM	Confirm your subscription to a mailing list
STats	Display statistical information about a mailing list
REGister	Register your name at a LISTSERV server

SUBscribe Use the SUBscribe (or SIGNUP) command to join a mailing list. You can also use this command to alter the name (but not e-mail address) by which you are known on a mailing list you have already joined. The format of the command is:

SUBscribe list-name < full-name >

Where *list-name* is the name of the list to which you want to subscribe. For example, the EARN User Group list located at the node IRLEARN has a list name of EARN-UG. Do not confuse this with the list's address (EARN-UG@IRLEARN) which should not be used. The optional *full-name* parameter allows you to give a name by which you want to be known on a mailing list. If specified, it should be your full, real name (at least your first name and last name) and not your e-mail address. If you send this command to LISTSERV via command mail, the name for the *full-name* option will be taken from the name given in the *From:* mail header should it be omitted from the command text. For example, the following command would subscribe you to the EARN-UG mailing list under the name of Alan C. Border:

```
SUBSCRIBE  EARN-UG  Alan C. Border
```

Since the EARN-UG list is located at the node IRLEARN, the above command should be sent to the LISTSERV server at the same node.

A request to join a mailing list can be processed in three ways: subscription to a list may 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.

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

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 LISTSERV, you need not include your name in subsequent SUBscribe commands for lists at that server. Unless you do provide your name in the *full-name* option, LISTSERV will search for and use the name it has for you in its *signup* file. 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.

UNSubscribe Use the UNSubscribe (or SIGNOFF) command to leave a mailing list. The format of the command is:

UNSubscribe list-name | * < (NETWIDE >

where *list-name* is the name of a mailing list from which you want to remove your subscription. For example, the MacIntosh Digest list located at the node CEARN has the name INFO-MAC. Do not confuse this with the list's address (INFO-MAC@CEARN) which should not be used. You can signoff all the lists to which you are a member at any particular LISTSERV site by using the '*' (asterisk) character in the place of a list name. If you want your UNSubscribe command to be propagated to all LISTSERV servers on the network, include the *(NETWIDE* option. For example, use the following command to leave the INFO-MAC list at node CEARN:

```
UNSUBSCRIBE INFO-MAC
```

It should be sent to LISTSERV at CEARN which manages the INFO-MAC list. To leave all the LISTSERV lists you belong to throughout the network, send the following command to your nearest (or any) LISTSERV:

```
UNSUBSCRIBE * (NETWIDE
```

Use this option if you are changing your e-mail address or are leaving your computer for an extended period.

List

Use this command to get a listing of available mailing lists at a LISTSERV server. The format of the command is:

List < options > < F= format > < CLASS= class >

Where *options* may be any of the following:

Short - This option displays a summary of all the lists managed by a LISTSERV in a brief, one line description. This is the default.

Long - The Long (or Detailed) option will send a file (called *node-name* LISTS) to you that contains a comprehensive description of the lists managed by a LISTSERV server.

Detailed - This is a synonym for the Long option.

Global < pattern > - This option gives a complete list of all known LISTSERV mailing lists at all servers at the time the command is issued. A file (called LISTSERV LISTS) will be sent to you containing the names, titles and e-mail addresses of these lists. This is a very large file, so make sure you have the necessary disk space to accommodate its size before you use the Global option. The optional *pattern* parameter can be used to match any string in the list name, list title or list address. So if, for example, you wanted details of all mailing lists related to chemistry, you could use the following command:

```
LIST GLOBAL CHEM
```

SUMmary < node | ALL | TOTAL > - This option displays the number of lists managed by a LISTSERV server and the total number of users that are subscribed to them. This will be given for the server to which the command is sent or for the LISTSERV at any node given in the optional *node* parameter. Using the *ALL* parameter will provide the same information but for all LISTSERV servers across the network. The output with this option will be broken down on a server by server basis. The *TOTAL* optional parameter provides similar information except that you are only given an overall summary of the number of mailing lists and list members at all LISTSERVs and are not shown the figures from individual servers.

REView

Use this command to receive a listing of a mailing list. It will be sent to you as a file called *list-name* LIST (or *list-name node-name* for peered lists). A mailing list is comprised of two parts: a control section and a subscription section. The control section holds the definition parameters for a list which includes information such as who is authorized to review or join a list and whether or not it is archived. The subscription section holds the e-mail addresses and names of all list members. The REView command allows you to receive a listing of either or both of these sections (the default is both) for any list, provided you are authorized to do so. Note that at the discretion of the list owner(s), the REView command can be restricted in use to list members only. In this case, you will not be authorized to review a mailing list if you are not a member of that list yourself.

Also, individual list members can restrict the appearance of their e-mail address and name in response to a REView command if they have set the *CONCEAL* mailing list option (see the SET command for more details). The format of the command is:

REView list-name < (> < options >

Where *list-name* is the name of the LISTSERV list you wish to review. The *options* follow the list name and can be:

Short - This option restricts the information you receive to the control section of a list (giving its definition parameters) and does not return the subscription section of a list (giving the list members).

NOHeader - This option is the converse of the Short option. You will receive the listing of list members, but not the list's definition parameters.

Msg - Use this option to have LISTSERV send you the output of your REView command as interactive messages (that will appear on your screen) rather than in a file. Note that this option is valid for EARN/BITNET users only.

Countries - If you use this option, the list of members will be organized by the nationality given in their e-mail addresses. This is similar to using the *BY Country* sort option of the REView command (see below).

LOCal - If the list is peered (that is, it is linked to other mailing lists of the same name but on different LISTSERV servers), you will receive listings of all of these mailing lists in response to a REView command. The LOCal option can be used to suppress the propagation of the REView command to the LISTSERV servers hosting these peered mailing lists. In this case, you will receive a listing only from the server to which you send the REView command and not the others. For example:

```
REVIEW XMAILER LOCAL
```

BY sort-fields - This option is used to specify a presentation for the members of a list you are reviewing. The *sort-fields* parameter indicates the way in which you want the list members to be sorted. If the BY option is not given in a REView command, the list members will be sorted by the computer node name given in their e-mail addresses. This is equivalent to using the the NODEid parameter with the BY option (see below). The *sort-fields* can be:

Country - Sort list members by the country of origin given in their e-mail addresses.

Name - Sort list members by their names (last, then first). Note that this is not the e-mail address but the full name that they provided when they subscribed to a list.

NODEid - Sort list members by their computer node. That is, the name that comes after the '@' (at sign) in their e-mail addresses.

Userid - Sort list members by their userids. That is, the name that comes before the '@' (at sign) in their e-mail addresses.

For example, if you wished to have the membership of a mailing list presented to you sorted by their userids:

```
REVIEW XMAILER LOCAL BY USERID
```

Note that more than one sort field may be specified when using the BY option. If this is the case, then the sort fields given must be enclosed in parentheses and the optional parenthesis that separates the *list-name* from the *options* of the REView command must also be present (although this will not be obligatory in future releases of the LISTSERV server). The list members will then be sorted on all the given sort fields (in order from left to right). For example, if you wanted the list members to be sorted by their countries and then by their names, your REView command would have the following format:

```
REVIEW XMAILER (LOCAL BY (COUNTRY NAME)
```

Query When you join any mailing list, you will be assigned a default set of list options that control such things as way you received mail when it is distributed and the type of notification LISTSERV will give you when it distributes mail you have sent to a list. These are the personal list options that may be altered on any mailing list to which you are subscribed to suite your circumstances. See the SET command for a complete descriptions of these options. The Query command can be used to review the personal list options you have in effect at any mailing list. It has the format:

Query list-name | *

where *list-name* is the name of a list to which you are subscribed. If you use an '*' (asterisk) character instead of a list name, you will receive information about your personal options for all lists to which you belong at the LISTSERV to which you send the command.

