This is once again a totally new manual for Fido and
FidoNet. It replaces all older documentation; the
information contained here overrides any previous
documentation.

Fourth Edition, August 1985

Dial Up or Fidonet Mail   Mailing Address

Tom Jennings   Fido Software
Fido's Board   Tom Jennings
(415)-864-1418   2269 Market St. #118
300/1200/2400 baud   San Francisco CA 94114

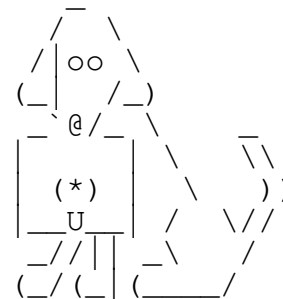FidoNet Address:
       Fido's_Board Net 125 Node 1

Usenet Address:
       ihnp4!encore!vaxine!spark!125!1!tom_jennings

       (Note: The usenet gateway, run by Bob Hartman
(FidoNode 101/101) is very new, and as of Aug 85, runs
usenet to Fido only; please contact Bob Hartman before using
the Unix/Fido gateway.)

```
                        The World's First    /¯ \
                           BBS Network      / |oo \
                           * FidoNet *     (_|   /_)
                                            _`@/_ \    _
                                           |     \ \   \\
                                           | (*)  |  \   ))
                        _____            |__U__| /   \//
                       / FIDO \            _//|| _\    /
                      (_____)          (_/ (_| (____/
```

Fido (tm) is a trademark of Tom Jennings
FidoNet (tm) is a trademark of Tom Jennings

        This manual is in four parts; you are reading <u>INTRO</u>.
Next is <u>OPERATE</u>, the manual on operating Fido and FidoNet,
<u>FIDONET</u>, how FidoNet works and what it does, and then
<u>INSTALL</u>, how to install Fido. Each section is totally
standalone, and has it's own table of contents and index.

        The <u>INTRO</u> section contains:

        <u>Supported</u> <u>Computers</u> <u>and</u> <u>Modems</u>

        At the time of this writing, Fido supports the
following hardware and software. Please contact Fido
Software for the most current list of supported hardware;
new machines are added as time goes on.

IBM PC, XT, AT, Junior
DEC Rainbow, 100, 100+, 100B
Otrona Attache 8:16
Sanyo 555
Victor 9000

D.C. Hayes SmartModem 300, 1200, 2400 and clones
  EXCEPT Anchor Automation
Racal Vadic VA212
DEC DF03
Novation SmartCat series

        Fido requires 256K of memory, and MSDOS 2.00 or
higher, and at least two floppies. It is recommended that
you use a hard disk.

        Fido is a software system that contains a non
realtime packet switched electronic mail system, plus a full
featured bulletin board system to support it. Fido uses the
hierarchical file structure of MSDOS version 2 and 3. The
basic features are:

<u>Fido</u> <u>Bulletin</u> <u>Board</u> <u>System</u>

        - Unlimited number of users
        - Absolute security
        - Multiple message and file areas
        - Consistent Email commands
        - Eight file transfer methods

        Users, commands, message and file areas all have
seperate privelege levels; these can be intermixed to allow
free access to all areas, limited access to some areas, or
locking each user into a specific area.

        Fido uses pathname strings for accessing all disk
files; this affords complete control of file uploads,
downloads, work files and messages.

<u>FidoNet</u> <u>Packet</u> <u>Switched</u> <u>Electronic</u> <u>Mail</u>

        FidoNet is a packet switch electronic mail system
contained within Fido. It supports almost any network
topology; amorphous, ring, star, or any mixture. Any node
can act as host for one or more nodes, to any practical
number of levels deep. Routing and hosting is controlled via
a text file; there is a true English Language route
processor available in the public domain. Routing is chosen
manually; FidoNet executes it automatically, with error
handling and error recovery, and a full log file of net
activity.

        The public domain FidoNet is organized into Nets,
Regions and Nodes. A node is an individual FidoNet system. A
net is a group of FidoNets in a local calling area; a net
provides many advantageous services. Regions are FidoNet
nodes in a geographical area that cannot yet take advantage
of packet routing within a local calling area.

        FidoNet supports up to 32767 regions and nets, each

containing up to 32767 nodes.

At this writing there are more than 400 nodes in more than 30 nets and regions in the continental US; others are in Indonesia, Sweden, England, The Netherlands, etc. Any FidoNet may access any other FidoNet in the world, usually overnight. Nets and regions are added constantly; please obtain a current NodeList file for the latest listings. The NodeList is published weekly.

The following people have contributed greatly to the design of Fido and FidoNet. I know I have omitted many people from this list; apologies ahead of time. Thanks also to the many Fido sysops and callers that contributed ideas (and found the many bugs).

Special thanks to John Madill, who thought out many of the nicer Fido features, and did much of my debugging for me. (And contributed from end to end; he finally got me to do PICKUP and POLL ...)

Fido and FidoNet Design Crew:

        John Madill, Baltimore, MD   (Fido #2)
        Tony Clark, St. Louis, MO   (Fido #4)
        Danny Feinsmith, NYC, NY   (Fido #8)
        Jim Ryan, Danbury, CT   (Fido #9)
        Ben Baker, St. Louis, MO   (Fido #10)
        Vern Crawford, San Jose, CA   (Fido #13)

FidoNet Scheduling and Routing:

        The St. Louis DECUS Group:
           Ken Kaplan, Mike Mellinger, Jon Wichman
           (Fido's 16, 17, 22)

And general thanks to all of the other Fido sysops for suggestions.

        For all commercial uses, except as noted below, all
that is required is to purchase a copy of Fido from Fido
Software, address below. This provides you with a diskette,
a printed manual, and unlimited updates (see Obtaining
Fido). If you will have a "small" number of installations
(under 5), make copies as necessary. You need not purchase a
copy for each site (though Fido Software wouldn't mind if
you did ...)

        If you are to have a large number of installations,
please write Fido Software at the address below for special
arrangements.

        Volunteer run, not for profit organizations
(irregardless of any non-profit status) serving their
community can use Fido freely; they can either purchase one
copy to get the update capability, or download it for free
and maintain it themselves.

        Obviously Fido Software relies on your honest
evaluation of the installation. The above scheme is
suggested because it is simple and non-obnoxious to both you
and Fido Software. Please contact Fido Software if you have
any questions.

        Purchasing a copy of Fido is considered by me to be
authorization for commercial use.

        Please note that these are guidelines; it is not the
policy of Fido Software to define rigid or unreasonable
licensing policies, and in any case contracts of any kind
are nearly impossible to enforce. It is up to you to

determine what type of installation you have, and purchase
copies or not as the case may be.

        There is a large public domain, public access
electronic mail network comprised of independent
Fido/FidoNet bulletin board systems in North America and
Europe. The net is growing very quickly; in May 84 there
were two nodes, Sept 84 fifty, March 85 there were 160, in
Aug 85, 400.

        Each of the FidoNet nodes (FidoNodes) is a totally
independent, generally privately owned bulletin board system
using Fido. Late at night, they all run the FidoNet portion
of the Fido system, and exchange messages and files.

        The messages and files transferred are ones entered
by users of the particular FidoNode; besides entering the
usualy BBS message type information (name, subject, etc)
they select which FidoNode; unlike Usenet, the FidoNet
software worries about such things as routing.

        FidoNet does it's electronic mail transfers using a
packet system that supports addresses within the packets and
the messages themselves, and a complex routing language that
lets networks or any arbitrary complexity be generated for
effecient use of the phone lines.

        FidoNet is getting more organized as time goes on;
updating nodelists (central to FidoNet's organization) will
be be totally automated very soon; the newsletter (FidoNews,

published by Fido 107/375) is already published
mechanically, and distributed further and further as time
goes on.

Getting a node number

The structure, organization of FidoNet, and the
software itself are all constantly changing; it is not
possible for me to tell you exactly what to do to get a node
number and join the net. What I can do is point you in the
right direction to figure out what the process is at the
moment.

At the moment, FidoNet is organized into Networks
and Regions; Networks are concentrated FidoNodes, such as in
Los Angeles, New York, etc. Regions are geographical areas,
such as the South West, New England, etc.

You need to determine who is your network or region
host; they provide you with a node number. After rummaging
through this manual and getting your system up and running,
contact Fido 100/51 (FidoNet Central) or Fido 125/1 (thats
me) and read the bulletins; 100/51 keeps the list of
networks and regions available.

I hope this is not all obsolete by the time you read
it; good luck!

Obtaining Fido

The Fido/FidoNet software can be obtained by
downloading it, for free, from Fido #1 or any other system,
or directly from Fido Software. Purchasing Fido enables you
to get unlimited updates for a small fee.

The following are the only computers and modems
supported at this time. Fido is MSDOS only and will not be
ported to any other DOS.

COMPUTERS SUPPORTED
        IBM PC, XT, Jr. and AT and HARDWARE clones
        DEC Rainbow 100 (MSDOS 2.05 and later)
        Otrona Attache 8:16
        Sanyo 555
        Victor 9000

MODEMS SUPPORTED
        DC Hayes Smartmodem 1200 and clones
        DC Hayes Smartmodem 2400 and clones
        US Robotics Courier 2400
        Racal Vadic VA212

DEC DF03
Ventel MD-212 with WECO EPROM

Updates

        Updates are provided to Fido purchasers; downloaders
must obtain their own updates. You can receive an update at
any time by returning your original diskette in a returnable
diskette mailer plus $20.00. You will receive the latest
distribution diskette, plus any errata sheets for the
manual.

        New manuals are not included with updates, though
errata sheets are. New manuals are released on major
revision changes (9 to 10, 10 to 11, etc).

        The latest version is always available free for
downloading from Fido #1. Fido is a public domain program,
and direct support cannot be done without special
arrangements.

Ordering

        Send your order to the address inside the front
cover, specifying what computer you want it for; if you do
not specify, the IBM version will be shipped. Please do not
order for machines not listed above.

        -- NEW PRICES: 1 SEPT 1985 --

Fido and FidoNet system, all versions   : $100.00
  CA orders add 6.5% sales tax   : $ 6.50
Printed Manual only   : $ 35.00
Software Update (plus orig. diskette)   : $ 20.00

        This manual is divided into three major sections. If
you download this manual, you will have found it in three
files; these sections do correspond to the logical layout of
this manual. I apologize for the intimidating size.

        Each section is more or less self contained, with
it's own index and table of contents. The bad part of this
is that you have to refer to the right section to find
something; luckily, Fido is partitioned well enough to allow
this kind of manual.

        OPERATE FIDO-1-3.PRN

This is the section you are reading now. It covers
operating Fido; it assumes that you understand basically
what Fido is about, and that you already have it installed.

The Operating section is first, because it's assumed
that once you install Fido you will refer to this section
more that the Installation notes.

FIDONET FIDO-2-3.PRN

This covers the theory, installation and operation
of the FidoNet packet switch electronic mail section of
Fido. It tells you how to find out the process invloved in
hooking up to the network in the U.S.; for Europe and Asia,
you need to find the systems located in England, Sweden, or
Jakarta; the sysops there can point you in the right
direction.

INSTALL FIDO-3-3.PRN

This section covers installing the software on the
hardware you plan to use. There are step by step
instructions for setting up a minimal board, and hints on
customizing it and making Fido do what you want.

Fido's Complete Operating Manual   Page OPERATE-2
Table of Contents

Fido's Complete Operating Manual   Page OPERATE-3
Sysop Only Commands

     The sysop commands are all single digits, and do not
show up in any prompts nor in any of the supplied help

files. Most are "dangerous", in the sense that they set and change Fidos environment, and are not for general use.

With one exception, all sysop commands are available at all prompts. This exception is at the "Read Command" prompt; entering numbers specifies a message number, not a command! Therefore, the 3 command (see below) is accessed only there as "#".

These commands are hardcoded for SYSOP privilege; this cannot be changed. This also means it is not possible for a NORMAL user to ever access these commands.

Sets paths, select and create new system files (SYSTEMn.BBS) and change the privilege level for each area.

The Installation Section walks you through setting up Fido with the 1 command, and provides many real life examples.

Entering 1 by itself lists the contents of the current system file. Changing and setting system file parameters is done by entering them after the 1, in a slightly peculiar way. The 1 commands are:

```
      1                   List the current set of paths
(a)   1 <number>          Select or create a system file.
(b)   1 <pathname>\M      Set the MSG Path
(c)   1 <pathname>\F      Set the FILE Path
(d)   1 <pathname>\U      Set the UPLOAD Path
(e)   1 <pathname>\H      Set the HELP Path
(g)   1 <privilege>\V     Set the Area privilege level
(h)   1 <number>\A        Set Attributes
(i)   1 S                 Save the changes
(j)   1 ?                 Get HELP! With the 1 command
```

If (a) is not done, then the system file is the one last selected when in a Message Section or File Section. You can always find out by just entering 1 by itself; it lists which system file and all its contents. (You could do the same thing by using the A)rea command in the Message or File Section.)

To create a new system file, enter 1 <number>, where <number> is one higher than the highest one. Fido will not allow creating SYSTEM99.BBS, for instance, when the current highest is only 8. When a new one is created, it copies the contents of the current one into the new one.

For (b) through (e), the specified path is selected. You can check your changes by entering 1 by itself.

(g) sets the privilege level for the Message and File areas in this system file. The default is DISGRACE. (This is not copied from the previous one; it is always set to DISGRACE.) To change it, enter as shown. The privilege levels are:

```
      TWIT
      DISGRACE
      NORMAL
      PRIVEL
      EXTRA
      SYSOP
```

You must spell fully the privilege level name.

Each system file has an attribute word. Currently, only

one bit of it is used, so the method of changing it is
crude. The 1 bit determines whether the MSG Path is a
regular Message area or a FidoNet Mail area.

        1 0\A   Set normal Message Area
        1 1\A   Set Fidonet Area

        This should be set on only one MSG Path; the one you
use for Fidonet mail.

NOTE:
        Any changes you make are not permanent until you save
them with (i). If you make changes, then select another area
with (a), then the changes are lost.

        The 2 command is helpful to manage your message
areas. It is used to delete messages by age or (RECV'D)
status.

        The first option is to delete messages by age. YOu
enter the number of days old, and Fido kills messages older
than that many days. Message numbers are listed as they are
deleted; you can Control-C to stop it at any point.

        The second option is whether messages that have been
received by the addressee (the message will be listed
(RECV'D)) should be deleted. If you answer Yes, then any
message with (RECV'D) in the header will be deleted,
regardless of it's age.

        Note that if you answer No to both questions, Fido
is stupid enough to read all the messages anyways; no harm
will be done. Just Control-C to abort it at any time.

HINT:
        Try not to use this command when you are not in the
Message Section. Until you enter the Message Section, Fido
hasn't selected a set of paths, etc. and I have no idea what
happens if you try to use this command then.

        The same applies to the 8 command.

-- IN THE READ COMMAND PROMPT ONLY

NOTE:
        The "3" command is the "#" command while in the
R)ead Message prompt, because "3" is a message number.

     The 3 command is used to change and list the privilege
levels for each command in each Section. 3 by itself lists
the command names and the privilege level needed to be able
to use it. If the users privilege level isn't high enough,
then the command will not show in the prompt.

     To change a privilege level, you enter:

     3 <cmd> <privilege>

     Where <cmd> is the command letter as normally entered
(for instance, in a Message Section, K for Kill message,
etc) and <privilege> is the name of the privilege level, as
listed under the 1 command. The change is stored
immediately.

     The following are considered separate areas, as far as
the 3 command goes:

          Main Section,   MAINPRIV.BBS
          Message Sections,   MSGPRIV.BBS
          Mail Sections,   MAILPRIV.BBS
          File Sections,   FILEPRIV.BBS
          Read Command Prompt   READPRIV.BBS
          CHANGE Command Prompt   CHGPRIV.BBS

     Though FidoNet message sections are almost identical to
regular Message Sections, it has it's own set of privilege
levels for the commands.

HINT:

        If you really mess up, you can start over by

deleteing the PRIV file, or all of them. Next time you run
Fido, it will start over with all of it's orignal defaults.

        If you haven't yet, read the FidoNet Manual first,
otherwise this may be meaningless.

    The 4 command sets and displays FidoNet parameters.
These parameters are used mainly by FidoNet, and possibly
external programs.

        4 ?   Get help!
        4 <number>\N   Set Node Number
        4 <number>\T   Set Net Number
        4 <number>\H   Set Alternate Node number
        4 <number>\A   Set Alternate Net number

        4 <pathname>\P Set Mail Path
        4 <pathname>\F Set Mail File Path

    The Mail Path is the message directory you want to use
for the FidoNet mail area. It must be the same as one of the
message paths set with the 1 command, and the one that has
the MAIL attribute set with "1 1\A".

    The Mail File Path is where files received via FidoNet
are placed. Generally a good place is the same as where you
put uploaded files, or a seperate directory only for mail
files.

    The node number, net number, and the alternate net and
node numbers are the way that your Fido system is identified
to others on the Net. These numbers must be unique, and
known to everyone else in the Net. Please refer to the
FidoNet section for details. Use this command to set your

node and net numbers once you have figured out what they
are.

5 -- List scheduled events

     Lists all of the scheduled events that Fido can
execute. Events are added, changed, enabled, disabled, etc
with the 6 command and the 7 command. The default is one
event enabling the Y)ell command.

     The actual scheduler operation is covered in the
Scheduler chapter of the FidoNet section. Basically, these
events are things like when to execute FidoNet, days and
times when the Y)ell command is allowed, and with additional
work, when external programs are to be executed.

         See the 6 -- Enter/Change Events section (next page)
for details on what events mean and how to change them.

6 -- Enter/change events

     This command allows entering and changing events. There
is room for up to 35 events. You are asked for an event
number, then you are asked for each of the components that
make up an event. These are:

Event # [0 - 35]:

     Which event to work on. Events can be in any order;
Fido executes them according to the day and time. If you
enter a number that is already used, you will replace it
with the new one you are entering. Hitting RETURN will exit
the 6 command.

Day of week [All, 0=Sun, 1=Mon ...]:

     The day of the week that this event should be run on.
Entering A (for ALL) means run it every day. For example,
FidoNet gets run every day; enter A. An alternative (just
for an example) would be to enter seven events, one for each
day of the week. This is just for convenience.

Start Time: [hh:mm]:

    The time this event is to start. Enter as 10:44, etc,
or use a space instead of the colon. Entering just 10, for
instance, is the same as entering 10:00. 00:00 (or just 0)
is midnight; 23:59 is one minute til midnight.

Window width, Minutes:

    How long the specified event should be run. This can be
from 1 to 1440, the number of minutes in a day. If you had
entered A for the day of the week, 10:00 for the start time,
and 60 for window width, then the event would be run from
10:00AM to 11:00AM.

    For external events, you should enter one minute; Fido
cannot "call back" an external program when it's time is
over, it is up to that program (and your batch file) to
return to Fido.

Schedule Tag [A - W, X, Y, Z]:

    This tells Fido what event it is you are running. A - W
are FidoNet events; Fido's FidoNet Manual describes this in
detail. Tag A is the normal, nation wide FidoNet time slot.
If in doubt, or for new systems, enter A. This will then be
compatible with older FidoNets.

    Y is the yell command. The Y)ell command (which rings
the bell annoyingly on the console) can be enabled for any
time of the day, by setting it up as an "event". Fido will
not terminate or do anything different; it will just allow
the yell command to be used while the event is running. The
default is to allow the yell command from 10:00AM til

10:00PM.

    Tag X is the external event; with these, you can run
other DOS programs through a specially modified batch file.
X terminates with DTR low (the modem hung up, and auto-
answer disabled).

ERRORLEVEL [4 - 255]:

    Asked ONLY if Tag X is entered, above. This is the
ERRORLEVEL code passed back to DOS, where it can be detected
by the batch file, for selectively running programs.

7 -- Enable/disable/remove an event

        This is used to enable, disable and remove set events.
You are asked for an event number (as entered above) and
then what to do with it. If you enter an event number that
does not exist, it will tell you so.

        Enable "turns on" that event, that is, allows it to
run. Disable causes it to be ignored, though it will still
exist, so you can enable it later if you want. Remove

deletes it entirely.

8 -- Renumber Messages

        The 8 command will renumber all the messages in the
current message area, so that they start at 1, and increase

by 1. It will preserve Replies and See Alsos.

HINT:
        It is recomended that you use the 8 command when you
have either a lot of messages, or lots of "holes", ie.
groups of numbers that are missing. When searching for
messages, Fido searches through the missing message numbers,
which takes time. Use of the 8 command regularly (try once a
week at first, adjust accordingly) will speed up your
system.

        Renumbering causes temporary great upheaval to the
messages themselves; interrupting the 8 command (by powering
off or resetting) is not reccomended. To prevent losing the
connection (and disconnecting) from aborting the renum,
nothing is displayed to the modem while the command is
working. The local console will show 8 command status.

        This is a highly specialized command, that works
only when a /W value is specified on the command line.
Please note that Fido does *NOT* run programs through the
modem; all the 0 command is for is terminating Fido in an
orderly way so that yo can use OTHER programs, not suppiled
with Fido, for running other programs through the modem.

        When set, the 0 command causes a immediate
termination to DOS with the ERRORLEVEL specified by the /W
switch. It is assumed that your RUNBBS batch file will trap
this errorlevel, and do something appropriate.

        Note that it is NOT adequate to just use CTTY to
redirect the console; you must have a special "watchdog"
program to monitor the Carrier Detect line from the modem,
and reboot the system if carrier is lost. Otherwise, if the
connection is lost, the next person that calls in has
control of your system, from the DOS level.

     There are a number of commands that can be used from
the system's local console while a user is on. This is
sometimes called "spy mode", as the user does not have any
indication that you are spying on them. The commands are:

     ?
     /
          Displays the name of the current user, and how
long he has been on. It lists a small menu of other things
that can be done, described here.

     C
          Chat with the caller. (Also see the Y)ell command.)
The user will get an appropriate message, and you will then
be in terminal mode with the caller, allowing you to type
back and forth. The chat mode is terminated by typing
Control-Z, at which point the user is back in normal Fido
operation.

     Z
          Clear the current users limits. This clears the
accumulated download, time on system, and accumulated time
on system. (same as user calling back in 48 hours.) This is
useful for extending someones time limits temporarily.

     ^A (Control-A)

     Simultaneous keyboards. All keystrokes typed are
treated as if they were from the callers modem. Terminated
by ^Z, Control-Z, below. This can be used for intensive
handholding of terrified users.

     ^Z (Control-Z)

     Terminates both Chat mode (or the Y)ell command) and
simultaneous keyboards.

     ^X (Control-X)

     Immediately terminate a call. The user is logged
off. Ignored during file transfers. To be used sparingly. If
you need to get into the system while a caller is on, refer
to ^T.

     ^O (Control-O)
     ^N (Control-N)

     These two are for special purposes only. They turn
off (^O) and on (^N) the local screen output. Normally, Fido

displays what the user sees, on the local screen.

        For use with a multi-DOS type program (XAP,
Multilink, etc) this console activity greatly slows down the
system. It can be disabled with ^O, or via the /I command

line switch. These two keys override and /I command line
switch.

        ^T (Control-T)

        When entered while a user is on, after the user
hangs up it causes Fido to disable the modem, and alert the
operator that the system is free. Fido rings the bell once a
second for ten minutes.

        When Fido is ready, type ^C to quit to DOS, or ^Z to
stop the terminate and resume normal operation. If there is
no response in five minutes, then Fido resumes normal
operation automatically.

    Space bar

     When a user has started a Y)ell, hitting the space bar
immediately causes the "sysop not available" message.

        The scheduler can be used to limit the hours in which
the Y)ell command can be used.

     A questionnaire is a pseudo-language file that is
interpreted by Fido. If someone invokes the A (Answer
Questionnaire) command in the Main Section, and there is no
questionnaire, it says:

  "No questionnaire today"

     Questionnaires are simple text files that are
interpreted by Fido. There is a very simple line oriented
pseudo language that interprets these scripts. There are
only a couple of "commands", and only two crude
conditionals.

     The answers to the questionnaire are put in a seperate
file, without the questions. They are also normal ASCII
text, but in a fixed field format with field characters, to
make post- processing of the questionnaire easier. (I do
not, and probably will not ever, provide a processor for the
answers files.)

     The questionnaire consists of two files:

QUESTION.BBS   The questionnaire script
ANSWERS.BBS   The answers

     Answers are appended to the end of the answer file; it
is create first if it does not exist.

     There is also a second questionnaire: QNOPWD.BBS. If
this exists, users who forgot their password, or if the /M
(private system) is specified, they get to fill it in. You
can put in requests for passwords, address and phone
numbers, etc here.

     The first character of each line tells Fido what to do
with the line. This first character is a normal printable
character. If the first character is not one of the known
commands, the line is merely displayed.

HINT:
    Do not have blank lines in the questionnaire.
Interrogation will stop at end of file or a blank line.

    The questionnaire commands are:

/<prompt string>
    Display <prompt string>, then input a single line of
text. Maximum line length is 78 characters. The usual
editing is available. The inputted text is written to the
answer file.

*
    Put users info to the answer file: name and last
(current) signon time and date. Useful for the "what kind of
system ..." type questionnaires. Leaving this out allows

confidential questionnaires, as there is no other record of
which caller filled on in.

+N <prompt string>
    Multiple choice. The single digit number N is the
number of possible choices. (1 - 9) If an illegal number or
the number is missing, the number of choices defaults to
one, so at least it wont hang up. The choice entered by the
user is checked for in range 1 to N, and saved in the answer
file.

    There is a special feature of multiple choice: an
internal flag is set if the last choice (choice N) is
selected, that can be tested by the ? and ! commands, below.

? <prompt string>
    To be used ONLY immediately after a multiple choice
question. This command will be executed only if choice N is
made. This allows implementing "A, B, C, Other" type
questions. Put "other" as the last choice; if selected, then
the ? command following it will be executed. See the example
below. If any other selection was made, this line will be
skipped.

    The conditional flag is cleared by: + command, ?
command, _ command. No other commands affect it.

!
    To be used ONLY after a multiple choice question,
similar to ? above. If the last choice of a multiple choice
was selected, this terminates the questionnaire. Allows
asking "Do you want to fill this out (1) yes (2) no".
_

(Underscore) Unconditionally clear the conditional
flag.

Questionnaire Example

        The following is an example of an executable
questionnaire, though the questions are useless.

This is not a command, and will merely be displayed. This
can be used as a header or prompt for long questions.
* Insert user info (text after * is ignored)
Want to fill this out:
+2 (1) Yes (2) No
!
/Prompt, input a line of text:
+3Multiple choice: (1) green (2) blue (3) other:
?What other color?
?will never be executed; ? above cleared flag.
+2Multiple choice: (1) this (2) that
This is just plain text. Does not affect the flag.
?What is that?
Text can be used as a prompt for a long question, like this:
What time did you last jump out the window?
/

Example Answer File

    The answer file is a specially formatted text file. It
can (has to be for now) interpreted manually. (The answer

file for QNOPWD.BBS is ANOPWD.BBS) For the questionnaire
above, here's what the answer might look like: (manual
comments in parens)

```
* Some User 32 Jan 44 23:59:59
  1: A line of text    (text inputted by / command)
  2: 2                 (choice 2; ? after it not executed)
  3: 2                 (choice 2. == N, so ...)
? 3: the other thing   (conditional question, "other")
  4: at midnight       (ans to "when ... out the window?")
```


    Each question is numbered by BBS. Characters are 3
digits max, right justified.

Special Characters in Column 1:

* User record info.

? Answer to conditional question. Note that the number is
the same as the question that triggered the conditional
question, even if there was text inbetween.

(blank)   Normal answer.

    The SYSOP utility is used to maintain the user list
file, USER.BBS. It is a simple data base type program, has
online help, and is fairly easy to use.

    First, a general description of the user list. The user
list is a variable number of records, with one record per
user end to end. New users get added to the end. SYSOP
allows going back and forth through the user list, changing
things such as names, passwords, privilege levels, and more
global things like purging old users from the list.

    It will be much easier to use this section while
actually running the SYSOP program. When first run, SYSOP
displays the first record in the list, usually the sysop.
You can get a brief description of each of the commands by
entering a ?, then the command key listed in the menu.

    SYSOP automatically saves any changes you make. To
change a password, for instance, use the P command (below)
and thats it. SYSOP will save any changes before exiting to

DOS or selecting another user record.

Selecting a User Record

     There are a number of ways to look at (and maybe change) user records. You can use the following key to move to the next and previous record:

          Next Highest   Previous
            >   <
            +   -
            .   '
            _   =

     The last four don't appear to make any sense, but they happen to share the same keytop, either shifted or unshifted.

B -- Beginning of the User List

     B takes you directly to the first record.

Z -- End of the Users List

     Z takes you to the end of the list.

A -- Select Record Number

     You can also enter the record number directly with the A command. This is handy only if you know the record number.

F -- Find a Record

     This works similar to the I command in Fido. You are prompted for a string; SYSOP then searches for that string

in the name portion of the user records, and stops at the first one it finds.

     It starts searching from the next record; i.e. if you are currently at #100, it starts looking at #101. When it reaches the end, it wraps around, and continues from record #1. It stops if a match is found, or until it reaches the point where it started. Therefore, if no match is found, it leaves you where you were.

     If you are searching for "John", you will probably get more than one match. If it matches one you did not want, then just repeat it; it will start searching again with the next record. For speed, you can use the Control-R key as in Fido, to recover the text you had entered ("John" in this

example, by typing:

```
    F          Find command
    ^R         Restore old line
    CR         Execute command
```

E -- Enter New User

      E adds a new user to the list. First, it goes to the end of the list (may take a while) then makes a clear, empty user record. You use the usual SYSOP commands (below) to fill out the record.

      After doing an E, SYSOP will save the new user record only if you change at least one field in it. If you just do an E, then a command such as B or - it will not save the new record. Once you have changed anything (name, password, etc) it will save it automatically.

N -- Name

      N prompts you for a new name. You can enter one or two words. Entering a blank line leaves the present name.

P -- Password

      Enter a new password.

V -- privilege Level

      Change users privilege level. You can enter only the first letter of the privilege level name, or enter the whole thing. The levels are listed by "? V". They are:

```
    TWIT
    DISGRACE
    NORMAL
    PRIVEL
    EXTRA
    SYSOP
```

Q -- Quit to DOS

      Saves any changes, and returns to DOS.

O -- Mark Old Users

      This is used to mark old users that haven't called in for a long time, to be later purged from the list. (One time callers, etc) The user records are not deleted by this

command; you must use the ! command to do it after you have them all marked.

It starts at the current record, and if a caller hasnt called within the specified number of days, that record is marked as "deleted". Users within the number of days are marked as "Un-deleted". You can use this if you change your mind, for instance, to un-delete everyone use the O command to mark all users that havent called in, say, 1000 days. This will undelete anyone that has called in within 1000 days.

A good number to use is 32; you will find that almost all users that get deleted have called only once. SYSOP displays each user record that it is marking.

D -- Mark Current User Record as Deleted

This marks the current record as deleted, as if it were marked by O.

! -- Purge deleted Users

This does the actual deletion from the user list of any marked records. First, it makes a backup copy of USER.BBS in USER.BAK. As it deletes records, it puts the deleted ones into USER.OLD, where they can be recovered from if a mistake is made.

You can recover deleted records by either renaming USER.BAK to USER.BBS, or by appending USER.OLD to USER.BBS by the DOS command:

        REN USER.BBS USER.WRK   rename for now,
        COPY/B USER.WRK+USER.OLD USER.BBS

Note that SYSOP will not delete users that have a privilege of EXTRA or above or any credit left.

$ -- Set Credit and Debit

This is used for supplying users with credit for sending FidoNet mail. It prompts you for a dollar value, which is the added to the current credit, after subtracting any debit from it. If there were already $5.16 credit left, entering "10" would leave them with $15.16 credit.

To set the credit to a specific number, enter:

        =10

This clears any remaining credit before adding the $10.

M -- Set Default Message Area

Fido saves the message area that a user was in last. Normally, new users end up in message area #1. For certain types of systems, you might want to eliminate the A (Area-Change) command, and lock users into particular areas. This determines what area the user will end up in when entering the Message Section.

Note that if the specified area does not exist, or it's privilege level is too high, Fido will automatically bump them into area #1. Make sure you have these set up right.

K -- Set Default File Area

Same as M, except it sets the default File Area.

Aborting a SYSOP Command

Some of SYSOPs commands may take a while, such as searching for a name or marking users. You can abort most commands by hitting the ESCape key. It will stop at the current record.

        Fido keeps a record of all callers in a simple log
that records the number of callers per hour, for every day,
for seven days. When it is seven days old, it gets renamed
to the current date, and a new one is started.

        The TIMELOG.BBS file contains 168 slots: one for
each hour of the day, for seven days, plus the date when
started and ended, etc. Every time someone calls in and logs
in succesfully, (enter their name, and guess their password)
Fido increments the appropriate slot. When all seven days
are used, it renames it to the current date (TIMELOG.BBS
becomes for instance, 01JUL84.TLG, or 21MAY84.TLG, etc) and
starts a new TIMELOG.BBS.

        These .TLG files are fairly small; 378 bytes each,
so you dont have to worry about them filling your disk for
quite a few months.

        Note that TIMELOG.BBS is not started on Sunday; it
starts whenever you first run Fido. If you want to
synchronize it to start on a particular day, delete
TIMELOG.BBS on that day of the week, and when Fido is run,
it will create a new one.

Running TIMELOG.EXE

        TIMELOG.EXE will run on any MSDOS machine, and is
not hardware sensitive. To run it, enter:

        TIMELOG <name>

        Where <name> is the name of a .TLG file. It will
display the contents as crude bar graphs. There are two
options:

        TIMELOG <name> /P   Pause
        TIMELOG <name> /F   Format

        /P tells TIMELOG to pause between each graph (so
they dont run off the screen). /F formats the output for a
printer, putting formfeeds in between each graph. (To send
to the printer, use redirection: TIMELOG <name> /F >PRN)

        Fido keeps a running log of all system activity.
Every time a caller connects to Fido, information is being
logged. The log is added to whenever someone calls in, and
logging never stops, unless the disk fills up. (See the HINT
below ...)

        The log file is organized line by line. The first
character on the line defines what type of entry it is;
types defined so far are:

        + user name and time on
        ! errors of all sorts
        = download and upload info
        - line between entries

        When a call is logged, Fido installs a line of
dashes to make it readable. followed by the users name, time
of connect and baud rate.

HINT:
        The log files generate by Fido can get quite large.
If you don't want to keep the logs around, or if you want to
print or COPY them then delete them, you can do so
automatically with Fido's scheduler.

        Create an External Event, and in your RUNBBS batch
file, add the following or something like it. (Assume for
now that the X event's ERRORLEVEL is 10.)

```
:fido
        fido_ibm /switches ...
        IF ERRORLEVEL 10 GOTO LOGSTUFF
        ...
:LOGSTUFF
        copy *.log prn          print the log or whatever
        del *.log
        goto fido
```

        You can of course think of varioations on this
theme.