SET Use the SET command to change your personal options for a mailing list. These options will remain in effect until they are explicitly changed. The Query command can be used to display your current optional settings for any list to which you are subscribed. The format of the SET 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 list name. After processing your SET command, LISTSERV will send you conformation of the successful alteration of your mailing list options via e-mail. Note that only the options you specify with this command are altered while the rest remain unchanged. The *options* are:

Mail | DIGests | INDex | NOMail - These options of the SET command alter the way in which you receive mail from a mailing list. The Mail option means that you wish to have list mail distributed to you as mail. This is the default. The DIGests and INDex options are available only if a list has had these features enabled by its owner(s). Digests hold all the mail messages sent to a list over a certain period of time. Instead of receiving each mail individually as it is distributed to list members, you will receive all the mail in one batch for a given day, week or month. You may then browse through these mail messages at your leisure. Note that mail is not edited with the DIGests option, you will receive copies of all the mail messages in their entirety. The INDex option will provide you with only the date, time, subject, number of lines and the sender's name and address for all mail messages sent to a list. The text of the mail message will not be included. You may then select and retrieve any mail that interests you from the list archive (see the chapter **LISTSERV DATABASE Functions** for more information on how to do this). Both the DIGests and INDex options provide a means of *listening in* to discussions on mailing lists without having to deal with large quantities of incoming mail messages. The NOMail option means that mail sent to the list will not be distributed to you. This is useful when you are leaving your computer for an extended period of time and you do not want mail from the list to fill your mailbox. When you return, you can send the SET command with the MAIL option to restore mail service. For example, if you want to stop receiving mail from all the lists at SEARN to which you belong, send the following command to LISTSERV at SEARN:

```
SET * NOMAIL
```

Files | NOFiles - Mailing lists may be used to distribute files that have been sent to them via NJE file transfer. This is the file transfer system used by users on EARN/BITNET network. Note that is it not possible to send files to a mailing list via FTP or the file transfer systems of any other networks. Transferring a file to a mailing list can be used when your computer does not have a mail system installed (so you cannot send e-mail) or for sending executable files to mailing lists. When a file is transferred to a mailing list, LISTSERV will distribute it to EARN/BITNET list members in the same format in which it was originally sent and to non-EARN/BITNET list members, as e-mail. This option of the SET command allows you to specify whether or you want to receive any files that were sent to a mailing list via NJE file transfer. The default is Files.

ACK | MSGAck | NOACK - This indicates the kind of acknowledgement you want to receive from a LISTSERV server when it distributes any mail you have sent to a list. ACK means that you want to receive a mail message from LISTSERV indicating that your message was successfully distributed. MSGAck means that you want this confirmation via interactive messages rather than by e-mail. NOACK specifies that you want just a short, one line interactive message from LISTSERV. Note that MSGAck and NOACK acknowledgments are not sent to non-EARN/BITNET users. The default for this option is determined by the owner(s) of a list.

REPro | NOREPro - Indicates whether or not you yourself want to receive a copy of any mail which you send to a list. The default is NOREPro, meaning that mail which you send to a list will be distributed to all members of the list except you. For example, if you want to get a copy of your own mail messages sent to list EARN-UG at IRLEARN, send the following command to LISTSERV at IRLEARN:

```
SET EARN-UG REPRO
```

SHORThdr | FULLhdr | IETFhdr | DUALhdr - All mail messages are comprised of header and body sections. The header section provides details such as the recipients, the original sender and the date ant time a mail message was sent. The mail body section contains the text of a mail message. These options of the SET command indicate the type of mail headers you want to receive in the mail distributed from a mailing list. SHORThdr means that the mail header will include only the essential informational headers (for instance the *Date:*, *To:*, *From:*, *Subject:*, *Sender:* and *Reply-to:* headers). This is the default. You may change this to FULLhdr, which means that all (including non-essential) mail headers will be present in e-mail. The IETFhdr option means that LISTSERV will not change the headers of a mail message it distributes to you other than to add a *Received:* mail header (and also a *Message-id:* and *Sender:* headers if either of these were not already present). This option is designed specifically for compatibility with SMTP exploders (as used, for instance, on the Internet network). Lastly, DUALhdr is very similar to the SHORThdr option except that LISTSERV will also insert mail headers at the beginning of the mail body. Therefore, when the mail is received and read by a recipient using this option, it will start with this information (for instance, the first three lines of a mail message may contain the *To:*, *From:* and *Subject:* mail headers). This option is useful for users of some PC based mail packages that will not display this information from the real mail headers.

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. Note that a complete list of members is always given to list owners and LISTSERV administrators regardless of this option.

TOPICS: < + | - > topic-names - A mailing list to which you are subscribed may support the separation of its mail into a set of topics. If a list owner has defined a variety of different discussion topics for a list, you may use the TOPICS: option to select those that are of interest to you and exclude all others. You will then receive only those mail messages posted to the list that relate to the one or more topics you have chosen. If you do not select any topics for such a list, you will receive all the e-mail sent to it. Note that you must have the MAIL option enabled for a mailing list before you can then use the TOPICS: option since the DIGests and INDex options to not support topic selection and the NOMail option will stop all mail from a list being sent to you (see these options of the SET command for more information). Additionally, a list owner must have enabled topic support for a mailing list before you can use the SET TOPICS: command on that list. This is done at a list owner's discretion. You may use the REView command to determine if topics have been enabled on any particular mailing list.

A mailing list may have up to eleven topics defined for it. You may select one or more of these topics by including their names in the *topic-names* parameter. For example, if a mailing list called NUCHEM-L had the topics neutron, proton and electron defined, you could send the command:

```
SET NUCHEM-L TOPICS: NEUTRON PROTON
```

to receive list mail related to only the neutron and proton topics. Once you have subscribed to various topics, they may be changed by using the '+' (add) or '-' (minus) symbols to increase or decrease the topics you wish to receive from a mailing list. This is useful when you do not want to retype a long list of topic names with the TOPICS: option. For example, to add the topic electron and remove the topic neutron from the NUCHEM-L list:

```
SET NUCHEM-L TOPICS: +ELECTRON -NEUTRON
```

You would now be subscribed to the topics of proton and electron for the NUCHEM-L mailing list. Note that you can review the topics which you have selected for any mailing list through the Query command. You may also use the special topic name of *ALL* to select all the topics available on a mailing list. This is useful when used in conjunction with the '-' (minus) symbol to select all but a few of the topics available on a mailing list. For example, to subscribe to every topic except neutron on the NUCHEM-L list, you could use the command:

```
SET NUCHEM-L TOPICS: ALL -NEUTRON
```

To send mail to one or more topics defined on mailing list, the topic names must be present in the *Subject:* mail header when the mail is posted. If more than one topic name is given, they must be separated by commas. The topic names section should then be finished with a ':' (semi-colon), after which the normal contents of the *Subject* mail header can be entered. For example, to send mail to the users subscribed to the proton and electron topics on the NUCHEM-L list, the *Subject:* mail header would look like this:

```
Subject: proton,electron: Hydrogen Nucleus Results.
```

You may also use the topic name of *ALL* to send mail to all of the available topics on a mailing list. If topic names were incorrect or were not included in the *Subject:* mail header, then this mail is distributed to the special topic name of *OTHER*. You may also subscribe to this topic to receive mail that is distributed under this topic name (note that a subscription to the topic *ALL* will receive mail distributed to the topic *OTHER*).

CONFIRM 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. This command must be sent from the same e-mail address that received the confirmation notice. The syntax of the CONFIRM command is:

CONFIRM list-name

where *list-name* is the name of the mailing list to which you are confirming your subscription. LISTSERV will send a message that your subscription has been confirmed. For example, if you receive a message from LISTSERV at IRLEARN to confirm your subscription to the EARN-UG list, then send the following command to that server:

```
CONFIRM EARN-UG
```

STats This command will cause a statistical report for a specific mailing list to be sent to you. Note, however, that statistics may not be collected for some lists or they may not be available to everybody. The format of the command is:

STats list-name < (LOCal > < F= format > < CLASS= class >

where *list-name* is the name of the list for which you are requesting statistics. The *(LOCal* option can be used to indicate that only local statistics are wanted and not the statistics from other LISTSERVs if the list happens to be peered. You will be sent a file (called *list-name STATREP*) containing the relevant statistics. It will give details such as the name or address and number of mail messages each sender has posted to the list and the network load (in hundreds of bytes) that the list contributes to the network traffic.

REGister The REGister command can be used to register your full name at any LISTSERV server (note that this is your name and not your e-mail address). Once you have registered your name at a LISTSERV, that server will then always refer to you with that name (unless you provide a name in a mailing list subscription request). The command has the format:

REGister full-name | OFF

Where *full-name* is your name. For example, the command:

```
REGISTER Mark A. Taylor
```

sent to any LISTSERV will cause that server to always refer to you with the given name. This can be removed by using the *OFF* option.

LISTSERV FILE SERVER Functions

LISTSERV also functions as a file server. That is, files can be stored at a LISTSERV server and made available for retrieval by users upon their request. These files are stored in LISTSERV under a hierarchical system of *filelists*. As the name suggests, a filelist is a special file that holds a *list of files*. Each entry in a filelist describes a file that is available for retrieval and gives details such as the file's name and size and also a file's access code (also known as a *FAC*) which describes who is authorized to retrieve it. These files may themselves be filelists, in which case they too will hold information about available files or further filelists. In this way, filelists are organized in a hierarchical or tree structure.

There are two basic types of filelists on LISTSERV servers. The first type holds files that have been specifically placed there by file owners or LISTSERV administrators. These files may be documents, maps or diagrams, or even programs. The second type of filelist is associated with a LISTSERV mailing list. These are referred to as *list* filelists and retain copies of all e-mail distributed on a particular mailing list. They consist of a series of files, each of which holds a copy of the mail messages distributed on a list over an interval of time (usually, one month). These files may then be retrieved to recover any list mail distributed during a specific period. Note that e-mail distributed on a mailing list may also be retrieved via the database functions of LISTSERV (see the **LISTSERV DATABASE Functions** chapter for more details). Not all mailing lists have an associated *list* filelist, this is enabled at a list owner's discretion.

A further aspect of the file server functions of LISTSERV is the concept of packages. A package consists of one or more files that appear in a filelist. These may be, for example, a series of files that go to make up a software package. A package is declared in a LISTSERV filelist through a file that has the special name of *package-name* $PACKAGE. Within it will be a list of all the files that comprise the package. This list can be reviewed by sending a retrieval request for that file. However, all the files that comprise the package may also be retrieved through a request for the file: *package-name* PACKAGE. Note that in this case, the '$' (dollar) symbol is dropped from the name. The files that go to make up the package will then automatically be sent to the requestor. This enables users to retrieve an entire set of files that belong to a package with only one command, instead of having to retrieve each of these files individually through a series of commands.

Here is a sample filelist:

```
*  SAMPLE FILELIST for LISTSERV@EARNCC.
*
* ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
*
*  The GET/PUT authorization codes shown with each file entry describe
*  who is authorized to GET or PUT the file:
*
*     ALL = Everybody
*     LMC = LISTSERV Master Coordinators
*     PRV = Private, ie list members
*     OWN = List owners
*:    DAV = 'DAVID@UKACRL'      /* David Boon - Author              */
*
* ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

**********************************************************************
*
* LISTSERV SAMPLE FILELIST
*
**********************************************************************
*                             rec                  last - change
* filename filetype   GET PUT -fm lrecl nrecs   date     time   Info ..
* -------- --------   --- --- --- ----- ----- -------- -------- ------

  LISTFILE MEMO       ALL LMC V      72   609 92/01/31 14:10:27 File ..
  PASCAL-L FILELIST   PRV OWN V      72    49 93/02/15 19:03:45 Pascal.
  PROGRAM  $PACKAGE   ALL DAV V      80    25 90/11/06 11:29:23 Packag.
```

Comment lines begin with an '*' (asterisk) character and are used to provide introductory and explanatory text for the filelist. The most important comments are those explaining the FACs (File Access Codes) in the section entitled 'The GET/PUT authorization codes shown ...'. These FACs are used against the document entries at the bottom of the filelist. In this example, the first four FACs are generic in that they do not refer to specific userids but rather to special groups of predetermined users. All of the FACs specify who is authorized to issue GET (retrieve) and PUT (store) commands for each document. The LISTFILE MEMO file entry shows that everybody

(ALL) is authorized to GET this document while only users who have been designated LISTSERV Master Coordinators (LMC) are allowed to store it. The filelist also shows that this document contains 609 lines, each of up to 72 characters in length.

The PASCAL-L FILELIST entry refers to a sub-filelist. You may go on to retrieve this filelist to investigate what files are held in it. In this case, it is a *list* filelist and is associated with the PASCAL-L mailing list. It would contain a series of files that hold a copy of all the mail messages distributed on that list, each one covering a certain period in time. The GET and PUT FACs specify that only list members (PRV) and list owners (OWN) may retrieve or store this filelist, respectively.

The final document entry is for a package called PROGRAM (as denoted by the file type of $PACKAGE). This file will contain a list of other files in the SAMPLE FILELIST (not shown here) that go to make up the PROGRAM package. Its FAC shows that everybody (ALL) is allowed to retrieve this package but only David Boon (DAV, the program's author) can store it.

Locating LISTSERV Filelists

The *list* filelists that are associated with a LISTSERV mailing list will be located at the server managing that list. If you are searching for this type of filelist, it will be at the LISTSERV located at the same node that appears in the list's e-mail address. For example, the BITFTP mailing list is located at the node EARNCC and has an e-mail address of BITFTP-L@EARNCC. Its filelist will therefore be located at the LISTSERV server at node EARNCC. All *list* filelists will also have the same name as the mailing list with which they are associated. Consequently, the above filelist will have the name BITFTP-L FILELIST.

There are no rules governing the location of files that are not in *list* filelists. Unless you already know the whereabouts of a file, you will have to search for it. However, all LISTSERV filelists are organized from a common starting point. The top (or root) filelist is always called LISTSERV FILELIST and this can be retrieved to determine what files or sub-filelists are available from any particular LISTSERV. This filelist can therefore be the point of departure when searching for any file at a LISTSERV server. Some other standard filelists (which always appear as sub-filelist entries in the LISTSERV FILELIST) include the INFO and TOOLS filelists. These hold LISTSERV's help and informational files and a standard set of software tools, respectively. Other files or filelists may also be present in the LISTSERV FILELIST, which you may then go on to investigate. Note that it is usual that files of a common nature are stored together in a single filelist whose name describes its contents. The information held in each filelist (listing its available files and sub-filelists) may be obtained through the use of the INDex command. For example, to retrieve the LISTSERV FILELIST from LISTSERV at EARNCC and look at the entries it holds, the following command should be sent to that server:

```
INDEX LISTSERV
```

Commands for FILE SERVER Functions

The following LISTSERV commands enable general users to manipulate files that are stored at a server. This includes commands to search for, retrieve and subscribe to files. When sending file server commands to LISTSERV, you must address them to the server and not to any mailing lists. Note that where the *PW=* keyword appears in a command description, this need only be included in the command text if you have defined a personal password at the server to which you are sending file server commands. Otherwise, it need not be used. The optional *F=* and *CLASS=* command keywords may included as desired (see the section **LISTSERV Command Keywords** for more information).