FidoBBS is a Bulletin Board system that takes full
advantage of MSDOS version 2. Most of the features were
modeled after the usual CBBS and RBBS software, since the
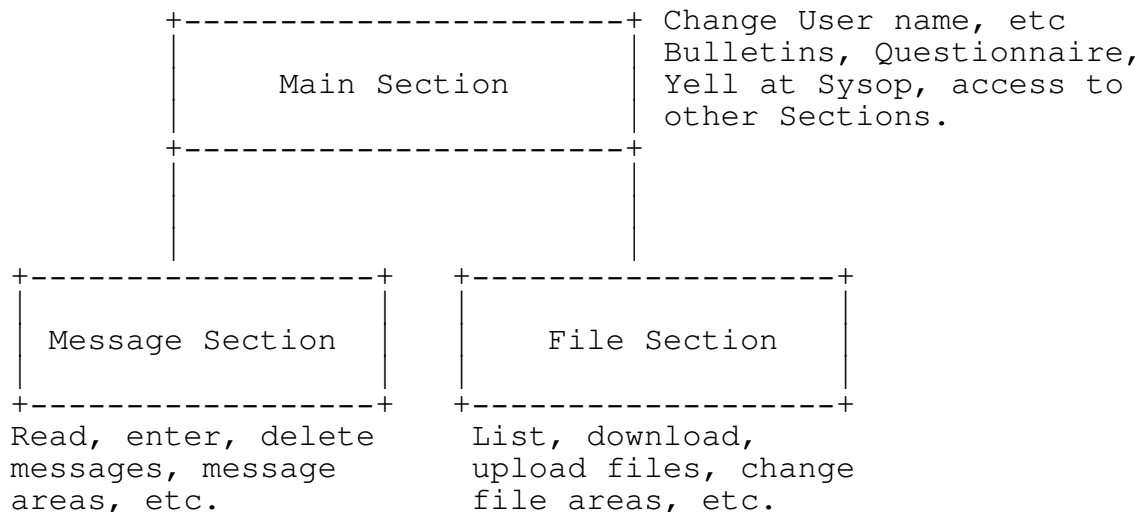command system is so popular and familiar.

There are three help levels; the default for new
callers is NOVICE. This provides a small menu of commands,
and usually gives further description as commands are
executed.

REGULAR is the command letter prompts, minus the
full words, and does not tell you what command you have just
executed. Useful for after you understand the structure,
but have not memorized all the command mnemonics.

EXPERT is just a very short command prompt; there
are no menus or other things; prompts are very small and
fast. Once you are very familiar with Fido, this will make
using it very fast.

Regardless of help level, you can enter ? at almost
any point for a list of the options at that point.

From the Main Section, all of Fido's available
commands can be accessed. The actual commands are described
in detail below. Fido is arranged into three sections:

```
        +------------------------+ Change User name, etc
        |                        | Bulletins, Questionnaire,
        |      Main Section      | Yell at Sysop, access to
        |                        | other Sections.
        +----------------------+
        |                        |
        |                        |
        |                        |
+------------------+   +------------------+
|                  |   |                  |
|  Message Section |   |   File Section   |
|                  |   |                  |
+------------------+   +------------------+
Read, enter, delete    List, download,
messages, message      upload files, change
areas, etc.            file areas, etc.
```

<u>Commands</u>

All commands in all areas are one letter, though some require further arguments or options. Most command allow entering the further arguments right after the command key; if not, or if not supplied, each necessary component is prompted for, usually with help available at each point.

For instance, to download a file you must enter the

D command, the download method (one of seven) then the filename. If you entered D, then Fido would ask for the download method. After that, it asks for the filename. Once you become familiar with it, you can enter it all as:

D X FILENAME.EXE

All on one line, skipping the prompts. This works for most commands.

All commands can be aborted with a Q or a blank line.

All Sections

The following are commands that are available in all Sections.

G)oodBye

Logoff the system, optionally leave a private message to the sysop. If a message is left, it goes into the special message area (in system area 0) that is accessable only by the sysop. The usual message editor is available for comments.

If there is no message path set for area #0, then Fido will not ask for a comment to the sysop.

S)tatistics

This command lists the time on the system, time remaining, etc. Also, depending on the area, it will list further information.

Files Area
Statistics also lists the free space left on the default drive.

Message Area
        Optionally lists messages to or from you.

FidoNet Area
        As in Message areas, and also lists FidoNet
statistics.

                Main Menu

        This is the section you are in after signon, right
after the Bulletins and quotations. From here, the "system"
type commands are accessed, such as Change user settings,
etc. The commands accessible from here are described below.

        M   M)essage-Section
        F   F)ile-Section

        Y   Y)ell at the Sysop
        A   A)nswer a Questionnaire
        B   Get B)ulletin again
        E   Get the E)ditorial
        C   C)hange User settings
        S   S)tatistics
        U   Display U)sers List
        G   G)oodbye, logoff

M)essage-Area

        Enter the message section. From there, the message
areas can be accessed.

F)iles-Area

        Similar to Message-Area, but where the upload and
download commands are.

A)nswer-Questionnaire

        If there is a questionnaire in the system, you get
to fill one out with this command. A questionnaire is an
text pseudo-language file, described in the Questionnaire
section.

C)hange User

        A small sub-menu of commands to change various user
settings, such as signon name, password, screen dimensions,
etc.
        U   U)sers name
        C   C)ity and State

```
P    P)assword
W    Screen W)idth
L    Screen L)ength
H    H)elp level
M    "M)ore?" on and off
T    T)abs On and Off
N    N)ulls, 0 - 20
```

You can change your name, password, city and state, and anything else at any time. (Unless the options were removed by the sysop.)

NOTE: For security reasons, when your name is changed, the entire users list is searched to make sure the name is unique. This is to prevent having two or more people with the same name.

The screen dimensions are used by the page pauser ("More?") and the message system. Messages are listed to fit within the set screen boundaries, and message entry is done via a word wrap editor. If "More?" is ON, then Fido will pause after each screenful. All of the command prompts are made to fit within this.

Help level is either NOVICE, REGULAR or EXPERT. NOVICE is the default; it causes all prompts to be displayed fully, and each command to be described with a one liner. REGULAR is similar, but does not list the entire wordy prompt. EXPERT is almost no prompt at all.

"More?" on and off controls whenther Fido asks "More?" every screenful.

Tab expansion converts all tabs sent into spaces, for those systems that cannot handle tabs.

Fido can be set to send nulls after each linefeed. It defaults to 0 nulls.

B)ulletin

This just displays the signon bulletin, BULLETIN.BBS again.

E)ditorial

Similar to the Bulletin, (but displaying file EDTORIAL.BBS) but not displayed automatically.

Y)ell at the Sysop

By yelling, the caller can attempt to contact the sysop. It beeps for 30 seconds, and if the sysop does not respond by then, displays "Sysop not available".

The sysop can chat with the user at any time.

U)sers List

Lists the names of all current callers to the system. Not very useful.

G)oodbye

Terminates the call. It is not really necessary; however, callers seem to get upset when they cant figure out how to log off, so there it is. (Its OK to just hang up.) It also allows entering a private message to the sysop. After (optionally) entering a message, it causes a disconnect, exactly what would have happened if the caller just hung up.

The Message Section is accessed by the M command from the Main Menu. All message commands are available here. Return to the Main Menu is via the M command.

```
M     Return to M)ain Section     G)     Goodbye
A     A)rea-Change                R     R)ead messages
E     E)nter a Message            K     K)ill a message
L     L)ist Messages             S     S)tatistics
I     I)nquiry
```

A)rea-Change

Change the current message area. If an area number is entered after the A, that area is selected if available. If not, a list of areas is displayed, from which you may choose one.

If a new area is selected, then the messages there are counted (why it says "Wait ...") and if it's the first time you've selected that area, asks if you want to list

mail addressed to you. Nothing happens if you select the
same area, or blank line or Q.

        The mail search searches all messages, newest to
oldest, and lists: New message to you (ones you haven't
read yet) Old messages to you (ones you have read) and
messages you have entered. Typing Control-C or Control-K
aborts the search. If there are lots of messages in any of
the above catagories, it reminds you to Kill some.

        The message area numbers displayed may not be
sequential. Each area has an associated privelege level and
if the your privelege is not high enough, that area will not
be displayed nor available.

L)ist Messages

        Only the message headers are listed. A starting
number is requested; this can be:

        An ordinary number
        A number followed by + or -
        - itself.

        If just a number is entered, messages are listed
starting at that one, towards the highest message.

        A number followed by + or - controls the direction
messages are listed in. + is redundant. - means list
backwards; 100- then means start listing at 100, list
towards 1.

        - means list from the highest message towards the
lowest. This is the most useful, and for some reason the
least used. If you have called before, this is the easiest

way to see any new messages.

R)ead

        Actually a small menu by itself. From here messages
can be read, entered, killed, replied to, etc. See the
descriptions of the Kill and Enter commands for details on
those.

        Entering a message number reads that message. N
(Next) reads the next highest message, and P (Previous)
reads the next lowest message. Hitting CR repeats the
previous N or P; if neither was used, then it defaults to P.

        * is a special message number. It means the highest

message number.

        + and - are little used keys, but one of the most
powerful message features. Messages can have, and be,
replies. (See R command below.) Messages created with R are
linked to other messages. For example, a message about an
item for sale may have a reply. If you come across the
original for sale message, it will say "See also #nnn, use +
key" if it has a reply. Entering a + at this point will take
you directly to that message. This message will say "Reply
to #xxx, use - key". Entering - will take you back to the
original message.

        There is no limit to the number of messages that can
be linked this way. Further replies are added to the end,
and if one in the middle (or either end) is Killed, the
string of messages is "patched" to maintain continuity.

        This is my favorite feature (dreamed up by John
Madill) and I am slightly annoyed that no one uses it.

        R Replies to the current message. It acts basically
like Entering a message, but automatically fills in the
"To:" field, and links it into the original message.

        You can enter "RK" to reply to a message, and kill
it afterwards. This is extremely useful while reading your
mail.

        If you are replying to a message in the Mail area,
it also fills in the destination system. The message will be
mailed out that night.

        Please read about Entering messages, further below.

        E enters a message in the normal fashion.

E)nter a Message

        You fill in the To:, From:, Subject: fields, etc,
then the main body of the message. After terminating the
message, commands can be used to save, abort, edit, etc.

        If the area is a FidoNet Mail area, then in order to
enter a message, you must select the system to send it to at
the prompt. If you do not have enough credit, then Fido will

not let you send a message to that system. If you do, then Fido will deduct the cost of the message from your credit.

Message text entry is done with a word wrap editor. If the screen width is set properly (C command in Main Menu) then this is about as close to a full screen editor as you can get through a modem (on current DOSs anyways.)

Like WordStar, text is entered continuously without carriage returns. Words that would go past the left edge are wrapped to the next line. Carriage returns can be used to form paragraphs as in WordStar. When entered this way, later reading of a message will conform to the callers screen width.

At the Enter Message command prompt, the options are:

C for continue. Add more to the message, adding to the end.

L lists the message so far.
T lets you change the To: field.
J lets you change the subject.
D Deletes a line
I Inserts a line

E edit a line. You enter the line number, then you are prompted for the Old string (what to change) then the New string (what to change it to). Both old string and new string can be anything that you can enter, except a carriage return.

NOTE: it is possible to edit a line such that it goes beyond the edge of the screen. It will list correctly after it is saved.

A aborts the message. You will be asked yes or no.

S saves the message. The message is written out to disk, and the highest message number, etc is updated.

K)ill

Deletes the message. Will not let you delete

messages that are not To: you or From: you, unless you are
the sysop. If the message you killed was mail to a remote
system and it was not sent yet, you get your credit back;
you are not charged for that message.

I)nquiry

        Prompts you for a string. It then searches all
messages, newest to oldest for the thing you enter. It looks
in the "To:", "From:", or "Subject:" fields. If the string
you enter is in any one of those parts of the message, the
header of the message will be listed.

        Typing Control-C aborts the search. An insensitive
sliding match is done; ON matches TONY or PITON, etc. Each
message header matching is listed, then a list of all
matching message numbers.

        ? will match any character. For example, if there
are the following messages in the system:

#100 From: Joe Bungpop To: All Subject: Musical Instruments

#102 From: No Body To: W.S. Burroughs Subject: Guns

#200 From: All To: Board Subject: Music

        If you enter MUS, it will list messages 100 and
200, as MUS is contained in the Subject: fields of both. If
you had entered BU, then it would list messages 100 and 102,
as it would match "Bungpop" and Burroughs". Entering B??R
would cause #102 and #200 to be listed, as B??R matches
BURRoughs and BOARd.

        The file Section contain all the file oriented
commands for downloading, uploading, etc. The M command
returns to the Main Menu.

            M   M)ain Section
            G   G)oodbye
            A   A)rea-Change
            F   L)ist Files
            L   L)ocate specific Files
            D   D)ownload
            U   U)pload
            R   R)aw-List
            K   K)ill a file
            T   T)ype a file

A)rea-Change

        Similar to the Message Area-Change. If a valid area
number is entered, it is selected directly, otherwise a list
is displayed.

        Like the message areas, there may be numbers missing
from the sequence. These are either blank paths or high
privelege areas.

L)ocate

        Given a file specifier (wildcards reccommended) it
searches through all file directories for the filename, and
lists all occurences of it. L;*.ASM lists all .ASM files in
all areas, etc. L;*.* lists all files everywhere.

        This is not the general purpose list-files command.

F)iles

        Lists the files in the current directory. If no
specifier is given, it lists all files, otherwise only
matching ones. The size of the file and the description is
displayed after the filename.

        If "MISSING" is displayed instead of the file size,
then the file is not there, even though it is listed.

T)ype

        Very simple. Displays an ASCII file on the screen.
It statistically checks the file to make sure it is a text
file.

U)pload

        Upload a file or files to Fido. I will not even
attempt to document all the upload methods. Suffice to say
you select the upload method and do it.

        The total uploads made is kept in the user record.
These count towards further downloads, by effectively
raising the download limit on a 1 for 1 basis. It will not
give you credit however. There is no limit on uploads.

        After all files are uploaded, you are asked to
describe each one; this is added to the list of files. (You
may not be able to see the newly added files though.)

D)ownload

        Again, pick the method and do it. Before the
download starts, Fido checks time limits and K byte limits,
and prevents too long downloads. Uploading will remove some
of the download limit.

          Control characters are used by Fido for editing
command lines, pausing and aborting commands. Control- C and
Control- K always are identical and have the same effect:

          Control-C Aborts anything
          Control-K Aborts anything

          Control-S pauses the display. Any key except a
second Control-S will let it continue.

          When not using the word wrap message editor, Fido
allows simple editing of command lines. It assumes the your
or computer terminal can support at least backspace. It
supports:

          ^C, ^X, ^U, ^Y

           Delete the entire line.

          ^R

          Restore the previous line, from last use or
deletion. (Most useful.)

          ^S, ^G, ^H, DEL

          Delete one character to the left.

          ^D

           Restore one character.

          ESC, CR, LF

           Enter the line.

        Once the system is up and running, you will need to
look at messages, put files in the download display, etc.
This should be done daily or so.

TIMELOG.BBS and *.TLG FILES:
        These record the number of calls to your Fido.
Please read the section on the TimeLog.

SYSOP.LOG
        This is a record of callers and any messages they
leave at the Goodbye command. Also in here go error
messages (missing files, disk full, etc) These can be
printed or whatever, then deleted. Fido will create a new
one if it does not exist, otherwise it adds new info to the
end. This is the most useful way to watch system activity.

MAILER.LOG
    This is a record of all FidoNet activity.

USER.BBS
        This is your list of users. It grows with every new
caller, and the SYSOP utility program can be used to
maintain it.

NEW FILES
        If you use a separate upload directory, look in it
once in a while. Copy desireable files into a download
directory, and update FILES.BBS accordingly. Deleting them
afterwards makes it easy to keep track of uploaded files.

MESSAGES
        Log on to the system (via test mode or calling in)
and read and prune as necessary. SYSOP level users can read
all messages, private or not, and can delete anyones
messages.

     This is a brief accounting of some of the internal only
files that Fido can create. These are managed automatically;
you do not need to worry about them. They are included here
for the curious.

NODELIST.SYS
NODELIST.IDX

     Fido creates these files from NODELIST.BBS whenever
they are missing or NODELIST.BBS has changed. The .SYS file
is a formatted version of the .BBS file, and .IDX is an
index to make accesses faster.

     Never delete or modify any of these files from
within Fidop by U)ploading, K)illing, etc.

ROUTE.BBS
ROUTE.A
ROUTE.B
ROUTE.etc

     These files control where and how FidoNet mail is
sent. They are described in the FidoNet section.

MAIL.SYS

     Created and maintained by the 4 (sysop only) command,
this is where FidoNet related info is kept. If you delete
this, you will need to reenter the node number and mail
paths.

MSGPRIV.SYS

```
MAILPRIV.SYS
MAINPRIV.SYS
FILEPRIV.SYS
READPRIV.BBS
CHGPRIV.BBS
```

Each of these contain the names of all the normal user commands, and the privilege levels necessary to access them. One or more of these files may not exist; they are created only if a command privilege level is ever changed.

Deleting these files causes the command privileges to revert to their default settings.

```
*.IN
*.OUT
*.FLO
*.FLI
```

These are working files created and usually deleted by FidoNet. You will never see these files, except when you reset or reboot in the middle of FidoNet operation.

Fido's Internal Files

The .IN files are packets received from other FidoNet systems; they contain messages which are unpacked. The .OUT files are packets sent to other remote FidoNet systems. .FLO files are the list of files to be attached to that node; it may be empty. .FLI is the list of files received from a remote system, and may be empty or non-existent.

   15

FIDONET Table of Contents

This manual explains (hopefully) what FidoNet does,
how it does it, and how to install and run it. FidoNet is
built into, but seperate from, Fido the bulletin board
system.

Please read the FidoNet Operation chapter for
details on how Fidonet works; briefly, FidoNet is a method
of connecting many Fido Bulletin Boards in different
locations for the purpose of transferring messages and files
automatically. The net operates for a short period of time
each day; for instance, Fido the Bulletin Board runs for 23
hours each day, and Fidonet for 1 hour. During this hour,
any messages marked as mail are combined into packets, and
transferred to the each of the Fidonet programs. They are
then unpacked, and normal BBS operation resumes.

In most local areas (St. Louis, Los Angeles, London,
Karlstaad, etc) local networks have been set up for local
traffic, and use centralized hosts to send mail nationally.
If there is a local net in your area, you must join it or
the organization in the geographical area. The host or
region manager can supply you with lists of other systems,
newsletters, and other timely information on FidoNet.