INDex	Receive a listing of a filelist
GET	Retrieve a file from LISTSERV
SENDME	A synonym for the GET command
AFD	Receive a new copy of a file every time it is updated
FUI	Receive notification of file updates
Query File	Receive information on a file
GIVE	Have LISTSERV send a file to another user
PW	Manipulate your personal password

INDex Use the INDex command to get a listing of the files in a particular filelist. The format of the command is:

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

Where *filelist* specifies the name of the filelist you wish to get. If no name is specified, an index of the root filelist (called LISTSERV FILELIST) will be sent to you. For example, to get a listing of the files in the DOC FILELIST, the command:

```
INDEX DOC
```

should be sent to the LISTSERV at node EARNCC where this filelist is located. Note that the above command is the same as issuing a GET DOC FILELIST command.

GET The GET (or SENDME) command is used to retrieve a specific file or package from a filelist, provided you are authorized to do so. It has the format:

GET filename filetype < filelist > < F= format > < CLASS= class >

Where the *filename* and *filetype* identify the file or package you wish to retrieve. The optional *filelist* parameter identifies the filelist within which the file or package resides. If it is not supplied in the command, LISTSERV will determine the filelist through a search of its own internal filelist index. For example, to retrieve all the files that make up the PROGRAM package in the SAMPLE FILELIST (and which are listed in the PROGRAM $PACKAGE file), send the command:

```
GET PROGRAM PACKAGE SAMPLE
```

AFD AFD is an acronym for **Automatic File Distribution**. This command enables you to *subscribe* to a file or package in a filelist. Every time this file or package is updated in its filelist, you will then automatically be sent of copy of it by the LISTSERV server. You may subscribe to any number of files for packages from any filelist with the AFD command. You may also review or delete your subscriptions at any LISTSERV server. The command has the following format:

AFD options

Where *options* must be one of the following:

ADD - The ADD (or REP) option allows you to subscribe to a file or package. This means that a copy will henceforth be sent to you automatically by LISTSERV when it is updated on its filelist. The ADD option has the following format:

ADD filename filetype < filelist > < text > < PW= password >
< F= format > < CLASS= class >

Where *filename* and *filetype* identify a file or package to which you want to subscribe. The *filelist* option allows you to specify the name of the filelist in which it is located. If this is not specified in the command, LISTSERV will locate the filelist through a search of its own internal filelist index. Also, if you wish to have an informational text message inserted automatically at the top of the file or package when it is sent to you, you may use the *text* parameter. Note that if you omit the *filelist* option, the text you provide in the *text* parameter must be enclosed in double quotes. This parameter is most commonly used by network server applications that subscribe to files at a LISTSERV on behalf of users. If you are subscribing to files directly from LISTSERV yourself, you should not need to use this option. For example, if you wanted to subscribe to the file LISTFILE MEMO in the SAMPLE FILELIST, you would send the following command to LISTSERV:

```
AFD ADD LISTFILE MEMO SAMPLE
```

If you wanted the text "New memo update." inserted at the top of the file before it is sent to you, the command would take either of the following formats:

```
AFD ADD LISTFILE MEMO SAMPLE New memo update.
AFD ADD LISTFILE MEMO "New memo update."
```

REP - This is a synonym for the ADD option.

DELete - Remove your subscription to a file or package for which you have an AFD. The DELete option has the following format:

DELete filename filetype < filelist > < PW= password >

Where *filename* and *filetype* identify a file or package to which you wish to remove an AFD. These names may also include the '*' (asterisk) wildcard character so that multiple files or packages may be removed with one DELete command. The *filelist* option allows you to specify the name of the filelist in which it is located. If this is not specified in the command, LISTSERV will locate the filelist through a search of its own internal filelist index.

List - The List (or Query) option shows the files or packages to which you are currently subscribed at a LISTSERV. The List option has the following format:

List < (FORMAT >

A list of your current AFDs at the LISTSERV processing this command will be sent to you. If you include the *(FORMAT* option, then the file format to be used when a file or package is sent to you will also be displayed.

Query - This is a synonym for the List option.

FUI

This command is an acronym for **File Update Information**. It enables you to *subscribe* to a file or package in a filelist. Every time this file for package is updated in its filelist, you will then automatically be informed of this by the LISTSERV server. This command is similar to the AFD command, except you will only be sent notification of an update rather than the updated file or package itself. You may subscribe to any number of files for packages from any filelist with the FUI command. You may also review or delete your subscriptions at any LISTSERV server. It has the following format:

FUI options

Where *options* must be one of the following:

ADD - The ADD (or REP) option allows you to subscribe to a file or package by adding an FUI to it. It has the following format:

ADD filename filetype < filelist > < PW= password >

Where *filename* and *filetype* identify a file or package to which you want to add an FUI subscription. The *filelist* option allows you to specify the name of the file-list in which it is located. If this is not given in the command, LISTSERV will locate the filelist through a search of its own internal filelist index. For example, if you wanted to subscribe to the file LISTFILE MEMO in the SAMPLE FILELIST, you would send the following command to LISTSERV:

```
FUI ADD LISTSERV MEMO SAMPLE
```

REP - This is a synonym for the ADD option.

DELete - Remove your subscription from a file or package for which you have an FUI. The DELete option has the following format:

DELete filename filetype < filelist > < PW= password >

Where *filename* and *filetype* identify a file or package to which you want to delete an FUI. The *filelist* option allows you to specify the name of the filelist in which it is located. If this is not specified in the command, LISTSERV will locate the file-list through a search of its own internal filelist index.

List - The List (or Query) option shows the files or packages for which you currently have an FUI subscription at a LISTSERV server.

Query - This is a synonym for the List option.

Query File This command can be used to get update information on various files. It has the format:

Query File filename filetype < filelist > < (FLags >

Update information about the file or package specified in the *filename* and *filetype* parameters will be given. Optionally, you may include a filelist name in the *filelist* parameter. If it is omitted, LISTSERV will locate the filelist through a search of its own internal filelist index. You may also specify the *(FLags* option to display additional technical data about the file (which can be useful when reporting problems to LISTSERV administrators).

GIVE This command can be used to have a LISTSERV server send a file or package (that you are authorized to retrieve via a GET command) to an e-mail address other than your own. It has the format:

GIVE filename filetype < filelist > < TO > userid@node
< F= format > < CLASS= class >

Where the *filename* and *filetype* identify the file or package which you wish LISTSERV to send to the specified address. The optional *filelist* parameter identifies the filelist within which the file or package is held. If it is not supplied in the command, LISTSERV will determine the filelist through a search of its own internal filelist index. The *userid@node* parameter is required and gives the e-mail address of the intended recipient. For example, the following command would direct LISTSERV send the LISTFILE MEMO document from the SAMPLE FILELIST to the user *JACQUES* at the node *FRMOP11*:

```
GIVE LISTFILE MEMO SAMPLE TO JACQUES@FRMOP11
```

PW The PW command enables you to add, change or delete a personal password on any LISTSERV server. A personal password is designed to give you added command security on a LISTSERV since it helps prevent impostors using your e-mail address. The use of personal passwords is strongly encouraged for this reason. Every server will accept a password registration request from you at any time. Passwords are comprised of one to eight alphanumeric characters. You may change or delete your password at any time. If you forget a password you have registered at any LISTSERV server, contact that server's postmaster (whose e-mail address is given in the RELEASE command; see the section **Commands for INFORMATION**). The format of the PW command is:

PW options

Where *options* must be one of the following:

ADD - Add a new personal password on the LISTSERV processing your command. The ADD option has the format:

ADD new-password

Once you have registered a password on a LISTSERV server, you will be obliged to use the *PW=* command keyword in the commands where it appears as an option.

CHange - Change your personal password on a LISTSERV server where you already have one. It has the format:

CHange old-password new-password

or

new-password PW= old-password

DELete - Remove your personal password from a LISTSERV where you already have one. The DELete option has the format:

DELete old-password

Once you have removed a password from a LISTSERV server, you will no longer be obliged to use the *PW=* command keyword in the commands where it appears as an option.

LISTSERV DATABASE Functions

LISTSERV provides the facility for users to retrieve old mail that has been distributed on mailing lists. Each mailing list has an associated database (called a *notebook* or *list archive* database) in which list mail is stored. Note that databases may not be maintained for every mailing list, this is done at the discretion of a list owner. The notebook databases are the most commonly used of the LISTSERV databases. However, every LISTSERV server also has a database of all the EARN/BITNET computer nodes (called the BITEARN database). This is available to all LISTSERV users. The *backbone* LISTSERV servers also have a database of all the LISTSERV computer nodes (called the PEERS database). In addition to these databases, a LISTSERV server may have any number of different databases that have been created locally.

DATABASE Keywords

The documents in a database all have specific attributes in common. These attributes are exploited by defining a set of keywords for that database. A database keyword is something like a luggage tag that is attached to each document in a database. Any number of tags may be attached to a document, each one containing information about that document (for example, the date and time it was stored in the database or the author's name and e-mail address). These keywords can then be used when searching for documents, or the information they contain may be displayed. The keywords used by each database may differ, but there are certain keywords common to all of the databases maintained by LISTSERV. These are:

DATABASE The name of a database (one to eight characters)

DATE The date of the document's entry into a database

TIME The time of the document's entry into a database

#RECS The number of lines in a document

The document number in a database

Notebook Database Keywords

The most common of the LISTSERV databases are the *notebook* or *list archive* databases. Each of these databases is associated with a mailing list and holds copies of all the mail messages distributed on that list. These notebook databases have the following additional keywords available for use when searching for or displaying information from a document:

Subject Set to the contents of the *Subject:* mail header in a mail message.

SEnder Set to the e-mail address of the sender of a mail message

The notebook databases also support the division of each document into predefined segments or sections. The keywords associated with each document section can be used to display information from either or both of these sections. They are:

Header The section of the document that contains the headers of a mail message (which include, for example, the *From:* and *To:* mail headers).

Body | Text The section of the document that contains the text of a mail message.

All Both the Header and Body sections of mail message. This is default.

Note that not all databases have the same additional keywords available for use, nor are all documents split into sections. To be able to use keywords effectively when searching for or displaying information from documents, you should know what keywords and document sections are available for each different database.

Locating LISTSERV DATABASES

Each LISTSERV maintains its own independent set of databases and it will be necessary for you to first locate the server that holds the database that interests you. To determine what databases are available at a LISTSERV server, use either the DATABASE or DBase command. LISTSERV will return the names of all the databases it maintains along with a brief, one line description of each. The command format is:

DataBase LIST

When you are searching for the notebook database of a mailing list, it will always be located at the LISTSERV server that manages that mailing list. Therefore, the address of a mailing list (for example, IBMPC-L@CEARN) shows which LISTSERV will maintain that list's notebook database (in this case, LISTSERV at node CEARN). Once you have located a database you wish to search, you may access it either through a **batch** or an **interactive** database session.

Batch DATABASE Access

Users from any network that can reach the LISTSERV servers may send database commands to a server through the *batch* facility. Users on the EARN/BITNET network where the LISTSERV servers are located may also use the *interactive* database facility (see below). The *batch* facility entails sending e-mail to a server that contains a *database job*. Note that e-mail should, in this case, be sent to the LISTSERV server where you want the job to be executed and not addressed to any mailing lists. A *database job* consists of a number of database commands that are formatted in a special way. These commands go into the body of the mail (one per line) and are executed sequentially by LISTSERV. Note also that LISTSERV will ignore the contents of the mail's *Subject:* header, so ensure the commands are in the body of the mail message. The results of these commands will be returned to you by mail once all the commands have been executed. If the *database job* contains unknown commands or syntax errors, these will be ignored and processing will continue with the next command in sequence. However, if the error count passes a certain threshold, processing of the *database job* will be stopped and any remaining commands ignored. LISTSERV will notify you of any errors that are encountered in a *database job*. The following sample format of a *database job* can be used for any database commands:

```
// JOB
DATABASE SEARCH DD=RULES
//RULES DD *
command1
command2
...
/*
// EOJ
```

In the above example, the first line starts the *database job* and any line before this will be ignored (as will blank lines found anywhere in the mail message). The next line specifies that this is a *database job* and gives the name of the section that holds the database commands (this comes after the *DD=* keyword). In the example, the command section is called *RULES*, but any name of any length can be used instead. Following this, the command section starts, signaled by its name and the *DD ** string. From this point until the line that ends the command section, any number of LISTSERV database commands may be entered. The last two lines signal the end of the command section and the end of the *database job*, respectively. If any lines of text are found in the e-mail after the line that finishes a *database job*, LISTSERV will execute these as non-database LISTSERV commands (see the section **Sending Commands to LISTSERV**). This commonly occurs by mistake when signatures are automatically put in to a mail message when it is sent. If you wish to avoid this and the accompanying invalid command notification that LISTSERV will send to you, ensure that such features of your mailing system are disabled prior to sending *database jobs* to LISTSERV servers. For example, a *database job* to search for Mathematical Institutions in the BITEARN database would be:

```
// JOB
DATABASE SEARCH DD=FINDMATH
//FINDMATH DD *
SEARCH MATH IN BITEARN
INDEX
/*
// EOJ
```

Where the commands that will be executed are SEARCH and INDEX (these commands are explained in detail in the **DATABASE Commands** section). After having posted this *database job* to any LISTSERV (since the BITEARN database is resident at all of them), you would expect to receive a response similar to the following:

```
> SEARCH MATH IN BITEARN
--> Database BITEARN, 47 hits.

> INDEX
Ref# Conn  Nodeid   Site
---- ----  ------   ----
0156 85/08 BENGUS   Ben-Gurion University Mathematic...
....
```

For more information on **batch** access to LISTSERV databases, send the command GET LISTDB MEMO to your nearest (or any) LISTSERV. For more information on the *Command Job Language* of LISTSERV (used to construct *database jobs*), send the command GET LISTJOB MEMO to your nearest (or any) LISTSERV.

Interactive DATABASE Access

As well as using **batch** access to LISTSERV databases, EARN/BITNET users that work on VM/CMS or VAX/VMS systems can use a special user interface program (called LDBASE) for **interactive** database access. That is, they have the ability to open a database session with a LISTSERV server during which commands and their results are passed between the user and the server by means of interactive messages. The LDBASE interface is the *client* program that establishes and controls an interactive database session between the user and a LISTSERV server. If this program is not already available at your site, it can be retrieved from any LISTSERV server. For VM/CMS users, send the following commands to your nearest LISTSERV:

```
TELL LISTSERV AT node-name GET LDBASE EXEC
TELL LISTSERV AT node-name GET LSVIUCV MODULE
```

where *node-name* is the location of your nearest (or any) LISTSERV server. The LDBASE command will be ready for use as soon as you receive these two files onto a minidisk.

For VAX/VMS users, send the following command to your nearest LISTSERV:

```
$ SEND LISTSERV@node-name GET LDBASE.COM
```

where *node-name* is the location of your nearest (or any) LISTSERV server. The command to install the LDBASE program in your directory is:

```
$ @LDBASE
```

For VM/CMS and VAX/VMS systems, these two programs have the same function: once an interactive session is requested, they attempt to establish a network connection to the LISTSERV holding the databases you wish to access. The format of the command is:

LDBASE < node-name >

Where *node-name* specifies the location of the LISTSERV server with which you wish to commence an interactive session. The first time you use this command without specifying the *node-name* parameter, you will be asked to supply a node name you wish to have as your *default* node. From then on, omitting a node name with this command will automatically give you a session with the LISTSERV server at the location you have given. Otherwise, you will be given an interactive session with the LISTSERV located at the node you specified with the command. The LDBASE command will inform you when an interactive session has been successfully established. From this point, you may enter and re-enter any number of database commands until you terminate the session with the QUIT command.