FidoNet inherently "knows" about hosts and regions,
which operate similar to the telephone companies' area code
system. FidoNet routes messages accordinly; this is covered
further in a later section.

There are numerous things to control the setup, such
as available remote systems, cost control (or prevention)
and so on. Once setup, FidoNet is totally automatic,
requiring no sysop or user intervention.

        Fidonet's purpose is to link Fido based Bulletin
Board systems together for the purpose of automatic message
and file transfers.

        FidoNet is a true dialup packet switch network
system, that supports many different topologies. It supports
routing, message forwarding, scheduling, and uses a tuned
collision detection algorithm over normal phone lines, for
the lowest possible cost and highest effeciency.

        The simplest scheme, and the one to setup initially,
supports point to point messages. Most major geographical
areas have a host that will accept mail for itself and it's
local nodes.

        After you have contacted any other Fido sysops in
your area, you can tie into their local network, and take
advantage of the lower cost. Each local area runs things
differently, and cannot be covered here. If you can't find
your local region or host, contact Fido #1 at 415-864-1418,
where you can find the latest node list and other files to
help steer you in the right direction.

        The original design was built around the current
Bulletin Board architecture (if there is such a thing) which
is basically: an unknown number of completely independent,
standalone systems, with extremely low overhead in both
maintenance and cost. FidoNet was designed to be compatible
with this, in that it should be:

        (1) No extra work for the sysop,
        (2) No effect on normal BBS operations,
        (3) No unexpected extra costs,
        (4) No effect upon system reliability.

        FidoNet is totally automatic, and requires no extra
work, once set up. Other than the effect of allowing
network-wide message traffic, the only other affect to the
current BBS is that it is "down" to normal (human type)
traffic during the mail periods.

        Costs, if any, are controlled by the sysop. Unless
specifically enabled, mail will not be sent out from a node.
(Sending mail costs; receiving mail is free) Fido provides
accounting and cost limitation functions (all automatic) to
prevent unauthorized mail from being sent. There can also be
"free" traffic to non-toll call nodes. The usual privelege
levels can be applied to each of the Mail commands, to
control their use.

FidoNet message traffic success/failure does not in any way affect BBS operation. (Unless of course of hardware or other unavoidable failure.) Failure to make a connection and transmit a packet, or errors during incoming packets,

affect only the mail sent or received. In the case of transmission, the messages will not be sent, nor will charges (if any) be applied to the senders credit account.

For a paying system, the sysop must occasionally set the users credit, using the SYSOP command. If reasonably large sums are used as a minimum ($10.00 or more) this will not need to be done more than once every few months.

        Each node maintains a node list, that contains the
accessible remote systems. Systems not in the list cannot be
mailed to. The sysop controls this file, updating or
changing it with any text editor. It can only be read from
within Fido; users cannot otherwise access it, and cannot
see the actual phone numbers used to dial.

        If the bulletin board system is a public "free" one,
then probably no toll call systems will be allowed for
general use. All users credits default to zero, so that mail
will not be allowed to non-free systems. Toll call systems
can always be accessed by the sysop.

        Occasional local networks have a host that will
accept toll call mail for free; contact the Fidos in your
area to find out about this.

        There are also a number of ways to control access to
the mail area; these are described in more detail in the
other sections of this manulal.

        If a system in the node list is marked as free, then
anyone may send messages to it. Fido controls all costs, and
balances the accounts after messages are sent sucessfully.

        Each user record contains a credit and a debit.
Messages entered are charged to debit, and cannot exceed the
credit. Deleteing a message (that has not been sent yet)
entered lowers the debit by the cost of the message; the
actual cost of the message is stored within the message.

        Fido and FidoNet are both contained in one program,
run as specified by the scheduler. Fidonet is a time driven
system, and the national time slot is at 1:00 AM Pacific
Time. (4:00 AM Eastern Time) During normal Fido operation,
users can enter FidoNet messages, and during the FidoNet
time, these messages are made into packets and sent to the
right destination. The messages may be destined to any one
or more of the available remote nodes in the nodelist.

        At the predetermined time, Fidonet takes control. It
then creates mail packets, one per node, containing the
messages for each node. If there is no mail to a node, no
packet is created, and no call is made to that system.

        After the outgoing packets are made, FidoNet
alternately waits for calls and attempts to place calls.
Mail packet transfers are done on a collision detection
basis. After the first few collisions, the network
synchronizes. If there are a number of nodes to send mail
to, each one is called in turn, until all are sent, or mail
time is over. If it fails with one node, it goes to the
next, and repeats the failed one only after trying all of
the others first. All of these steps have the appropriate
error checks and limits. The list of nodes to call is the
same order in which they appear in the node list.

        In between outgoing calls (if any) FidoNet delays a
random interval, during which it waits for incoming calls.
This interval, along with the redial algorithm, synchronizes
the net after the initial collisions.

        If an incoming call is detected, it attempts

connection with it. The baud rate is determined (same as a
normal caller would in Fido) and a message to human callers
is displayed (warning them that it is accepting only other
FidoNodes) and a synchronization process is started. This
process must complete within 60 seconds, or the call is
terminated. Once synchronized, the packet transfer is made.
The receiver just stores that packet for later use, and then
disconnects. Note that at this time, Fidonet has no idea who
originated the packet it just received.

        After an incoming call is received, FidoNet calls
out immediately afterwards (assuming there are calls to be
made) since there is a high probability that the line is now
clear. This helps synchronize the network.

        To place an outgoing call, the sender dials the
number, performs the sync process mentioned above, and
transfers it's outgoing packet. (Messages to a given node
are again checked against the node list at mail time; if
they do not match, the packet is not sent, and an error is
logged.) If the transfer was successful, the destination
node number is deleted from the senders list of nodes to
call.

        The collision detect algorithm is optimized such
that during the first few minutes of mail time, there are
many collisions, after which the net synchronizes, and none
or few collisions occur. Of course assumptions have been
made on average traffic per node; for heavy (or low) traffic
nodes, the collision detect algorithm can be modified by
command line parameters, the scheduler, and the various
routing control files.

        When mail time is over, FidoNet deletes all it's
outgoing packets that were assembled, and for each one that
was sent sucessfully, marks those messages (in the mail
area) as SENT, so the originator can tell if they went out
or not. Then, the incoming packets are unassembled, and the
messages placed sequentially in the mail area. These packets
are then deleted. A complete log of all call attempts,
incoming calls, and messages sent and received are kept in
the log file.

        If any mail at all was sent, the user credits are
balanced. This is somewhat unsatisfactory, as it balances
the accounts even if the mail was not sent. This is to
prevent extremely long processing time necessary to account
for each message and user. (Users lists run upwards of 600
entries typically; on a floppy based system this would
become unworkable.)

FidoNet then terminates, and if there were no fatal errors invokes Fido for another day. Messages received are then accessible like any other message.

        First, a small glossary of terms used in this manual
to describe FidoNet and all it's permutations.

Network:
        A group of two or more Fidos, with the same net
number. A "network" witihn FidoNet implies that there is
routing between the systems, and that one of the systems is
a host.

Region:
        A geographical area of a continent, arbitrarily
chosen during the First FidoNet Conference in St. Louis.
Regions are where Fido systems that do not belong to a
network are put; it is used only for keeping track of
systems. A region has a manager like a network does.

        From Fidos point of view, a region is just like a
network except it does nto do automatic routing to the regin
host.

Node:
        One Fido, running FidoNet. It may be a host as well.

Host:
        A node that handles mail for itself as well as other
nodes. It may accept mail, and send it on to another node.
The sysop of the host system is usually the manager for that
area.

Manager:
        The person who manages the node list for a region or
network. Usually, the sysop of the zeroth node in the region
or net.

Zeroth Node
        The node that acts as host for a region or network
has two node numbers; the normal node number (1 - 32767) and
zero; zero indicates that it is the "hub" or central node
for that region or network. For a newtwork, the zeroth node
is where all mail for that network is routed to automaticall
by Fido.

Local Net:
        A group of nodes, presumably within a small
geographical area, (ie. non-toll call or cheap calls) that
consists one or more nodes and a host. Basically, a small,
self contained Network.

Local Node:
        Really the same as just "Node", but implies that it
is a node that has a host serving it.

Topology:
        Basically, if you were to draw a map showing all the
nodes and their interconections. Some of the "shapes" made

by the map are called "topologies" if they show some
recognizable pattern, such as star, ring, etc.

Traffic:
        The actual messages being sent. If you picture
messages as things that travel along the phone connection,
it makes more sense.

Bombing Run

        The technique of sending the same message to all or
many FidoNet nodes. Frequently used for advertising purposes
in the past (highly frowned upon; a sure fire way to make
enemies these days) and very useful for disseminating
information rapidly throughout the net.

Fidonet topologies

        Fido works by having "time slots", or windows, in
which it sends mail. The national window is one hour, at the
same time across the country; 1:00AM Pacific Time, 4:00 AM
Eastern Time. During this time, all Fidos send their mail to
wherever it is supposed to go. When the time is over, Fido
is run again, and callers can read the messages sent and
received.

Simple topology

        Fido's simplest layout is "amorphous", ie. a nice
name for a randomly interconnected blob. All nodes have
access to all other nodes; to send a message, a call is
placed directly to the desired other Fido. To illustrate
this topology, draw 50 dots on a piece of paper,
representing each Fido, and from each, draw a line to every
other Fido. (You will end up with a lot of lines)

        The disadvantages of this are mainly the fact that
the window is very, very crowded; it is sometimes difficult
to "get into" a busy system, as many others may be calling
at the same time.

        The advantages are mainly simplicity. New Fidos can
come into the net just by letting others know of it's
existence. A "down" system does not affect any other system.
It is very easy to set up a FidoNet, as all you have to do
is get a node number and run FidoNet.

        This is how the original FidoNet was organized
(until Sept 84 or so). It collapsed into utter confusion at
about the 50 node point.

Two Layer topology

        If you draw in all the Fidos on a map of the world,
you will find that they tend to occur in geographical areas.
St. Louis, Los Angeles, San Francisco, London, Boston, etc.
There are occasional "stragglers" by themselves.

        To alleviate the crowding problem, the new
arrangement is to break the current FidoNet system with all
it's many nodes, into local geographic areas and make those
into self-contained FidoNets, of the "amorphous" type. Mail
traffic within the local area works as it always did; if
there is mail to another local system, it is sent directly.
Each area is given a net number, to distinguish it from the
other nets and regions.

        There are two types of local organizations; nets and

regions. Nets are where a number of systems are all within a non-toll call area, ie. within basic service. Regions are just geographical areas of the country, used to logically organize a group of systems. Except for the fact that Fido does automatic routing to nets, the two are the same.

        To interconnect all of the local nets together, each local area designates one node as a "host". This node then has two node numbers; the regular one, plus 0. For local traffic, this has no effect; mail is just sent as necessary. The host is only used to access nodes outside the local area. This is completely automatic and transparent to users.

        In other words, net and node numbers work the same as ATT's area codes. Systems in a limited area all share the sam enet number (area code), and have unique node numbers. Systems in another network may have the same node number, but will have a different nuet number, just as the seven digits in your home phone number may be the same as someone elses on the other side of the continent.

        When a person on one net enters a message for a person on another net, at mail time Fido routes that message for effecient use of time. All messages for nodes within a given network are packaged into one packet, and sent to the zeroth node in that net, not directly to the desired net. Later, the zeroth node (the host) unpacks the messages, and passes them on to the desired destination.

        The advantages of this are mostly effeciency and low cost. Since all outgoing messages are bundled together, costs go down because more messages are sent in a single call (by way of the phone companies one minute minimum and cheaper rates for longer connections) and because messages from two nodes to another single node are sent at the same time. The traffic in each of the time slots is lessened, because of the smaller number of nodes using a particular window.

        The disadvantages are that if a host goes down, all local nodes do not send nor receive mail that cycle. However, the mail will be again attempted when the next time slot comes around.

        The "stragglers" mentioned above just connect to the net through the national window, and act like hosts and nodes by themselves, because Fido does not do this automatic routing for regions; only nets.

        This is the defacto standard organization designed

at the First FidoNet Conference, and implemented on June
11th, 1985. Who knows what it will be by the time you read
this.

        This method has both advantages and disadvantages,
as you may have already guessed. Most nodes (even local
ones) still run the national window; this means that if you
want, you can send mail directly. This is still necessary,
because File Attach (logically attaching MSDOS disk files to
messages) subverts the automatic routing, and send them

directly.

        Regardless of the configuration, any node may access
any other node during the national time window.

        As mentioned before, Fidonet uses net numbers and
node numbers. How this is done is described in detail here.

        To send a message to another system in FidoNet, you
need to know the other systems net/node number. There are a
number of ways to determine this.

        For a telephone system type analogy, net numbers can
be thought of as the Area Code, and the node number as the
Phone Number. To place a call (or send a message) within
your own area code (net) you do not need to specify the net
number; the node number will do. For an out of state (to
another net) message, you pick the net first.

        Unlike the phone company, you can either enter the
net and node numbers directly, or let Fido give you a list
from which you can pick one or more. Fido will let you enter
more that one at a time as well, so that the message you
enter can be sent to a number of places at the same time!

        When entering a FidoNet message, you see the prompt:

        Currently NET #125
        ?=Help, #=List NODES /=List NETS

        This says that you are in Net #125. Entering / gives
you a list of networks and regions; this is like getting a
list of phonebooks. If you pick another Net, then the prompt
will tell you the one you have selected.

        Entering # lists the nets within the net; this is
like reading the phone book, only a lot smaller. You then

enter the node number.

There is also a shorthand for all these steps, and the syntax will be a familiar one by the end of the manual:
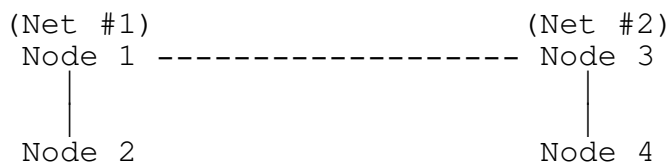
NET/NODE

For instance, you could go through the lists and enter 125 for the net, and 1 for the node, you just enter "125/1" at the first prompt. This is obviously handy when you send mail to the same node frequently.

At any time Fido expects a node number, you can enter this shorthand format. This includes the routing control files.

Routing is accomplished by the router and the scheduler. The scheduler controls when to run FidoNet (ie. when to send and receive mail) and the router controls where to send the mail. Fido does most routing automatically, but you can also completely control routing with route control files.

As an example, here is a very simple system, consisting of four nodes, two of which are acting as hosts.

```
        (Net #1)                        (Net #2)
       Node 1 ------------------- Node 3
          |                           |
          |                           |
       Node 2                      Node 4
```

The interconnecting lines are phone calls. (While it is still true that any node can call any other, that does not use the routing.) Assume that Node 1 to Node 2 is a local call (ie. no charge) and 3 and 4 also. Node 1 to Node 3 is a toll call.

Someone at node 2 wants to send a message to someone on Node 4. The message is entered normally; Node 4 is chosen for the message. (Entering mail is covered elsewhere) This is identical to previous FidoNet mail.

When Fidonet is run (because the scheduler says so)
FidoNet determines what mail to send, and to where. It
creates packets of mail, places phone calls, and sends them
out.

Without the router, Node 2 would call Node 4
directly, and send the mail. The router however, tells it to
send it to Node 1 instead. Once it is sent, that is the last
that is heard of it as far as Node 2 is concerned.

When Node 1 runs FidoNet, it also makes the packets
as do all other Fidos. However, it's router tells it to send
all mail destined for Node 4 to Node 3 instead. (Since in
this case Node 3 is a different net, and Fido need not be
told to do it; it's done automatically but you could tell it
to manually.)

Similarly Node 3 sends mail to Node 4. Since the
message has reached it's destination, it is forwarded no
longer, and stops. Node 4 has received mail from Node 2.

Even though this discussion is about mail in only
one direction, it can actually be going in both (or any, or
all) directions simultaneously, and in practice does.

Mail that is "in transit" disappears when the
message has finally be forwarded.

Previous Fidos had only one route file, called
ROUTE.BBS, and all it did was control routing. Route files
now control many different parameters for packet and mail
operation.

Instead of just one file, there can be one per
schedule. The Schedule Tag also reflects the name of the
route file used:

        Schedule A   ROUTE.A
        Schedule B   ROUTE.B
        ...   ...
        Schedule W   ROUTE.W

If the specified route file cannot be found, then it
looks for ROUTE.BBS; if that isn't found, one is not used.

IMPORTANT NOTE:

Schedule A is a very special schedule. Fido normally

will not send any mail unless you very specifically tell it
to do so; for this reason, Schedule A was made to simplify
life for sysops not part of a local net.

        Schedule A should be used only if your only schedule
is the single, national FidoNet event. If you run any kind
of routing or network, please do not use A.

        Schedule A basically basically executes the
following routing statement without the need for a route
file:

SEND-TO ALL

        If you need to do fancy routing from Fido to Fido,
you will need to set up schedules for each leg of the
journey, as covered above. It is recommended that you make
each schedule a seperate tag, so that you can have a route
file to control each.

        The best way (well, the only way, since this is how
I'm going to cover it all) to get an idea of the
capabilities of the route language is to read about each
statement.

        The route file contains an ASCII text "language",
that is more or less format free. There are a number of
keywords, that take a variable number of arguments. The
commands are:

                ROUTE-TO
                SEND-TO
                ACCEPT-FROM
                NET
                SEND-ONLY
                RECV-ONLY
                EXTERNAL-MAIL
                PICKUP
                POLL

NO-ROUTE
                    HOLD

        The command modifiers and metanumbers are:

                    ALL
                    OURNET
                    HOSTS
                    NOT

        Arguments to commands are always numbers or
metanumbers; the numbers specify node and net numbers.
"Metanumbers" represent useful groups of node and net
numbers.

        Route file contents are completely free form;
keywords can appear anywhere, in any order. Leading and
trailing spaces, commas or tabs are ignored, though you need
at least one between each keyword and number.

        You can use the same keyword more than once. The
following two blocks of code accomplish the same thing:

                    SEND-TO 99
                    SEND-TO 1,77

                    SEND-TO 99,1,77

        Fido will let you put in contradictory commands; the
last one encountered will have effect. The two lines below
are treated as if the first one does not exist.

                    ROUTE-TO 99,1
                    ROUTE-TO 1000,1

        Comments

        Comments are considered to be anything starting with

Fido's Complete Operating Manual   Page FIDONET-20
Route File Contents and Syntax

a semicolon and continuing to the end of the line. Comments
can be put anywhere in the route file.

        Node Numbers

        Most route file commands require numerical
arguments, usually a list of nodes. There are a few ways you
can specify this; use which ever looks nicer or is shorter.

99   a plain number

        A number by itself means a node number within your
current network or region. For example, if your net is #125,
99 means node 99 in net 125.

            125/99   net/node

        This very explicitly means net 125 node 99; if your
net were not 125, then you need to specify the net and this
is one way to do it.

        Another way is to change the "default" net; see the
NET command description.

        For all route file commands, you can enter any
number of node or net/node numbers, either on the same line
or continuing onto the next. For example, the following are
all OK:

            SEND-TO 99
            SEND-TO 129/99
            SEND-TO 1,33,76,55
              5,33,77/456,   5/23,5/24
            SEND-TO
            1
            99

            Metanumbers

        While you can, and frequently need to, specify long
lists of numbers, you usually do this in ordered sets, such
as member of your own net, all nodes, or all nodes except
your own net, and so on. Fido supports some special
"metanumbers" to accomodate this.

ALL
        Means "every node in the node list"

OURNET
        Means "every node in out network or region".

HOSTS
        Means "all network and region hosts". (All node
number 0's, the zeroth node for each net and region.)

        To extend the above example, you could use:

```
        SEND-TO ALL
        SEND-TO OURNET
        SEND-TO HOSTS
```

NOT

        "Undo" the command for the list of nodes to follow.
The following example should explain it's main side effect:

        SEND-TO ALL NOT OURNET

        If your net is 125, then this means SEND-TO all
nodes except those that are in net 125. It is not exactly
the same as the word "except"; Fido does not use the word a
way a person does. This is a very subtle difference, and
exactly how people can get burned by computers.

        Extremely literally, the above statement means:

        "Mark ALL nodes as SEND-TO, then remove the marks
from nodes in OURNET".

        Using NOT with the SEND-TO command "undoes" the
effect of the SEND-TO command; it is like saying:

        SEND-TO ALL
        (UN-SEND-TO OURNET)

        There is no UN-SEND-TO command of course. Fido scans
the line from left to right, first marking all nodes as
SEND-TO, then it goes back and removes the marks in OURNET
nodes.

        The only time this will matter is if you had done
something like:

        SEND-TO 99
        SEND-TO ALL NOT OURNET

        The first line says to send to 99, in our net. The
second line does what you think it does, EXCEPT that NOT
OURNET means "undo" all of OURNET, including the 99 from the
previous command.

        I may be overemphasizing the importance of this, but
better warned than not.

ROUTE-TO is the most basic and powerful routing
command. It modifies the destination of the basic message
unit. It has the syntax:

        ROUTE-TO <dest-node> <node, node, node, ...>

        Which means in English "Route to dest-node, messages
for node, node, node ...".

        Please keep in mind that internally, Fido will route
all mail for nodes inanother net, to the host (node 0) of
that net. This could be done via a ROUTE-TO statement (and
in previous versions was) but is so universally necessary
that it was made implicit. You can use ROUTE-TO to override
Fido's default host routing.

        The list of nodes can be replaced with either ALL or
OURNET; for an outgoing host, you could specify ROUTE-TO
<host> ALL, which means send to the host node all outgoing
mail.

;
;Send all of our outgoing mail to fido 99. Fido 99
;makes all long distance phone calls.
;
ROUTE-TO 99 ALL

;
;Route mail for nodes 100/51 and 100/22 to 125/99 instead.
;
ROUTE-TO 125/99 100/51,100/22

HINT:
        There are certain things that can override any
routing that may be in effect. The following things cause a
message to go directly to it's final destination rather than
to the host or ROUTE-TO node.

        File attach
        NO-ROUTE <nodes>
        POLL <nodes>

        SEND-TO tells FidoNet that during this schedule,
whatever it is, that it is OK to send mail to the list of
nodes that follow. If none are specified, Fido will not send
any outgoing mail.

        The exception to this is Schedule A, which
implicitly allows mail to all nodes; this is done for
convenience, and historic backwards compatibility.

        In a multiple schedule system, you may set aside a
schedule to send mail to and from your local nodes, say 1/2
hour before or after than national time. Since you only want
to deal with your local nodes, the statement:

SEND-TO OUTNET

        Tells Fido to send mail to only the local systems,
as defined in the node list. You could of course specify any
legal node number, including ALL.

        If you do not specify a SEND-TO command, then Fido
will not send any mail! Most Fidos run the national mail
schedule using:
;
;Normal national schedule route file.
;
SEND-TO ALL
;
;We use ROBOT.EXE to generate messages to all the
;hosts in FidoNet (net/0 and region/0) and now we want
;to send to them.
;
SEND-TO HOSTS

          ACCEPT-FROM is another basic command, and is used
for cost control and security purposes. It controls whose
mail your Fido will forward onward.

          Normally, your Fido system will send only mail that
originated on your system; this is to prevent another system
from accidentally forwarding it's long distance mail to you
for further mailing.

          There are many cases where you do want to forward
others' mail. If you are a host for a number of local
systems, you need to accept mail from anyone, for forwarding
to your locals only. The statements:
```
;
;We are an incoming host, and receive all mail for the
;other nodes in our net. We need to pass this on to them,
;and only them!
;
SEND-TO OURNET            ;our net only
ACCEPT-FROM ALL           ;but from anyone
```

          During a local schedule tells Fido to send mail to
only the local systems, and to accept it from anyone. If you
were an outgoing host for your locals (you take their
outgong mail and send it around the world) you could do that
by:
```
;
;Accept mail from only our local nodes, for mailing anywhere
;in the world.
;
ACCEPT-FROM OURNET        ;from our own net only
SEND-TO ALL               ;but send to anywhere
```


NOTE:
          ACCEPT-FROM does not mean that Fido will reject
messages from any other Fido; it means that your Fido will
not accept them for FURTHER MAILING.

        For all routing commands, Fido assumes, unless told
otherwise, that the network is your Fido's network. That is,
if you specify a node number without a net number, as in:

ROUTE-TO 99,1,2,3,5,77,88,101 ...

        Fido assumes all of those nodes are in your net. The
NET command changes the default network for further routing
commands. For example, if you were part of net 125, and ou
wanted to specify a number of nodes in another net, the
following two are equivalent:

ROUTE-TO 99,101/1,101/2,101/3,101/4 ...

NET 101
ROUTE-TO 125/99,1,2,3,4 ...

        Remember that Fido will remember the last set NET;
use NET 125 to change it back if necessary.

        You may be able to guess at the purpose of these
two. They do not control how Fido routes, but how it dials.

        SEND-ONLY tells Fido not to pause waiting for
incoming calls, but to dial as quickly as possible. This is
a very special purpose command, only for outgoing-only
systems used as outgoing hosts.

        For example, in the SF Bay Net, there is an unlisted
Fido that does outgoing mail only; since it is unlisted,
there are no other Fidos trying to call it.

        RECV-ONLY does just the opposite. Fido makes all of
it's outgoing packets normally, but never makes any outgoing
calls. this command is useful for incoming hosts that want
to stay maximally available, or for systems the rely on
other systems using PICKUP or POLL to retrieve their mail.

```
;
;We never send mail during this schedule, only receive
;it. Fido 99 sends it. So sit and receive only.
;
RECV-ONLY
;
;Fido 1 passes all it's outgoing mail to us, Fido 99.
;Since its an unlisted number, no ne ever calls us; do
;nothing but make outgoing phone calls.
;
SEND-ONLY
```

        This is another very special purpose command. It is
meant for use as a gateway from Fido to different networking
system.

        All it does it cause Fido to not delete it's packet
files and file list files when a schedule ends. Normally,
after Fido finishes a schedule, it deletes any unsent
packets and lists.

        Using EXTERNAL-MAIL, you can coerce Fido into
creating it's packets and file lists in a coherent fashion,
and leave them for you to handle. The way to do this is to
make a route file that contains any routing you need, and
the EXTERNAL-MAIL command, and make the schedule only ONE
MINUTE LONG. (If it is longer, Fido will still attempt to
make outgoing calls!)

        When EXTERNAL-MAIL is specified, Fido will treat all
the packetized messages AS IF THEY WERE SENT; messages are
marked (SENT) or deleted, as appropriate.

        It is assumed that you will deal with the packet
files yourself.

        There is a fixed naming format to these files. For

each node that there is mail for, there are two files:

        nnnnmmmm.OUT   Message Packet
        nnnnmmmm.FLO   File List

        The .OUT file is a FidoNet Packet, as defined in
FIDOMAIL.DOC, and contains all the outgoing messages.

        The .FLO file is a list containing all of the
pathnames of any files that were attached by messages in the
a variable length null terminated string.

        The net/node number is the filename portion. Both
net and node numbers are expressed as zero filed hexadecimal
numbers:

        nnnnmmmm = net/node
        007D0001 = 125/1

        PICKUP is a command that lets one system "pick up"
it's own mail when it calls another system. For example, if
two seperate systems have mail for each other, normally Fido
requires that each system make an outgoing call to send it's
packet to the other. (This is done in the interest of
accountability; it totall yeliminates arguments ...)

        If PICKUP is specified, in the above example who
ever called first would transfer it's packet as it normally
would, then additionally ask the system it just called for
if there is any mail to "pick up" while it's connected. If
there is, the calling system receives the packet that
normally would have been delivered by the other system
making an outgoing call.

        Instead of relying on chance determining who calls
out first, the RECV-ONLY command can be used on one system.

        The syntax for PICKUP is:

PICKUP n1, n2, n3, n4, n5, ...

        The calling system requests to pick up mail only if
commanded to do so; the receiving system will allways let a
calling system pickup it's own mail.

        Note that routing will affect this command in a very
important way. If you desire to PICKUP mail for say, 100/99,
your FIdo has to actually call that system, not the host
(100/) that it will normally. It is your responsibility to
ensure that routing is setup properly.

        Please refer also to the POLL command, which has a
similar function.

        If you want to send mail normally, but leave one
nodes mailfor it to pickup, refer to the HOLD command.
;
;I have a WATS line, so why not pick up mail when I
;make outgoing phone calls. (I wish)
;
PICKUP ALL

Fido's Complete Operating Manual   Page FIDONET-29
POLL

        POLL is used to force a node to make outgoing calls
to the specified nodes even if there i sno mail to send. The
purpose of this is to support polling systems that may have
mail for pickup.

        POLL is similar to PICKUP, in that it tells FidoNet
to request pickup mail from the specified system; POLL
additionally causes FidoNet to place a call and transfer an
empty packet even if there are no messages to send.

        POLL overrides any routing in effect for the listed
nodes.

The syntax for POLL is the same as PICKUP:

POLL n1, n2, n3, n4, ...

Call each of the specified nodes even if there is no mail for them, and request any pickup mail.

One use of this might be where you have a centrally located system acting as host, and a numebr of locals. To make billing easier, you can arrange it so that the locals never make phone calls; the host calls them in turn and picks up it's own mail.

```
;
;Host: call all of our nodes and pick up the mail.
;
POLL OURNET

;
;Local: make packets, but do not send them; let the host
;call and get it.
;
ROUTE-TO 99 ALL
RECV-ONLY
```

Refer to the HOLD command for more ways to hold mail for another system to pick up.

This command causes Fido to ignore any routing specified for the list of nodes specified. This command is necessary in many cases to make PICKUP work as desired.

The syntax for NO-ROUTE is:

NO-ROUTE n1, n2, n3 ...

If you have mail for a number of nodes waiting to do
PICKUP, make sure you specify them all as NO-ROUTE. The
reason is that if the nodes are in another net, then Fido
will address the packet to the host, not the nodes
themselves; when a node calls in, it will find no mail,
because it is addressed to their host!

        NO-ROUTE tells Fido to not do routing for the
specified nodes.

;
;Fido's 100/22 and 100/51 call in for their mail; make
;sure that Fido doesnt address it to 100/0, their host.
;
NO-ROUTE 100/22, 100/51

        The HOLD command causes Fido to not make outgoing
phone calls for the nodes listed. It's main use is so that

another system may call in using PICKUP or POLL and retrieve
their own mail, by preventing Fido from calling that system
to deliver mail.

        An example might be where you want to deliver most
mail normally (let Fido packet it and make phone calls)
except for one system; for that system, you want Fido to
wait for the other node to come and pick it up.

        The syntax for the HOLD command is:

        HOLD n, n2, n3, n4 ...

        For instance, you could have your system send it's
mail out as would any other Fido, except make one system
call it to get it's mail.
```
;
;Normal outgoinging mail, but make 100/111 call in
;and pick up its mail.
;
SEND-TO ALL             ;OK to send to all,
NO-ROUTE 100/111        ;no routing for this one,
HOLD 100/111            ;hold it
```

        I feel the best way to describe routing commands is
by example. The example here is for a network, it's incoming
host and local nodes. The host receives mail from the rest
of the network and passes it on to the locals. There are two
schedules for the host:

        Schedule B   1:00 - 1:50
        Schedule C   1:50 - 2:00

        Note that this schedule accomplishes all local
system routing within the national time slot. It does not
need any further schedules, but does assume that the host to
local traffic is fairly small.

        The locals just run one schedule, during the normal
national mail time. The schedule tag is arbitrary; it can be
anything A to W. A is handy in this case because it does not
need a route list.

```
;
;Schedule B route file for the host; just do
;normal FidoNet mail, as most of the systems in the
;world. Since there is no ACCEPT-FROM, it will only
;send mail actually entered on this system.
;
SEND-TO ALL                 ;we can send to anyone
```

```
;
;Schedule C route file for the host; the locals are still
;running schedule A, and are waiting for mail, and maybe
;sending some of their own. This passes any mail received
;during schedule B to the locals.
;
SEND-TO OURNET
ACCEPT-FROM ALL
```

        Fido processes the route files sequentially; you can
take advantage of this to make it easier to create custom
routing. For example, if you wish tou route all mail to a
certain system except for one or two, it is not necessary to
list them individually merely to exclude the one or two. You
can do it by:

        ROUTE-TO 99 ALL
        ROUTE-TO 500 500

        With the esception of POLL and NO-ROUTE, commands
can overwrite the effect of any previous command. The above
effect could be had by:

        ROUTE-TO 99 ALL
        NO-ROUTE 500

     The scheduler is the way that Fido controls when, as
opposed to how, FidoNet (and other things) are run. The
details of operating the scheduler commands is described in
the Sysop Only Commands section. This section tells how to
use the events.

     The scheduler contains a list of "events", each of
which consists of a day and time and other information on
what to do at that time. While Fido is waiting for a caller,
it is also watching the scheduler.

     When Fido accepts a caller, it watches the time until
the next scheduled event; if it is less than that callers
time limit, it sets their limit to be the time until the
event, and warns the user that they have less time than
normal. Fido will enforce this limit to ensure that the
caller logs off so that the event can be run.

     The scheduler keeps information on what is to be done
to execute the event. The schedule Tag is what determines
what kind of event it might be.

     Tags A to W are reserved as FidoNet schedules; when an
event tag A to W is found, FidoNet is run. This tag controls
which nodes in the list mail is sent to; this allows sending
mail to certain other systems during certain times, by
setting a schedule with each desired tag.

     This example covers making Fido run FidoNet in the
national time slot. The time of day depends on your
location, that is your time zone. The example below assumes
Pacific Time; adjust accordingly.

          Pacific Time   Central Time   Eastern Time
           01:00 AM   03:00 AM   04:00 AM
           60 minutes   60 minutes   60 minutes

      Do two things: Set Fido to run FidoNet every day at
1:00AM for one hour (normal FidoNet operation) and allow the
Yell command to be used only between the hours of 9:00AM and
5:00PM.

(Enter command '6')
Event # [0 - 34]: 0   (over write the default one)
Day of week [All, 0=Sun ...]: A (every day)
Start Time [hh:mm]: 9   (9AM)
Window Width: 480   (8 hrs)

Schedule Tag: [A - W, X, Y]: Y   (Yell command)
(Yell command now enabled from 9:00 AM to 5:00 PM)

(Enter command '6')
Event # [0 - 34]: 1   (New event)

Using the Scheduler

Day of Week [All, ...] A   (every day)
Start time [hh:mm]: 1:00   (at 1 AM)
Window Width: 60   (1 hour)
Schedule Tag [A - W, X, Y]: A   (Normal FidoNet)

        That is it. It was not necessary to choose events 0 and
1; any two could have been used. Using 0, however, replaced
the default yell command enable (10:00 AM til 10:00 PM) with
the desired one.

        To add other events, just choose another event number
and enter them as above. There is room for up to 35 events.

External Events

        The scheduler can be used to cause Fido to terminate
to DOS (to the RUNBBS.BAT batch file, actually) with an
ERRORLEVEL value that can be used to run programs at certain
times of the day.

        This isnt a general purpose way to run programs like
data based programs, etc. It is meant for use with programs
that can run unattended, such as Fido's RENUM.EXE utility
program.

        You set a schedule normally, except select X for the
tag. When you do this, it asks an additional question:

ERRORLEVEL [4 - 255]:

        This is the ERRORLEVEL passed back to DOS. This
value can be tested for in the batch file by:

        IF ERRORLEVEL (xx) GOTO LABEL

        Where (xx) is the errorlevel you specify, and LABEL
is a label in the batch file.