Some database commands can result in large amounts of output (for example the Print command). To reduce the amount of information passed over the network by such commands, output has been restricted in the following way: if you have established an interactive database session with a LISTSERV server that is not located on the same computer as you, command output is limited to the first thirty lines. You will be informed if any output to you has been suppressed. If this is the case, you can retrieve the complete command output by using the SENDback command or by constructing a *database job* where command output will be e-mailed back to you.

DATABASE Commands

These commands are designed to give the LISTSERV user access to any databases for which they have appropriate authority. With them, databases may be searched and documents displayed or retrieved. Note that some of the commands you can construct may become too long to fit onto one line. If this is the case, commands may be split over as many lines as desired by using the '-' (hyphen) continuation character. To use this, end each line with this character and continue entering your command on the following line. LISTSERV will not execute the command until you enter the last line of the command (which should not end with a continuation character).

Search	Search a for documents holding a given text string
SELect	A synonym for the Search command
Index	Display the list of documents selected in a Search
Print	Display the contents of documents selected in a Search
SENDback	Send yourself a copy of one or more selected documents
Format	Change the format and data displayed by an Index command
List	Display data from selected documents in a given format
QUIT	End an interactive database session

Search The Search (or SELect) command is used to search through the documents in a specified database for a given string of text. The text may be one word, a series of different words or a phrase. You may also include a number of optional search criteria with which a search can be restricted. Once a search has been initiated with this command, you will be informed of the number of documents that match your search criteria (these are called the number of *hits*). These documents then form the group on which subsequent database commands are executed. The Search command should be the first database command you use when you start a database session. You must have a number of hits before issuing any of the other database commands. The format of the Search command is:

Search search-rules < optional-rules >

search-rules

The *search-rules* specify the text string to be searched for in the documents within a database . If you are interested in making your selection of documents based solely on the other *optional-rules* (described below) then you may give a *search-rule* value of '*' (asterisk), otherwise, a text string must be provided here. A text string may be comprised of one or more words separated by logical operators or it may be a phrase (in which case it must be quoted). If a text string is given as a series of words that are not quoted and no logical operator separates them, the logical operator AND is assumed. For example, the following commands are identical:

```
SEARCH PC VIRUS
SEARCH PC AND VIRUS
```

These commands will result in a search for documents that hold the words *PC* and *VIRUS* anywhere in their text. Note that both of these words must be present in a document for it to be selected (as denoted by the logical operator AND). A full list of logical operators is given in the **keyword-rules** section below. A phrase is constructed by the use of quotes in the text string. Phrases may also be separated by logical operators. For example:

```
SEARCH 'PC VIRUS' OR 'Virus Warning'
```

results in a search for documents that hold the phrases *PC VIRUS* or *Virus Warning* in their text. When using quotes to specify a phrase, ensure that the correct number of blank characters are present between each word in the phrase. Searching for *PC VIRUS* with one blank character between *PC* and *VIRUS* will not match strings that have multiple blank characters in the same position. You may use either single or double quotes to enclose a phrase. However, there is an important difference between them. Using double quotes will result in a case-sensitive search for the text in a phrase while single quotes (or no quotes at all) yields a non-case-sensitive search. If a phrase itself contains a quotation mark in it, then it must be doubled:

```
SEARCH 'The Forum''s decision'
```

optional-rules

The *optional-rules* part of the Search command is comprised of a *database-list*, *date-rules* and *keyword-rules*. One or more of these elements may be included in a Search command. They may appear, in any order, after the search-rules have been specified and may be identified by the appearance of their reserved-words in the Search command text. Each of these *optional-rules* are described below under their own headings.

database-list

A *database-list* identifies the database whose documents are to be included in the search. It must be present in a Search command until a number of documents have been selected as *hits*. Once you have a group of hits, the *database-list* can be omitted and all subsequent Search commands will act solely upon this selected group of documents (including additional Search commands which do not specify a new database. This can be used to further reduce the number of selected documents). If you wish to search for new documents not already among the group of hits, a new *database-list* should again be included in the Search command. It has the following format:

IN db-name

Here, the database name that you specify must be preceded by the reserved-word **IN**. Therefore, if you wanted to search for the phrase 'Infinite Loop' in a database called BUGS, you would form your search command in the following way:

```
SEARCH 'Infinite Loop' IN BUGS
```

Within a database, each document is assigned a unique, sequential number. You will see the number assigned to each document in response to an Index command (which lists all the documents selected). This number may change between different database sessions, however, it will not change during a session. Therefore, they may also be used to further narrow a search for documents within a database by specifying a list of one or more document numbers, an upper or lower limit for document numbers or a range of numbers. This information is appended to the given database name with the following format:

db-name. < (> number <,> number- <,> number-number <,> -number <) >

All document numbers are appended to the database name by a '.' (full stop) character. If you are going to specify more than one document number, they may either be enclosed in parentheses or else they must be separated by commas. For example, in a database called *PROJECTS*, you could include in your search all the documents with numbers up to and including 10, document number 15, all those between and including numbers 20 and 30 and all documents above and including number 45 with either of the following commands:

```
SEARCH * IN PROJECTS.(-10 15 20-30 45-)
SEARCH * IN PROJECTS.-10,15,20-30,45-
```

Including document numbers in a Search command is useful only when reducing the quantity of hits from a previous database search.

date-rules

The *date-rules* option can be used to further restrict the search of documents to those that fall within a given time interval. The reserved-words in the Search command that indicate the presence of *date-rules* are:

SINCE date < time >

FROM date < time > TO date < time >

UNTIL date < time >

Here, the optional *time* parameter may be supplied in the form of hh:mm<:ss>; where *hh* represents the hour, *mm* the minutes and the optional *ss*, for the seconds. The *date* parameter can be supplied in any of the following formats:

TODAY
yy
yy/mm/dd
yy-mm-dd
mm/yy
mm-yy
dd mm
< dd- > month-name < -yy >

Where *yy* is a two-digit representation of the year, *mm* is a two-digit month number and *dd* is a two-digit day number. The special string *TODAY* inserts today's date into the date-rule clause. The *month-name* parameter can be used to provide the name of a month. This can be abbreviated to any length and the first month (in chronological order) that matches the *month-name* supplied will be selected for use.

keyword-rules

The *keyword-rules* use a predefined set of keywords for the database being searched. See the section **DATABASE Keywords** for a description of keywords specific to notebook databases that can be used in database searches. The format of keyword-rules is:

WHERE keyword-tag expression

Here, WHERE is the reserved-word that denotes the presence of a keyword-rule in a search command. (Note that WITH is a synonym for WHERE and they can be used interchangeably). The *keyword-tag* identifies a valid keyword for the database where the search is to be done. Note that these can differ between different types of databases. In the case of notebook databases, this includes the Subject and SEnder keywords. The *expression* that follows the *keyword-tag* must be satisfied if any document is to be counted as a hit. These *expressions* may be written in the following format:

IS value
= value
IS NOT value
ˆ= value
> value

\>= value
< value
<= value
CONTAINS value
DOES NOT CONTAIN value
SOUNDS LIKE value
DOES NOT SOUND LIKE value