        You need to understand how MSDOS batch files work;
it is too complex a subject to cover here. One reminder
though: the IF ERRORLEVEL (xx) GOTO LABEL does not mean "if
errorlevel EQUALS (xx) goto label", it means "if errorlevel
IS EQUAL TO OR GREATER THAN (xx) goto label". This means
that you must put the tests in decreasing order:

```
FIDO_XXX switches ...
if errorlevel 10 goto x1
if errorlevel 4 goto x2
if errorlevel 1 goto end
```

        The format of the node list has been expanded in
version 11 of Fido, over previous versions.

        The node list is the list of remote systems, the phone
numbers, etc. When you apply for a permanent node number,
you will receive the latest node list. Node lists will be
broadcast to all FidoNodes as it is updated.

        Most of this information is academic; the node list
is too large, and too complicated, to manage manually. The
LISTGEN program, by John Warren (see seperate documentation)
was written specifically to handle the processing needed to
customize the node for your specific area code, modem, dial
requirements, etc.

        As of August 1985, the nodelist is published weekly
by Fido 100/51. Please refer to the current lists, or
contact a local Fido system to obtain the latest node list.
The nodelist distribution process may change without notice.

        The node list, called NODELIST.BBS, describes each
remote system, one per line. You do not need to create one;
new ones are distributed on a regular basis, and they are
easy to find if you lose one.

        Note that each component of the nodelist cannot
contain spaces or commans; spaces and commas are used to
seperate fields. Therefore, use the underscore character to
seperate words in a given field. For example, to put in a
name of "Fido's Board" do it as "Fido's_Board". See the node
below about special text processing done on the node list
lines.

        The basic format for a node list entry is as
follows:

(KEY) 1 0 1200 Board_Name (415)-864-1418 SF_CA 24_hrs_etc

```
          ^       ^ ^ ^       ^                       ^           ^
          |       | | |       |                       |           |
          |       | | |       |                       |           +------ Cmt
          |       | | |       |                       +------------ City
          |       | | |       +--------------------------- Phone
          |       | | +------------------------------------ Name
          |       | +------------------------------------- Baud
          |       +--------------------------------------- Cost
          |       +----------------------------------------- Node
          +------------------------------------------------- *text
```

        Each of these items are seperated by one or more
spaces or tabs. Do NOT have spaces, commas or tabs in the
middle of any of these.

Key

        This word, if present, tells Fido what kind of line
this is. Currently, there are three possibilities:

(blank)
        If no word is present, then this line defines a
normal Fido node. The number field is the systems node
number.

NODELIST
        This doesn't define a node, but is used to identify
the nodelist itself. Fido will display the LIST entry when
listing the nodelist when entering messages.

        In this case, the node number field is the same as
the extention of the filename itself; the day of the year in
which the file was created. (000 for 1 Jan, etc) This number
cannot be zero. All other fields in the node list entry can
have any contents.

HOST
        This line defines a Network Host. The number is this
networks Net Number. All nodes that follow this line are
considered part of this Net until another HOST line is
encountered.

REGION
        Same as HOST, but has different meanings to Fido;
refer to earlier chapters.

COUNTRY
        This word is reserved; COUNTRY entries are otherwise

treated as HOST for all FidoNet purposes.

Number

     This is that systems net or node number, depending on
the key field. It cannot be zero.

Cost

     This is the cost, in cents per message, that you want
to charge a user for sending mail to this node. If the cost
is 0, it is assumed to be a local call. If 0, then anyone
may send messages to this node. Otherwise (if not zero) the
user must have enough credit in their user record, or Fido
will not allow them to enter mail. (Set by the SYSOP.EXE
Utility program.)

Baud Rate

     This is the speed to dial this system with when it is
time to send them mail. It is any number that is a multiple
of 300; i.e. 300, 600, 1200 ... 9600. Fido will attemt to

dial at this baud rate even if your modem cannot handle it.

Name

     This is just the descriptive name (mnemonic) for this
system. Luckily, most systems have some cutesy name, like
Fido's Board, Demon, etc. It must be one word. (Use
underlines.) It must also be less than 25 characters long
total.

          Please see the note below about special text
processing for this entry.

Phone

     This is the phone number for the system. It is
basically a standard D.C. Hayes SmartModem phone number,
with one exception. If you use Sprint, etc, you must put it
in each number. Do not leave spaces in the number. The legal
characters in a phone number are:

     1,2,3,4,5,6,7,8,9,0 # *
     -              ignored
     ( )   ignored
     .   1 second delay
     ?   Wait for tone
     @   Wait for quiet

_   See note below

      If a function is not supported by your modem, the
command is ignored or converted into the closest command,
usually a delay. Note that comma is not the delay character;
it is used to seperate items in lists. A dot is translated
to a comma when dialing.

      T   Touch Tone
      P   Pulse

      May be inserted at any point in the number as needed.
Fidonet dials in Touch Tone by default, if P is not
specified. The maximum length of the phone number is 39
characters.

City

      This is another single word field, usually the city and
state. Line the name field, underscores are converted by
Fido into spaces. See the note below on how text is
proccessed for this entry.

Comment

      The rest of the nodelist entry is considered to be
part of the comment field. For future compatibility, all
comments should be done as other fields.

NOTE:
      Underscore characters found in nodelist lines are
converted to spaces after processing. This allows a neater
looking nodelist. Underscores can be used in the Name, City,
Phone or Comment fields.

        Assuming you already have Fido running, setting up
FidoNet is simple. If you dont have Fido running, then skip
this part until you do.

        The basic steps to do are:

(1)   Set the basic mail parameters shared by Fido and
        FidoNet.

(2)   Set up the MAIL message and file area.

(3)   Make up a working node list (list of other systems.)

(4)   Set the scheduler to run FidoNet.

(5) If you have a host system, make a route list.

(6) Obtain a node number.

    That is all there is to it. Each step is described
below. Also, I can't emphasize how important it is to ...

    HAVE THE TIME AND DATE SET ACCURATELY!!!!!!!!

    Fidonet depends heavily on the correct time and date.
Within 5 minutes or less is fine, its not that critical, but
since this is a time driven system, it is very important.

    Assuming you already have Fido running completely,
pick or create a message area for FidoNet messages. This can
be an existing message area, or a totally new one. Install
it as you would a regular Fido message area, then:

    1 <number>   Select your mail area,
    1 1\A   Set MAIL attribute
    1 S   Save the changes

    This informs Fido that entering messages in this
area should prompt for net/node numbers, change prompts, and
other small internal differences.

    The "4" command is used to set the net and node
numbers, the message path and the file path used for
attached files. The message path is the duplicate of what
you entered for the FidoNet Message Area; the file path is
where Fido will place any files coming in over the FidoNet.
Please do not leave it blank! If this path is blank, Fido
will put files in the default Fido area, and could overwrite
files such as the user list, bulletins, etc.

    It's a good idea to make the file path the same as
your upload path, or possibly an area of it's own. It does
not have to be a Fido File Area at all; just an MSDOS
directory or disk.

    4 ?   help!
    4   lists current settings

    4 msgpath\P   Set Message Path
    4 filepath\F   Set File Path

    If you have a net and node number, set it now:

```
        4 nn\T              Set Net Number
        4 nn\N              Set Node Number
```

        If you are a host, or belong to another network, set
your alternate net and node to that, otherwise make it the
same as your main net and node numbers:

```
        4 nn\H              Set Alternate Node Number
        4 nn\A              Set Alternate Net Number
```


        The last item is to set the time to run FidoNet with
the Scheduler. The national FidoNet Schedule is based upon
1:00 AM Pacific Time, either Standard or Daylight Savings.
You will have to adjust this of course for your local time
zone.

        Follow the <u>Using the Scheduler</u> section, and set the
time.

Table of Contents

        If you are really lucky, this manual will help you
create and customize a Fido Bulletin Board system to your
liking. If you are not lucky, you will become hopelessly
lost and confused.

        This manual covers installing and operating the
software itself. If you are attempting to become part of the
world wide FidoNet, please contact a FidoNet sysop who is
currently in the net. The problem is that since it's
inception in May 1984, the basic net organization has
changed radically, every few months, and all attempts at
putting down information in a such slowly changing text such
as this is hopelessly outdated by the time it reaches your
hands.

        To get information on the current FidoNet, contact
Fido 125/1 (formerly known as Fido #1) and look for the
latest Fido History document, or other related files.

        As an example of the problems involved, when the
Third Edition of this manual was distributed in November
1984, there were 50 or so FidoNet nodes in the US, in a
simple amorphous network; today (Aug 85) there are over 350
systems in the US, Canada, England, Sweden, Holland and
France, in a multitiered tree shaped network. So much for
putting dynamic information is a static place as this.

        Fido is a software system, not just a single
program. Besides the program itself, Fido requires a number
of disk files, subdirectories or other disks to hold
messages and downloadable files, and obviously a modem and
cable.

        While Fido requires many disk files to run properly,
almost all of them are created and maintained by Fido
itself. The exceptions are the batch file to actually run
Fido, the welcome messages and other text messages that Fido
can display, and various other lists and messages. These are
covered in detail later, and are simple to create manually,
if need be, with an ordinary text editor.

        You can think of Fido as having an <u>inside</u> and an
<u>outside</u>. The <u>inside</u> is what you can change and see by

running the Fido program. These are things such as setting up message and file areas, privilege levels for the various commands, scheduled events such as FidoNet, and the like. These are all accessed by Fido commands.

The <u>outside</u> consists of the disks used, the modem and cable, the MSDOS batch file, subdirectories for the message and file areas, signon and welcome messages that the system operator creates, and so on. While these are of course used by Fido (messages are displayed, modem is used to answer and dial, etc) they cannot be created or modified

from within Fido.

Simply put, there are four stages to follow to get Fido properly installed. These are:

(1)   <u>The</u> <u>Environment</u>:

Making subdirectories, formatting blank diskettes if necessary, determining how many and where to put the message and file areas.

(2)   <u>Inside</u>:

Running Fido from the local keyboard (instead of the modem) in order to set up Fido to match the environment you created. Fido must be told about each message and file area subdirectory or diskette set up earlier.

(3)   <u>Outside</u>:

Customizing the batch file to run Fido from; this contains the command line parameters for users time limits, and other operating details.

(4)   <u>Hardware</u>:

Making the modem go. Up to this point, the modem is not needed at all, and can be ignored. If you have a modem already connected and running with a communications program, then this step is easy.

          This part wasn't mentioned above, and really should
be done before you even start. Ask yourself some very basic
questions: do you have a machine that is capable of running
Fido? (MSDOS version 2 or later, a supported machine or IBM
clone, enough memory, etc)? Do you have all the files? A
supported modem, or reason to believe yours will work?

          Most people will not have trouble with this, but be
warned that things are not always what the computer store
salesman said they were. "Hayes Compatibility" means more
than just executes AT commands. It implies things about the
signals on the rear panel connecter (Mainly Data Terminal
Ready and Carrier Detect) and other not so obvious
capabilities. A good test here is: does it run SMARTCOM from
Hayes? Standard PCTALK?

          If you are running "vanilla" hardware, such as an
IBM XT, DOS 2.1, a Hayes 1200 and a ribbon cable, you won't
have any problem. Those of you with less popular hardware
should be very careful.

                    Points to Check

          MSDOS 2 or higher
          A supported machine
          A serial port
          192K or more memory
          A usable modem
          A usable cable
          Lots of diskettes
          All the files you need

        The first step is to figure out what you want your
system to look like. This depends on your needs, your
hardware (mainly, how much disk space do you have) and the
needs of your users.

        It is not possible to cover all the combinations
that Fido can be configured in, and the one you end up with
will probably change as your needs change. Therefore, this
manual will cover a single simple configuration; others are
set up in the same way, and once it is understood, is fairly
straightforward.

Step 1: Setting Up Message and File Areas

        If you follow this installation guide exactly, you
will end up with a Fido Bulletin Board system that has the
following characteristics:

        - A public access system

        - Two independent Message Areas

        - Two independent File Areas

        - An additional, special Message Area where users
          can find special messages from the system
          operator.

        - Ready to be used with a DC Hayes SmartModem 1200

        Any and all of this can be modified, either now (if
you are confident) or later, after you get this example one
running. While this manual is arranged in the order covered
in the introduction, you may actually perform any steps in
any order; however, this order has been found, by trial and

error, to be the simplest and easiest to follow.YOu can of
course do anything you want!

        You will note that Fido uses MSDOS subdirectories in
huge quantities. If it's a subject you aren't comfortable
with, please learn to use them before tackling Fido.
Subdirectories aren't as hard as Microsoft makes you think
with their terrible manuals.

HINT:
        All of the Fido documentation assumes you have a
working knowledge of MSDOS paths. If you don't understand
what they do and how they work, you probably won't be able
to get through this manual, nor be able to maintain your
system if you do. Paths are just not that hard, and are very
handy even if you aren't intersted in Fido.

There isn't really any difference, as far as Fido is concerned, between hard disks and floppy disks. However, Fido loves to consume disk space, and floppies aren't that generous. Before we start, some general notes on hard disks and floppies.

HINT:
If you are new to subdirectories and all that, you can make your life easier if oyu set your system prompt to tell you where you are:

PROMPT ($P) $S

Execute this manually, or put it in AUTOEXEC.BAT.

Installation on a Hard Disk System

The easiest thing to do is start at the root, and create a new subdirectory just for Fido:


```
CD \                               log into the root
MD FIDO                            make a FIDO directory
CD FIDO                            enter the FIDO directory
```

Do the above, and you will be in the newly created subdirectory. You can then ignore ALL drive letters shown in the examples below. (B:, etc)

Installation on Floppy Systems

You are going to need all the space you can get. You should format two new diskettes, and make one bootable (format it with /S, to install the system on it). This will become your A: diskette. On it, put the minimum tools you will need: some sort of editor or word processor (for making

welcome messages, etc) and the fido .EXE files. Put the other formatted diskette in B:. YOu may have to do a lot of disk swapping to get everything created right.

If your drives aren't called A: and B:, then just substitute the letters that you want.

For all systems, you should set the number of

BUFFERS in CONFIG.SYS on the booted diskette. The
performance increase will be enormous. If you do NOT do
this, then your Fido system will run very slowly, and annoy
all of your users, and generally drive you crazy. If you
know how to use CONFIG.SYS, then just do it. If not, then on
your _boot_ _disk_ do the following:

```
A:
CD \                             log into the root
COPY CON CONFIG.SYS              copy keyboard to CONFIG.SYS
BUFFERS = 50                     type this exactly,
FILES = 20                       this too
^Z                               Enter Control-Z, then return

Copied 1 file(s)                 Message from MSDOS
```

        Next time you boot, MSDOS will read CONFIG.SYS and
set the number of buffers to 50. (It will not do it until
then.)

        Anyways ...

        Before even thinking about running any programs, the
subdirectories and such need to be created. As mentioned
above, we will end up with two message areas and two file
areas.

        Message areas:

                GENERAL   Msg area #1
                TECH   Msg area #2

        File Areas:

                IBMPC   File area #1
                CPM80   File Area #2

        Special Areas:

                ANSWERS   Special Msg area

        Please remember that the names of the subdirectories
are totally arbitrary; you can make them anything you want.
If the names reflect what's in them, though, it's much
easier a few months from now.

        First, create these subdirectories using the MD or
MKDIR command:

```
B:                    on drive B: ...
MD GENERAL
MD TECH
MD IBMPC
MD CPM80
MD ANSWERS
```

        Now the subdirectories exist on your disk. These are
where Fido will put messages and files that can be accessed
by users. (Of course, there areen't any messages or files
yet, so don't worry about it.)

        Now for a tedious part. Now you must log into each
area, and make a file called DIR.BBS. This is a very small
file, and contains a one line description of what the
subdirectory is for. (No, Fido cannot read; this small
description is merely displayed to a user when they log into
an area.)

HINT:
        Make these good descriptions, as they are what the
user sees, and is all they have to tell what is in there.
Normally, Fido will display the actual subdirectory name; an
option lets you hide it, and display the DIR.BBS contents
only. This is the /Y command line switch.

        makinbg DIR.BBS can be done with your word
processor, or more simply with the COPY command.

```
CD GENERAL                        enter the GENERAL subdir
COPY CON DIR.BBS                  make DIR.BBS
General Interest Messages         one line descrption
^Z                                Control-Z, then ENTER

1 File(s) Copied                  MSDOS message to you
CD \FIDO                          log into previous subdir
```

        Repeat this for each area, entering the description
for each, instead of the "General Interest Messages" in the
example above.

        At this point, you have made all the basic
subdirectories for Fido. Changes or additions can be made
later, following the same basic steps.

        Now that the basic subdirectories have been made,
you must run Fido and tell it where to find them. There is
one last step before you do this; it is to make sure that
Fido has the minimum files it needs to run.

        While they are not absolutely necessary to run Fido,
the HELP files are handy, since you probably havent
memorized all the Fido commands yet. All that needs to be
done is to copy them into the main Fido directory. While
logged into the main Fido directory, or the root for floppy
systems, put the Fido distribution diskette into B:, and:

        COPY B:*.HLP

        You also need as a minimum the initial user list,
USER.BBS, and the initial system file, SYSTEM.BBS. You will
also need the Fido program, of course. Copy these from the
distribution diskette, to the Fido subdirectory.

HINT:
        If you can't figure out why you are having problems
with the user list, USER.BBS, or can't remember the
password, etc, refer to the SYSOP.EXE program in the Table
of Contents.

        COPY B:USER.BBS
        COPY B:SYSTEM.BBS
        COPY B:FIDO_IBM.EXE   (or _DEC for DEC, etc)

        Now, Fido has all it needs to be run. Fido has many
command line options, which are covered in a later section.
Here, you need only one:

        /T   Test Mode

FIDO_IBM /T   (or _DEC for DEC, etc)

        After a few seconds, you will see the signon
message, and a prompt for your name. You will NOT see any
nice welcome messages and so on; these are created later.

        At the prompt for your name, fill in as follows:

Your FIRST Name: SYSOP <enter>
Your LAST Name: <enter>
Wait ...
Password: WHAT <enter>

        When you enter the password, it will be displayed as
dots, so sthat someone looking over your shoulder can't see
your password. After some messages from Fido, you will be at
the MAIN Command prompt.

        There are a number of commands that do NOT show up
in the prompt. These are the sysop commands, and are the
digits 1 to 9. Only a user with SYSOP privileges may use
them. (The initial user list contains one user, named SYSOP,
whose privilege level is also SYSOP. That's what you entered
at the example above.)

        The sysop-only commands are covered in detail in
elsewhere; however the minimum needed information will be
provided here.

HINT:
        One point to remember about the numeric sysop only
commands; you can get help with most of them by entering:

        n ?

        Where n is the command number.

        Fido keeps track of message and file subdirectories
by storing them in small lists kept in diskfiles named
SYSTEMx.BBS. Each list contains one Message Area
subdirectory, one File Area subdirectory, and one HELP
subdirectory (not used here). There can be up to 99 lists.
When a user chooses a message or file area, they are really
choosing one of these lists.

        You must now store the names of the subdirectories
in these lists, so that Fido will know where to find them,
and so that users can select them.

        The "1" command is used to maintain these lists. The
lists are numbered 0 to 99; you select one by:

1 <n>   Where <n> is a number 0 - 99

        First we set area #1, using the subdirectory names
chosen before. (Only what you enter is shown here; Fido will
display the Main Command prompt and other messages also.)

1 ?   Get HELP!

1 1   Select list 1
1 B:\GENERAL\M   Set Message Area name
1 B:\IBMPC\F   Set File Download Area name
1 B:\IBMPC\U   Set File Upload Area
1 S   Save the changes

        The first set of areas is now set. You can view your
changes by entering "1" by itself.

Now repeat this process for the second set of areas.

```
1 2            Select list 2
1 B:\TECH\M    Set Message Area
```

Step 4: Setting Up Areas

```
1 B:\CPM80\F   Set File Download Area
1 B:\CPM80\U   Set File Upload Area
1 S            Save changes
```

        The third list will contain only the ANSWERS message
area, there is no corresponding File Area. This is set in
the same way as the other areas.

```
1 3            Select list 3
1 B:\ANSWERS\M Set Message Area
1 F            Set BLANK File Area
1 U            Set BLANK Upload area
1 S            Save the changes
```

        The last list to set is the "special" list, number
0. Only the Message Area is used from this list. This area
is where comments to the sysop go, using the G (GoodBye)
command. Usually (and in this example) this area will be the
same as one of the regular areas; ANSWERS in this instance.
This is so that the sysop can reply to messages left at
GoodBye, and users can read them by choosing the ANSWER
area.

```
1 0            Select list 0
1 B:\ANSWERS\M Set message area
1 F            Blank areas, as above
1 U
1 S
```

HINT:
        If you leave the message path for area #0 BLANK,
then Fido will not ask "Leave a message to the sysop? (y,n)"
at the G)oodbye command, but merely disconnect.

        At this point Fido is more or less usable in Test
Mode (/T). You can now go into the various areas, enter
messages and operate most of the commands. There will not be
any files available yet, nor any messages. You should go
through each area and make sure it is set up as you think it
is.

Message Areas

        From the MAIN Command prompt, use M to enter the

Message Section. The A)rea command displays the message
areas available, and lets you choose one. You should see
three areas, and the small descriptions you entered
previously. If not, something is wrong; go back over the
previous steps and check. Probably just an incorrectly
entered name. If all is well, pick one of the areas, for
example "1" for area #1, GENERAL.

HINT:
        Use the E command to enter a sample message. After
saving it, it should say "Saving your message". If it does

not, but says "DISK FULL: Cannot save your message" it is
either that you disk is really full (not likely at this
point!) or the subdirectory name you entered is incorrect.
You must manually create, with the MSDOS MKDIR command, the
subdirectories where Fido expects to find them.

        Return to the MAIN Command prompt, then enter the
Files Section, and check the areas there, using the A
command as before. You should see the area descriptions.
Note however, that there will not be any files for
downloading.

        The software portion of Fido/FidoNet is more or less
ready to be used. At this point, there are a number of
options. The sections you need are listed below; be sure to
browse through the others for ideas and tips.

ALL SYSTEMS:

        Fido Message Files

        How to customize the welcome messages, bulletins,
etc., in common with all Fido configurations.

        Hints and Tips

        Tips on speed, problems to watch out for, and ways
to improve overall performance. Some things you can do and
shouldnt do.

PICK ONE:

        Public System

        How to make a public access bulletin board, so that
new users can log in for themselves, and how to handle
special cases.

        Semi-Public System

        How to make a semi-private system, where new users
can log themselves in, but have restricted privileges.

Private System

        Where new users are not allowed; the sysop has to
manually enter new users. How to keep security absolute.

ALL SYSTEMS:

        Modem Installation

        Installing and configuring the modem. Many modems
are listed as installed in actual installations. Tips on how
to fix problems.

AND THEN ...

        FidoNet Installation

        Setting up your Fido as a node in the public domain
electronic mail network, and what this means.

        There are a number of text files that Fido will
display at various points; welcome messages, the initial
signon, the bulletins and so forth. You will need to
customize them to your taste, or delete them if they areen't
needed. Fido does not require the presence of any of these,
though you should probably use most of them.

        These are the ones in common with all Fido setups.
There is a section for each configuration: public access,
private, and semiprivate.

WELCOME1.BBS

        This is the first text that a user sees, after
determining baud rate by hitting the CR or ENTER key. Most
of the public Fidos use the "doggie", already provided on
the distribution diskette.

WELCOME2.BBS

        Presented after a user enters their name and
password. Just a chance to give logged in users news or
other information. Many systems run without WELCOME2.BBS.

BULLETIN.BBS

Usually where frequently changing information is posted, for all user to see. This bulletin can also be read with the B (Bulletin) command from the MAIN Command prompt.

EDTORIAL.BBS

Similar to BULLETIN.BBS, except Fido does not display it automatically. It can be accessed by the E command from the Main Section. (Note that the name is missing the "i"; EDTORIAL not EDiTORIAL.

YELL.BBS

This file is displayed to the user if they attempt a Y)ell command and yelling is not scheduled. You can use this to inform them what the Y)ell schedules are. IF you do not have this file, then Fido displays "Yelling at the sysop isn't allowed right now.".

When set up as in the previous sections, Fido is by default a public access system. However, there are special files that you can use to make life easier for you and your users.

NEWUSER1.BBS

This is displayed once for new users, just before Fido requests their city and state, and to choose a password. It should probably explain the need for a password, etc.

NEWUSER2.BBS

This is displayed right after a new user enters their city, state, and password. Usually, a simple tutorial of the system, what to expect, where to find things.

If a previously logged in user forgets their
password, after three attempts at entering a password, and
no match is found, then there are a few options.

The presence or absence of QNOPWD.BBS, a special
questionaire, determines how Fido treats new users.

OPTION 1: LEAVE A MESSAGE TO THE SYSOP

Make sure that you do not have a file named
QNOPWD.BBS; if it exists, then Option 2 will be used.

NOPWD.BBS is displayed, then they get to leave a
message to the sysop, identical to when a logged in user
uses the "G" command to leave a comment. After leaving a
comment, they are logged off. They have 10 minutes (or the
specified /S value) to complete the message.

OPTION 2: FILL OUT A QUESTIONAIRE
OPTION 3: MESSAGE THEN LOG OFF

For these two options, QNOPWD.BBS is used. This is a
special file, a questionnaire. Briefly, a questionaire
causes Fido to ask the user questions contained in it, and
records the answers (in this case) in ANOPWD.BBS. If used,
you should look at this file daily to check for new users.
You can delete it whenever necessary, Fido will create a new
one.

Alternatively, QNOPWD.BBS can contain only text to
be displayed, and no questions at all. Usually, this is the
easiest thing to do; just tell them to log on again, but use
a new name and password.

If QNOPWD.BBS exists, the user gets to fill it out
(or not) and is then logged off; they do not get to leave a
message to the sysop as in Option 1.

        If you want to run a public board, but restrict new
users, this can be easily done.

COMMAND LINE SWITCH:

        /P   New Users get DISGRACE privilege

When new users log on, they are assigned DISGRACE
privilege. Then, by using the "3" command to set the
privilege level for the Fido commands, you can restrict the
commands available to new users. For instance, they can read
messages, but not enter any, not be able to change their
name or password, access the file area or the message area,
or any other Fido command.

If you use this method, then new users have to be
upgraded manually, using the SYSOP utility program.

Other than the new users privilege level, the is
identical to a public system.

You can accomplish the same thing by leaving new
users' privilege as NORMAL, and useing the 3 command to
change the privilege levels of the commands themselves to
limit thier use.

After following the previous steps to get Fido up as
a public system, follow these to make it a private system.

A private system is one where only already existing callers can log on; new callers are prevented from entering the system. You can let new callers leave a message to the sysop, fill out a questionaire, or merely log them off, with or without a messaeg as to why.

When running a private system, callers must be entered manually, using the SYSOP utility program. Operation of SYSOP is covered elsewhere.

The presence or absence of QNOPWD.BBS, a special questionaire, determines how Fido treats new users.

COMMAND LINE SWITCH:

/M   Private System

Add this switch to the command line in RUNBBS.BAT. It prevents new users from being added to the user list.

OPTION 1: NEW USERS CAN LEAVE A MESSAGE TO THE SYSOP

Make sure that you do not have a file named QNOPWD.BBS; if it exists, then Option 2 will be used.

This is one way in which you can make new users request entry to your system; NOPWD.BBS is displayed, then they get to leave a message to the sysop, identical to when a logged in user uses the "G" command to leave a comment. After leaving a comment, they are logged off. They have 10 minutes (or the specified /S value) to complete the message.

NOPWD.BBS is just a text file, that is displayed to new users, or users that forgot their password. In it, you can explain that this is a private system, and request that they enter whatever information you require.

OPTION 2: NEW USERS FILL OUT A QUESTIONAIRE
OPTION 3: NEW USERS GET LOGGED OFF IMMEDIATELY

For these two options, QNOPWD.BBS is used. This is a special file, a questionnaire. Briefly, a questionaire causes Fido to ask the user questions contained in it, and records the answers (in this case) in ANOPWD.BBS. You should look at this file daily to check for new users. You can delete it whenever necessary, Fido will create a new one.

Alternatively, QNOPWD.BBS can contain only text to be displayed, and no questions at all. This is frequently

used to tell users that the system is totally private, and
access to it cannot be had.

        If QNOPWD.BBS exists, the user gets to fill it out
(or not) and is then logged off; they do not get to leave a
message to the sysop as in Option 1.

          To further customize Fido, you can change the
privilege level needed to operate each of the commands. This
is done with the 3 command (covered in detail in Fido's
Operating Manual).

          Entering 3 by itself lists the current settings.
Each command area has it's own privilege tables:

          Main Section,   MAINPRIV.BBS
          Message Sections,   MSGPRIV.BBS
          Mail Sections,   MAILPRIV.BBS
          File Sections,   FILEPRIV.BBS
          Read Command Prompt   READPRIV.BBS
          CHANGE Command Prompt   CHGPRIV.BBS

          For example, you can change the Raw-Display command
in the file areas to be NORMAL, and let everyone use it, or
you can make the EnterMessage command in the Message areas
privil, so that new callers (first time callers) cannot
enter messages until you up their privilege level with the
SYSOP.EXE program. (They can always leave Messages to the
Sysop at G)oodbye.)

          Briefly, to change a command privilege, enter:

          3 <x> <priv.>

          Where:

          <x>   Is the command letter, as entered normally
to Fido. (E for E)nter message, K for K)ill, etc.)

          <priv.> Is one of the Fido privilege levels below.
You MUST spell it out fully.

          TWIT   Lowest
          DISGRACE
          NORMAL
          PRIVEL
          EXTRA
          SYSOP   Highest

        There are a number of things you can do that make
Fido faster. If you have enough memory for a memory disk,
put the messages in it. This will speed things up by a
factor of ten or so. Dont worry about the .BBS files or .HLP
files; they are small and infrequently accessed. The
downloadable files are generally too large, and again,
arent accessed in a way that they speed up from a memory
disk.

        Also, you should use a large number of FILES and
BUFFERS in CONFIG.SYS. This is probably the limiting speed
factor in most systems without memory disks. Use the maximum
FILES = 20, (each "file" takes only 36 bytes, so dont worry
about that) and lots of buffers. The following is an
example:

FILES = 20
BUFFERS = 50

        More than 70 buffers or so starts to actually slow
the system down, as DOS spends all of its time fiddling
around with the buffers instead of the data inside them.

        If you run out of memory, then lower the number of
buffers. Buffers are 512 (1/2 K) bytes each on IBM PC type
machines.

HINT:
        MSDOS is not too smart about checking for enough
memory when you use BUFFERS = . The actual buffers arent
allocated until they are needed; if at that time there isnt
enough memory, then you get the message:

        "Insufficient Memory"

        And the system halts, making you reboot. Make sure
you have enough memory. 40 to 50 buffers is plenty, and will
work fine with 256K of memory.

        Use GOOD diskettes. If you get a disk error, the
user will just see the system hung, and the "... error"
"Retry Ignore Abort" message on the local console. Not good.

Initially, use the RUNBBS.BAT batch file from the distribution set of files. Modify it to include any switches needed. Later on, you can include other things for system maintenance, using the scheduler. See the scheduler section for that.

When Fido terminates to DOS, it returns an ERRORLEVEL, that can be detected from within RUNBBS.BAT. Some of the ERRORLEVEL values are built into Fido; others can be set by

the sysop. The fixed ones are:

        0   No errors or special conditions
        1   Control-C from console or stack overflow
        2   Fido Disk File error
        3   Other fatal errors

If there is some fatal error, you'd probably like to see it, and in any case there's not any sense in running again. Also, you have to type Control-C on the local console to terminate Fido (for maintainance, etc) and this causes it to terminate the batch file.

The Fido software is not prone to crashing. Most troubles will be hardware ones, or power failures and brownouts. The system cannot be crashed by callers hanging up at random points, and there is no way in which debuggers can be run, etc.

WARNING:

Beware of other programs such as NEWKEY, pop up alarm clocks, SIDE KICK, automatic screen blank programs, and anything like them. None of them are compatible with any program that makes extensive use of interrupts, such as Fido.

Most of the programs do illegal things with the hardware, and will bomb Fido, and other systems. You may never have problems; probably you will, but it may take hours or days to show up. One symptom is that the system just "stops". If in doubt, start the system up again; if it happens again, reboot again, this time leaving out any such program, and see if that cures it. It may take several days to discover.

It is recommended that you do not use such programs while running Fido.

Fido is very friendly to multitasking programs such as Multilink, and has a command line option, /I, to acocmodate them. Please refer to the Command Line Switch section for details.

There are a number of ways you can run your Files Section, if you have one at all. You can of course combine the features any way you want, but here are some common ones:

OPTION 1: FREE ACCESS UPLOAD AND DOWNLOAD

Users can upload and download files at will. New files are uploaded into the same area and added to the files list, so that they are available immediately.

OPTION 2: FREE ACCESS DOWNLOAD, UPLOAD BY SUBMISSION

Users can download any file, but newly uploaded files cannot be downloaded until the sysop allows it. There are two ways to do this.

These are accomplished with three things:

File Download Path   In the list of areas
File Upload Path   In the list of areas
FILES.BBS   The files list

The upload and download paths are accessed by the "1" command, as mentioned before. FILES.BBS is a special file in each file area, that contains the list of available files.

Before describing the Options above, here is some information:

FILES.BBS is the list in each Files area of the
files available for downloading. You must create the first
one, but it can be automatically updated in a number of ways
later.

FILES.BBS is basically just a text file, that
contains the names and a description. Fido puts in the file
size, and aligns the columns neatly. For instance:

(Sample Files display)

```
AFILE.EXE       4545 Just a sample file.
PROG.BIN  323432434 A very large file
FOO.FIZ      MISSING Another sample
```

(The actual FILES.BBS contents)

```
AFILE.EXE Just a sample file.
PROG.BIN A very large file
FOO.FIZ Another sample.
```

You must enter the names as shown; the DOS DIR
format (FILE   EXT) will not work, it must be (FILE.EXT).

Note that if a file is really missing (not in the directory)
the file size will say MISSING instead.

There are special lines within FILES.BBS, that
control how it is listed by the various commands. The
special lines are determined by the first character. They
are:

```
- This is a comment only line
  So is this (starts with a space)
@ This is the "end" of the files list.
```

The first two are merely comments; you can use them
for whatever you wish. The third is a special marker, and
tells Fido to not list anything after this line, including
this line.

When a file is uploaded, Fido asks for a one line
description for that file. The filename and description is
added to the END of the files list, above.

If there is a line beginning with @, then the new
files are added after it. Since nothing is listed beyond
this point, these files are not listed, and are esentially
unavailable until the sysop edits FILES.BBS and deletes the
line containing the @, or moves it past the last file in the
list. This lets the sysop check out uploaded files before

making them available.

<u>Uploading and Downloading</u>

        As mentioned before, Fido has an <u>upload</u> and a
<u>download</u> path. The F)iles command lists the files in the
<u>download</u> path, using the FILES.BBS list there.

        <u>Uploaded</u> files go into the <u>upload</u> path, and their
names and descriptions are added to the FILES.BBS list in
the <u>upload</u> path.

        If the <u>upload</u> and <u>download</u> paths are the SAME, then
Fido will not let an uploaded file replace (or destroy ...)
an already existing downloadable file. This prevents good
files from accidentally being destroyed or changed.

        If the <u>upload</u> and <u>download</u> paths are DIFFERENT, then
Fido will let existing files be overwritten. Usually this
means there is a seperate upload area, not accessible to
users.

        You can use the above characteristics to your
advantage. Instead of trying to describe them all, here are
three examples.

OPTION 1: FREE ACCESS UPLOAD AND DOWNLOAD

<u>Files Section Organization</u>

        Users can upload and download files at will. New
files are uploaded into the same area and added to the files
list, so that they are available immediately.