Where the *value* given is either a number or a string of text. All text strings that contain more than one word must be enclosed in quotes. If no comparison operator is put before the *value* then *IS* is assumed. The above *expressions* are all self-explanatory with the exception of *SOUNDS LIKE* and *DOES NOT SOUND LIKE*. These are used to make phonetic searches on keywords where the *value* you supply in the *expression* does not have to match exactly to the information in the *keyword-tag*. Some examples:

```
SEARCH * IN PSYCH-L WITH SENDER 'gregson@une.nsw.au'
SEARCH * IN PROG-L WHERE SUBJECT CONTAINS 'Binary Tree'
SEARCH * IN EARN-UG WHERE SENDER SOUNDS LIKE 'Smith'
```

In the first example, a search will be initiated in the *PSYCH-L* notebook database for all mail messages that have a SEnder keyword value of *gregson@une.nsw.au*. The second will search the *PROG-L* notebook database for all e-mail whose Subject keyword includes the string *Binary Tree*. The last example will select mail messages that have been sent by anyone to the *EARN-UG* mailing list whose userid could be *smith, smythe* or even *Smithers*. It is also possible to construct complex expressions by making use of logical (or boolean) operators. These operators can also be used in the *search-rules* section of a Search command. They are:

NOT or ^
AND or BUT or &
OR or | or /

Note that when you use these logical operators, it is advisable that you enclose the expression in parentheses. For example:

```
... IN PHYSICS WHERE SUBJECT CONTAINS (Nuclear OR Particle)
```

Index This command is used to display information from all of the documents which have been selected with a Search command. This information will help you decide what to do next (for example, issue a new Search command or perhaps review the contents of one or more of the documents with a Print command). Information from each of the selected documents will be displayed in a predefined format (depending on the type of database you have searched). For every document, you can expect to be shown details such as its number, its size, the date it was created or last modified and any other information relevant to the type of database you are searching. Each type of database has its own predefined display format when the Index command is issued. The information shown and the way it is displayed can be customized through the use of the Format command. The format of the Index command is:

Index < fmt-name >

Where *fmt-name* is the optional name of a display format you have previously defined with a Format command. If you do not specify a *fmt-name* parameter, the default display format (which has the format name of *INDEX*) will be used.

Print This command can be used to review the contents of one or more documents that have been selected through a Search command. It is therefore designed to be used when you have narrowed down the number of selected documents (perhaps through successive Search commands) to those that are definitely of interest to you since this command can generate a lot of output. The Print command has the following format:

Print < options >

Where *options* provides a list of one or more documents whose contents you wish to have displayed. If you do not identify any specific documents by omitting the *options* parameter, you will be shown the complete contents (or the first thirty lines) of all the documents that you currently have selected. If you do wish to provide *options* with the Print command, they have the format:

< sections keyword-tags > < OF > < document-numbers > <, ... >

Here, the *sections* and *keyword-tags* parameters are used to identify which sections or keywords from the selected documents you wish to have displayed. This can be a list of one or more document sections or keywords valid for the database you have been searching. A description of document sections and keywords available for notebook databases is given in the **DATABASE Keywords** section. The *document-numbers* parameter may then be included to identify those specific documents you wish to have displayed. The numbers of all currently selected documents may be seen with the Index command. If no document numbers are given, the Print command acts on all of the currently selected documents. Otherwise, you may provide a list of one or more document numbers, an upper or lower limit of document numbers or a range of numbers. These parameters to the Print command may be given multiple times by separating them with a ',' (comma) character. Consider the following example for a notebook database:

```
PRINT SUBJECT OF -10, SENDER OF 13-15, 19 21, BODY OF 30-
```

This Print command gives four separate instructions for displaying information from the documents that have already been selected with a Search command. The first set of parameters asks that the information held in the Subject keyword for all documents whose numbers are up to and including 10 be displayed. The second set of parameters, that the information from the SEnder keyword for all documents in the range of 13 to 15 be displayed. The next set of parameters specifies that the entire contents (that is, both the Header and Body document sections) of documents 19 and 21 be displayed. Finally, the last set of parameters asks that the Body document sections of all documents with numbers greater than and including 30 are shown. Therefore, if a Search command has selected a group of documents whose numbers are 9, 10, 12, 13, 14, 15, 19, 20, 21, 29, 30 and 31, then the above Print command will display information regarding all of these documents except numbers 12, 20 and 29. Note that in interactive mode, the results of a Print command are sent directly to your screen. If you wish to retrieve the results of a Print command as a file, then you must use the SENDback command.

SENDback This command is used to have command output returned to you in a file. Its use is valid only if you have established an interactive database session with a LISTSERV server. If you are using the batch access method, you will always receive the output of any database commands as e-mail. The SENDback command has the format:

SENDback command

where *command* can be either a valid Print or Index command. You will receive the output requested as a file called *database-name* OUTPUT. For example:

```
SENDBACK INDEX
SENDBACK PRINT SENDER BODY OF 23-26
```

Format This command may be used to change the way information about the documents you have selected is displayed to you with the Index command. The Index command will use a default display format unless a new format is defined and specified as an Index parameter. The Format command can alter the selection, placement, length and headings of the fields that will be presented. Once a display format is defined with the Format command, it can be used repeatedly for the duration of the database session. The Format command syntax is:

Format fmt-name: field1 < field2 > < ..... >

Where *fmt-name* is the name you have chosen for the format you are about to define. You will use this name as a parameter of the Index when you wish to use it. The format name must be followed by a ':' (colon) character. The *field* parameters then go on to define the presentation of the information available from a database. Each *field* consists of a database keyword and a description of how the information from that keyword is to be displayed. See the section **DATABASE Keywords** for a description of keywords common to all LISTSERV databases and those specific to notebook databases that can be used in the Format command. Each *field* parameter has the following format:

keyword < (from <,to>) >< .width <just> > < "heading" <just> >

Here the *keyword* parameter identifies a database keyword whose information you wish to put in a display field. Immediately following this, you may optionally include the *from* and *to* parameters (in parentheses). They control the number and position of the characters passed from the database keyword to the display field you are now defining. For example, you want to define a display field for the database name. This information comes from the DATABASE keyword, however, you only want to use characters 4 to 7 from this keyword. This field definition would then start with DATABASE(4,7). If the name of database is *SU-ATOMS*, only the *ATOM* string will then be passed from the keyword to the display field. The default values for *from* and *to* are 1 and 255, respectively. Usually, you will not need to use these parameters. Following this is the section defining the *width* and optional new *heading* for the display field. It must be be preceded by a '.' (full-stop) character. The *width* is the number of characters you wish to use for displaying the keyword information for this field. The *heading* parameter defines a new title for the display field (the default is the name of the keyword being referenced). If a new title is given here, it must be enclosed in double quotes. To continue the above example, you want to display the database name in a field of 4 characters under a heading of "Name". The field definition would now look like: DATABASE(4,7).4 "Name". When the format is used with an Index command, the result would appear as:

```
Name
----
ATOM
```

If the information passed from the database keyword to the display field is longer than the *width*, it will be truncated and a '+' (plus) character appended to the name when it is displayed. This is true for any display field. In the above example, if the field definition was DATABASE.4 "Name" then the result would be:

```
Name
----
SU-A+
```

Note that because the *from* and *to* parameters were not used in the above example, the full database name was passed from the keyword to the display field. You may also justify information in the display by using the *just* parameter. The *just* parameter has the format:

L Left Justified

R Right Justified

C Centered

R0 Right Justified with leading zeros

For example, you want to define your own display format and include in it the database name and the document number. The database name should be in a field of 8 characters and have the title "DB Names" while the document number, a field of 6 characters and have the title "Doc#". Both of the titles should be centered while the actual database name should be displayed left justified and the document number, right justified and packed with leading zeros. The format definition would look like:

```
FORMAT MYFMT: DATABASE.8L "DB Names"C #.6R0 "Doc#"C
```

The DATABASE and # keywords provide the database name and document number information. The result of an Index MYFMT command would be:

```
DB Names  Doc#
--------  ----
SU-ATOMS 000103
CHEM-L   000089
```

Note that the *R0* justification parameter can be used only for fields that hold numeric information and never for heading definitions.