        Make the <u>upload</u> and <u>download</u> paths the same. New
files go into the download area. Make sure there is no @
line in FILES.BBS; when a new file is uploaded, it's name
and description is added to the end of the files list, where
it is immediately available for downloading.

OPTION 2: FREE ACCESS DOWNLOAD, UPLOAD BY SUBMISSION

        Users can download any file, but newly uploaded
files cannot be downloaded until the sysop allows it. There
are two ways to do this.

                VARIATION A:

        As in Option 1, make the paths the same, but put a
line beginning with @ in FILES.BBS. When a file is uploaded,
it is added to the download directory, but it's name and

description will not be displayed with the "F" command.

        This is the easiest to maintain way to do things; to
make new files available, all you have to do is delete the
line beginning with @, or move it to the end of the list.

        The disadvantage is that the file really is
accessible; it just does not show. If a user knows it's
there, it can be downloaded.

                VARIATION B:

        Make the upload and download paths different, where
the upload path is a disk or directory not available as a
File Area. It doesnt matter if there is an @ line in
FILES.BBS.

        Since new files are uploaded into a totally seperate
area, they cannot be accessed by users. (You can put the
path in a separate file area, with SYSOP privilege level, so
that you can get in there to look.)

        The disadvantage is that in order to make these
files available, you must use the DOS COPY command, and
update the FILES.BBS in the download area manually.

        This is the most secure method, is is reccomended
only you require absolute security on uploaded files.

        There are many command line options that control
Fido operation. You will have to edit these into the
batchfile that runs Fido. The switches are:

<n>/J   Set Modem Type

Selects the type of modem you are using. <n> is a number
that specifies the modem type:

            1/J   Hayes
            2/J   DEC DF03
            3/J   Racal Vadic VA212
            4/J   Novation/ IBM PC Junior

```
5/J      US Robotics
7/J      Prentice POPCOM
8/J      IBM PC Junior
9/J      Ventel MD212 with WECO option
10/J     GDC modem
11/J     Hayes 2400
12/J     US Robotics Courier
```

        See the section on modems for details.

/Y   Don't display pathnames

        Causes Fido to NOT display the subdirectory name
when listing file ans message areas with the A)rea command.
Normally, Fido displays the pathname as set with the "1"
command, followed by the contents of the DIR.BBS file.
Adding /Y to the command line causes Fido to display the
DIR.BBS contents only.

<task>/I   Set multi-task environment

        This does only two simple things: it changes the
names used for the log files, and affects the way messages
are created.

        The names for the logs will incorporate the task ID
number; ie. if you specify 9/I the log files will be:

        SYSOP9.LOG
        MAILER9.LOG

        You cannot use 0 as a number. The other effect of
setting a task ID is to force Fido to recount the messages
before saving a new message. This hopefully prevents
duplicate message numbers making a mess of things, or at
least minimizes it. It has the unfortunate side effect of
greatly slowing down saving messages.

        If you use /I in test mode, you will not be able to
see anything. Do not use /I with /T.

/T   Test Mode

        Test mode. Described early in this manual, not used
for a remote Fido system. Not a good thing to put into the
batch file.

/M   Private System

      Private message system. Please read the chapter on
private systems.

/P   Low Default User Privilege

      Set the default privilege level for new users to
DISGRACE. See the chapter on semiprivate systems for
details.

(number)/S   Log On Time Limit

      Set the time limit for logging on. This limit starts
as soon as carrier is detected, and stops after a valid
password is entered. At that point, the normal user or first
time user time limit starts. The minimum is 5 minutes; the
default is 10.

(number)/L   Normal User Time Limit

      Set the time limit per call to (number), where
(number) is the number of minutes. 0 means unlimited use,
i.e. no time limit. Must be 0 - 65535. Read a later section
on time limits. If not specified, then the limit defaults to
60 minutes.

(number)/F   First Time Users Time Limit

      Set the time limit for new users. The default here
is 60 minutes also. You can give first time callers a low
limit.

(number)/D   Daily Time Limit

      Set the total time limit for a 24 hour period. This
is usually 4 or 5 times the /L limit. This is the maximum
number of minutes a caller is allowed in a 24 hour period,
and prevents long term hogging, and keeps them from hanging
up and calling right back when the per-connect limit is

reached. The default is 300 minutes.

```
(number)/K      K byte Download Limit

        Limit the number of K bytes a caller can download in

a 24 hr. period. Along with /F, /L, and /D, allows full
control over call-in-once-download-everything
and-never-call-back type users.


<number>/V      Carrier Detect Mask Bit

        Define the bit used to determine the state of the
Carrier Detect signal from the modem. The default is CTS, or
16. For most machines, use 128/V. See the modem section in
this manual for details.


/U      No File Transfer Display

        Disables the file transfer information displayed
when a file upload or download is happening. Normally, Fido
will display status messages about each block being
transferred, and all errors that happen. This is very useful
when a caller is having trouble. Enter /U to disable this
display.


<n>/G      Set FidoNet Attempts with Connect

        Sets the number of attempts with connect for
FidoNet. The default is one. This is the number of times
that Fido/FidoNet will call a particular number, if the
modems connect but no mail is transferred.


/1      Set COM1
/2      Set COM2

        Select which physical IO device to use. The default
is /1, COM1 on the IBM PC. /2 Selects COM2. Not used on
machines other than the IBM PC, unless otherwise noted.


        These switches are entered thusly:

Fido 90/l 30/f 200/k 180/d

        90 minute time limit per call (normal users)
        30 minute time limit (first time callers)
        200K bytes/48 hrs download limit
        180 minutes time limit per user per 48 hrs
```

        Fido has a number of HELP files, that are displayed
by entering "?" at almost any point within Fido. There is a
help file for each place ? can be entered. The HELP path is
used to find the file needed. The standard Fido setup is a
single subdirectory called HELP, which contains all the help
files.

        All the help files have the extention .HLP without
exception. These are:

```
1.HLP               Sysop 1 command
2.HLP               Sysop 2 command
3.HLP               Sysop 3 command
5.HLP               Sysop 5 command
6.HLP               Sysop 6 command
8.HLP               Sysop 8 command
MAIN.HLP            Main Section command prompt
C.HLP               Within the C command, Main Section
MSG.HLP             Message Section command prompt
MQUERY.HLP          Within the R command, Message, Mail Section
ENTERCMD.HLP        Within the E command, Message Section
MAIL.HLP            FidoNet Mail Area command prompt
FILES.HLP           File Section command prompt
XFERTYPE.HLP        Within the D and U commands, File Section
NODES.HLP           At the "Fido to send to" prompt,
                    entering a FidoNet message.
```

        You can customize these if necessary, but at least
initially don't worry about them.

The modem should be installed ONLY after the
software is ready for it. There is no need for a modem while
configuring the software.

In order to test Fido and the modem, you will need a
second computer and modem. This may be difficult, but is the
only way by which to test the modem. Any modem and computer
or terminal may be used; all you require is some sort of
communications program to call your Fido with.

For IBM PC versions, you can use a command line
switch to select either COM1 or COM2. The default is COM1.
If you need to use COM2, add to your command line in
RUNBBS.BAT:

        /2   COM2

If you remember to set the modem type (/J) right,
Fido will initialize the modem for you; you need not set any
switches, except as mentioned below. If you have a modem
that is not listed, then you are on your own; if it is a
Hayes compatible, by all means try that.

Please check that you modem is not on the list of
problem modems, further below.

It is entirely possible you will have to modify your
modem cable or replace it; even if you have a modem/comm/
program combination that works, it may not have all the
lines needed to operate a bulletin board. The extra lines
are Carrier Detect (CD) and Data Terminal Ready (DTR). These
two are not normally needed by most communications programs;
they are however needed by Bulletin Boards, since they need
to be able to detect incoming calls and to force them to
disconnect.

        USING A STANDARD CABLE

You can change the RS232 pin that Fido uses to
determine the state of the Carrier Detect signal. This is
done with the /V switch; see the section on Command Line
Switches, and the section on Specific Computers, next. No /V
switch at all is the same as entering:

        16/V   default

Which is NOT what is needed by most machines. Refer
to the Specific Computers section for /V values. Legal
values for /V are: 1, 2, 4, 8, 16, 32, 64, 128.

When Fido ir first run, after initializing itself,
it will say:

- Waiting for a Call or Event

This means that Fido finished initializing the
modem, there is no incoming call, and that it is aiting for
an incoming call. This is the normal standby state of Fido.

When a call comes in, the modem (not Fido) answers
the call. (You can see the modem light the OH, Offhook,
lamp.) When the modem has completed the connection process,
it informs Fido of this with the Carrier Detect (CD) signal
(The modem lights the Carrier Detect (CD) lamp.) The /V
value is a bit mask used to isolate the CD signal from the
others.

When Carrier Detect is true, signaling an incoming
call, Fido displays:

- Incoming call
- Determining baud rate

At this point, Fido is waiting for the caller to hit
CR and/or space bar to determine the baud rate.

When the caller disconnects, the modem drops it's CD
signal, and Fido lowers Data Terminal Ready (DTR)
momentarily, raises it again, and goes back to the original
display:

- Waiting for a call or event

You must get Fido to perform this exact sequence of
events. If it does not, please refer to the Modem Problem
Determination section.

Custom Cabling

The idea here is to connect the cable such that you
can not only communicate to the modem, but also get to the
Carrier Detect and Data Terminal Ready signals to and from
it. If you already have MINITEL, PC-TALK, etc working with
the modem, thats a good start, but not enough. The Carrier
Detect line is not needed by programs such as MINITEL and
PC-TALK; Fido needs it to tell when a caller has made a

connection, or when they have hung up. Just because PC-TALK or MINITEL work with the modem doesnt mean that Fido will.

        The following are the pins that must be wired to an IBM Async Card. You will have to determine how to connect other computers using this as an example.

IBM PC	SMARTMODEM	USE
1	1	Ground,
2	2	Data, modem to PC

3	3	Data, PC to modem,
6	6	Carrier Detect
20	20	Data Terminal Ready

        The other pins are not needed by Fido, though may be needed by other comm. programs, and in any case can be left in.

        Carrier Detect: CD

        CD is generated by the modem, and tells the computer (Fido) that a call came in, and a connection made. Conversly, Fido can tell when a caller hangs up or loses connection, and knows to log that user off.

        Data Terminal Ready: DTR

        Fido uses this to disconnect users that have gone over their limit, and to prevent incoming calls while Fido is performing internal processing that may take a few minutes.

        DTR is also of extreme importance for FidoNet for placing outgoing calls. Fido will not run properly if DTR is not connected and working properly.

        FIDOMDM.BBS is a special, optional, file that Fido
uses to command the modem. If it does not exist, then it is
not used. You need to use this file only if you have special
requirements. Fido will automatically initialize the modem
for you.

        If FIDOMDM.BBS exists, the contents of it are output
line by line to the modem. The contents of the file are not
checked in any way; it is sent blindly.

        After sending one line, Fido will wait up to three
seconds for and responses from the modem; these responses,
if any, are thrown away. The purpose is to avoid sending
commands to fast. If there is no response, Fido waits three
seconds and goes on to the next line.

        FIDOMDM.BBS is processed after Fido initializes the
modem itself; this lets you override or modify the default
settings. For example, if you wanted to leave the
loudspeaker on, and have the modem answer on the second ring
instead of the first, for a Hayes type modem use:

ATS0=2
ATM1

        These are some notes on specific installations. If
your machine combination is not here, you will ahve to play
with the various settings of /V, /J and FIDOMDM.BBS
mentioned earlier.

        Fido has only been tested on the machines listed
inside the front cover; it may or may not work on other
machines.

IBM Jr, PC, XT, AT and ALL CLONES

        Use 128/V for all modem types. A 1 to 1 cable will
work with all modems.

DEC RAINBOW: ALL MODELS
SANYO 555

        USe 16/V for all most types. I have very little
personal experience with these machines.

VICTOR 9000

        Use 8/V.

OTRONA ATTACHE

Use 16/V. You will need to use the special Attache
MODEM cable; the PRINTER cable will not work without
modifications.

        The /J values are listed below. Please check for
notes on certain modem types.

        1/J   Hayes
        2/J   DEC DF03
        3/J   Racal Vadic VA212
        4/J   Novation/ IBM PC Junior
        5/J   US Robotics
        7/J   Prentice POPCOM
        8/J   IBM PC Junior
        9/J   Ventel MD212 with WECO option
        10/J   GDC modem
        11/J   Hayes 2400
        12/J   US Robotics Courier

HINT:
        The following is opinion only; it is not "fact". I
have had the chance to play with many modems while working
with Fido, so I'd thought I'd share some opinions.

This is the only plug you will find anywhere in this manual for specific hardware. The differences in quality between modems is great, and not always apparent to the buyer. "Hayes compatibility" is a misnomer, there are very few that truly are. Even some of Hayes' own products are not "Hayes compatible".

For a 1200/300 baud modems, the Hayes is still by far the best. It is expensive, but you cannot possibly go wrong; it performs flawlessly, there is obviously lots of software for it, and they are plentiful. Some of the clones I have worked with personally, or have suffecient second hand information to rely on.

Hayes clones: Prentice POPCOM, General Data Corp modems are fine modems, though they are "supersets" of the Hayes; you need to set further commands to make them compatible. Once done, they seem to be fine. The US Robotics 212A and Password are fairly good; they have some bugs/side effects problems, but Fido has software to compensate for them. Fairly minor bugs, somewhat of an annoyance for BBS use, but it is very inexpensive.

All Racal Vadic modems seem to be quite good. Only one is compatible with Fido. They tend to have very wordy "menus", which are nice for use with dumb terminals but terrible for use with computer programs.

The DEC DF03 is a terrible modem, mostly because it is a very ancient design. It dials only pulse. It is not a smart modem.

Ventel seems to be a popular modem with the "minicomputer" crowd, but it does not provide a Carrier Detect signal, making it about as useful as a telephone

without a ringer.

For 2400 baud, at this writing there seems to be mainly Hayes, US Robotics, and Multitech. I have not used the Multitech, but it is supposedly a good modem.

I do have both a Hayes and the US Robotics Courier; the USR unit is *far* superior to the Hayes. The Hayes is not "Hayes compatible", the USR is. The USR outsurpasses the Hayes 2400 in all aspects. Except the cabinet is flimsy plastic, oh well. I more or less reviewed the USR in FidoNews (FidoNet Users' Newsletter) and will not repeat it here. No contest, as far as I'm concerned; the supposed handy features of the Hayes 2400 turn out to be useless frills.

Please keep in mind that these are merely my opinions (of which I have more than my share) and not recommendations. If you are dissatisfied it is your own fault.

Hayes SmartModem 1200

Modem Type: 1/J
Switch Settings (External Modem)
        1 Up   Support DTR Lead
        2 Down   Digit Result Codes
        3 Down   Result codes sent
        5 Up   Autoanswer first ring
        6 Up   Supports Carrier Detect lead
        8 Down   Recognize commands
        Others as needed

Anchor Automation Signalman

        Fido will not work with the Anchor Automation, as it does not support the DTR control line. Fido cannot cause a caller to hang up, nor can it disconnect during FidoNet mail transfers. Since FidoNet would be unable to disconnect after placing a call, with a remote possibility of an extremely high phone bill, the decision has been made not to support this modem.

Bizcomp Modem

        This so called "Hayes compatible" is one of the poorest modems ever made; it is barely compatible, and unfortunately, cannot be used for Fido, nor many other comm. type programs.

        It will work OK at 1200 baud, but not at 300. For some unknown reason, the modem transmits all characters at 1200 baud, even when connected at 300. While it sounds like it simplifies things, by not having to switch baud rates, it

also means that you must "throttle" data sent to it; you cannot just send data at 1200 baud to the modem, when it is leaving the modem at only 300 baud.

        This has so many basic flaws that no attempt will be made at making Fido accomodate it.

Qubie 1200 Modem

Another "Hayes compatible" that does not work. This has many serious but subtle problems. The Carrier Detect signal is "faked", and does not work. Other problems were reported. It probably will not work with Fido, unless Basic Time revises the firmware.

US Robotics Modem

Modem Type: 5/J
Switch Settings:
          RNG: Off
          8HI: On
          BAL: Off
          DTR: On

          The 212A and Password have a strange side effect: if a partial command, or just garbage, is sent to the modem, it will not answer the phone. The modem expects a CR after each line, and will sit and wait until it gets one.

          Fido fixes this by issueing commands to clear the modem, hence it is not a problem. Make sure you select 5/J.

Ventel MD212A

          All of the Ventel modems do not normally supply Carrier Detect; this makes them absolutely useless for Bulletin Board use. The MD212A, however, has an optional ROM called "WECO" (dont ask me) that supplies the missing signal. You will need to contact the factory for this option.

          SYMPTOM:

Fido says "Waiting for a call or event" but will not answer the phone.

The modem is not in autoanswer mode. Did you use the right /J setting? Is the modem on? Plugged in?

SYMPTOM:

Fido says "Waiting for a call or event", immediately followed by "Incoming call, determining baud rate" when there is no incoming call.

You have the modem installed incorrectly. The problem is that Fido thinks that there is Carrier Detected when there isnt. There are a number of possibilities.

You may have the /V value wrong. If the one recommended one does not work, try the others. There are eight possible values; 1,2,4,8,16,32,64,128.

You may have switch settings wrong: you must set any switches needed to make the Carrier Detect and Data Terminal Ready signals active. Fido cannot do this.

You may have a cable that does not have all the signals needed; replace the cable.

SYMPTOM:

After a caller logs off, Fido rings the bell continuously, displaying the message "CANNOT DISCONNECT THE MODEM!".

Fido requires that it be able to cause a disconnect by using the DTR line. The Anchor Automation modem does not support DTR, and therefore cannot be used with Fido.

For modems with options switches, it may be merely a switch setting to enable the use of the DTR line.

This can also be caused by a modem cable that does not have all the connections made.

If you see this message, you can type Control-C to abort it. You should also make sure the modem is disconnected until you can repair this situation.