List This command can be used to display information on the documents you have selected (through a Search command) in a temporary format you define in the List command parameters. It is similar in function to an Index command that is called with the name of a display format you have previously defined in a Format command. However, it differs in the sense that the display format you choose with the List command is temporary and must be redefined every time it is used (the Format command saves the definition under a given name which may be used repeatedly). The List command is most useful for displaying information about selected documents that would not normally be shown with an Index command and without having to first define a whole new display with the Format command. The format of the List command is:

List format

Where *format* can be a series of parameters describing the display fields to be used (identical to those used in the Format command), or the name of a previously defined display format (either one you have given in a Format command or the default display name of *INDEX*). Alternatively, it can be a combination of the above. This makes the List command very flexible. For example, you have a number of documents selected and now wish to be shown a list of their senders. If the document sender is not given in the default display of the Index command, the following List command will supply this information:

```
LIST SENDER.10 "Written by"
```

The senders of all selected documents will be displayed in a field of 10 characters under the title of "Written by". However, only the document senders will be displayed for each selected document, and no other information. You may, at this point, decide to continue defining each consecutive field for the display (as in the Format command) or display the sender information in conjunction with another, already defined, format:

```
LIST SENDER.10 "Written by" fmt-name
```

Where *fmt-name* is the name you have assigned to a display format previously defined with a Format command or is the default name *INDEX* (which gives the standard information for the type of database you are searching). Using these parameters, the List command will display each document sender followed by the information normally displayed with the *fmt-name* format.

QUIT This command is used to leave an interactive database session. It has no meaning in a batch database session.

LISTSERV INFORMATION Functions

The LISTSERV server can provide a diverse range of information to the general user. This includes help files, release levels of the server and important configuration files, statistics and information pertaining to the EARN/BITNET network. When sending these commands to LISTSERV, they must be addressed to the server and not to any mailing lists it may manage.

Commands for INFORMATION

The following LISTSERV commands enable general users to request information from servers. For those commands that result in files being sent to the requestor (for example the Info command), the format and class of the file can be specified by the optional command keywords *F=* and *CLASS=* in the command text (see the **LISTSERV Command Keywords** section for more information).

Help	Receive a brief summary of commands
Info	Retrieve a help or informational file from LISTSERV
RELEASE	Displays information about a LISTSERV server
SHOW	Displays diverse information

Help Use this command to get a brief description of the most commonly used LISTSERV commands and also the name and e-mail address of the server's postmaster. The format of the command is:

Help

Info Use this command to get an informational (or help) file from a LISTSERV server. The format of the command is:

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

Where the *topic* option should specify the topic for which an informational file will be sent to you. You may get a list of valid topics by sending the Info command with no parameters to your nearest (or any) LISTSERV.

RELEASE The RELEASE command may be used to display information about a LISTSERV server. This includes its version, release and fix levels; the date and time stamps for important configuration files; and the name and e-mail address of the server's postmaster. It has the format:

RELEASE

SHOW Use the SHOW command to display a variety of information about a LISTSERV server and the EARN/BITNET network. The command format is:

SHOW show-option

Where *show-option* must be one of the following:

NADs node-list - Display the e-mail addresses of the registered node administrators for one or more computer nodes given in the *node-list* parameter. This information is taken from the BITEARN NODES file.

ALIAS node-list - Show the different names by which one or more nodes (given in the *node-list* parameter) are known. This is useful for obtaining the domain style address of a node (or vice versa).

VERSion - This option is a synonym for the RELEASE command.

FIXes - Display information about the LISTSERV program fixes installed on a server.

BITEARN - Display technical information about the BITEARN NODES (network definition) file.

NETwork - Display general information about the BITEARN NODES (network definition) file.

NODEntry node-list - Displays the information held in the BITEARN NODES file about the one or more EARN/BITNET computer nodes given in the *node-list* parameter.

LINKs node-list - Display information about all the EARN/BITNET network links defined for one or more computer nodes given in the *node-list* parameter. This information is taken from the BITEARN NODES file.

PATHs s-node node-list - Show the network path taken between one computer node (given in the *s-node* parameter) and one or more different nodes (given in the *node-list* parameter). You will be given a diagram showing each of the computer nodes on the path and a speed (or link resistance) value associated with each of these links.

DPATHs node-list | * - Show the path chosen by one LISTSERV server when delivering mail via the DISTRIBUTE algorithm to one or more nodes given in the *node-list* parameter. You will be shown a list of (backbone) LISTSERV servers that will be used for this distribution. If you use the '*' (asterisk) character in place of a *node-list* parameter, the distribution paths to all other known LISTSERV servers will be given. This output is extremely long, so if you issue this command via interactive messages, you should also use the MAIL prefix if you are sending commands interactively (see the **Trouble-Shooting** section for more information).

DISTribute - Show statistics of outgoing and incoming DISTRIBUTE jobs processed by a LISTSERV server.

Trouble-Shooting

Batch Jobs When a LISTSERV server receives command mail (that is, LISTSERV commands sent to a server via e-mail) it will execute the commands and mail the results back to the sender. See the section **Sending Commands to LISTSERV** for more details on command mail. LISTSERV will tolerate only a certain number of unknown commands or syntax errors from a command mail. If this threshold is reached, processing of the command mail is stopped and any remaining commands will not be executed. There may be cases when users sending commands to LISTSERV via e-mail will not have any commands executed due to an excess of errors found in the mail before any commands are reached. This can happen when mailing systems automatically insert text at the top of mail messages as they are posted and provide no means for disabling this process. This text is read by the LISTSERV server and each line is counted as an error until the threshold is reached and processing for the command mail is stopped. To deal with this and similar situations, LISTSERV has its own *Command Job Language* whereby commands to be executed can be presented in a special format. Commands sent to LISTSERV in this manner are known as *batch jobs*. For example, if you wanted to send the SHOW VERSION and LIST commands for execution on the LISTSERV at SEARN, you could construct the following *batch job*:

```
// JOB
SHOW VERSION
LIST
// EOJ
```

and then send it to that server. The first line in the above example indicates the start of a *batch job*. LISTSERV will recognise this and totally ignore any lines in the body of the mail message that precede it. Following this, any number of LISTSERV commands may be entered (one per line) and these will be processed normally. A *batch job* is ended by the appearance of the last line (as shown in the example above). For more information about the LISTSERV *Command Job Language*, send the command GET LISTJOB MEMO to your nearest (or any) LISTSERV server.

The MAIL Prefix

There may be times when you send a command to a LISTSERV server and wish to have the command output returned to you via e-mail. This will always be the case for users not on the EARN/BITNET network since they can only communicate with LISTSERV servers with command mail (see the section **Sending Commands to LISTSERV** for more information). EARN/BITNET users, however, may send commands to a server via interactive messages. In this case, LISTSERV will return most command output also via interactive messages that will be displayed on screen. For commands that generate a large amount of output, the MAIL prefix can be used to direct LISTSERV to send this output to you as e-mail instead of to your screen. To do this, the command MAIL is prefixed to the beginning of any LISTSERV command. For example, the SHOW DPATHS * command returns a large amount of information. This can be sent via e-mail with the following command:

```
MAIL SHOW DPATHS *
```

Other Sources of Information

A standard set of help files are available upon request from each LISTSERV server. To get a copy of these files, use the **INFO** command (see the section **Commands for INFORMATION**).

There are several mailing lists for discussion of technical LISTSERV issues. They are not intended for casual users, but they should be of interest to advanced users. They are:

LSTSRV-L Technical forum on LISTSERV

LSTOWN-L LISTSERV list owners' forum

LDBASE-L Forum on LISTSERV database search capabilities