

31.4 Is the net up?

This is a question a lot of your users will start asking you if you are only using an intermittent dial up connection. To answer this question automatically as part of the MOTD generation the program 'IsNetUp' was written (see Section 23.4 [IsNetUp], page 90).

Copyright © 1993 Stephen Vermeulen

This is the fourth edition of the IceBBS manual consistent with version v56 of the software.

Published by Ragnarok Software (Stephen Vermeulen)

45 Butler Cr. N.W.,
Calgary, Alberta,
CANADA, T2L 1K4.

Voice: (403) 282-7990

BBS: (403) 284-2048 and (403) 284-5625
(24 hours, V.32bis and HST)

FidoNet: 1:134/92.0

Internet: svermeulen@ragnarok.mtroyal.ab.ca

WWW: <http://www.cuug.ab.ca:8001/~vermeuls/>

The demonstration version of IceBBS and the AmigaGuide (tm) and PostScript versions of the demonstration version's documentation may be freely distributed. The full version of IceBBS along with the tools for FidoNet and USENET message processing and a printed copy of this documentation are available from the Author for US\$60.00 plus US\$10.00 shipping.

This is the demonstration version.

- A & B** The '&' (and) operator takes the lists of words contained in the two files and produces a new list containing only the words in both files. This is the set intersection operator,
- A | B** The '|' (or) operator takes the lists of words contained in the two files and produces a new list containing all the words in both files. This is the set union operator,
- A - B** The '-' (minus) operator takes the lists of words contained in file A and removes from that list any words which are also contained in file B. This is the set difference operator.

As well parentheses are available to allow you to group several operations:

`A & (B - C | (D & F))`

will evaluate 'D & F' first to produce 'X',

`A & (B - C | X)`

it will then evaluate 'B - C' to produce 'Y'

`A & (Y | X)`

it will then evaluate 'Y | X' to produce 'Z'

`A & Z`

and will finally evaluate 'A & Z'.

If there are no parentheses then evaluation proceeds from left to right as was the case when it evaluated 'B - C | X' above.

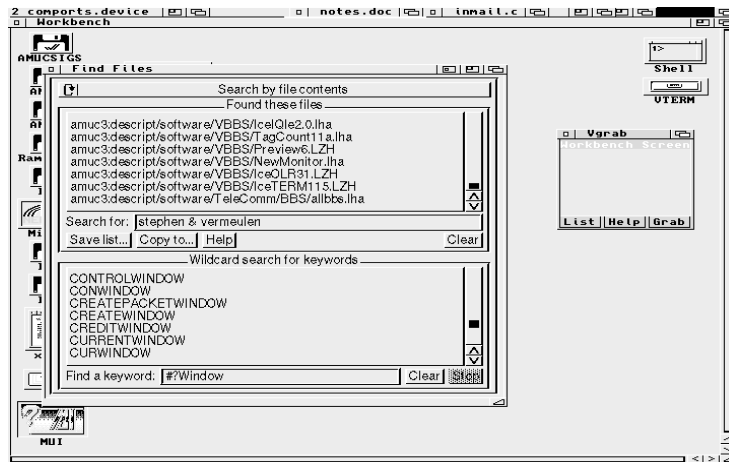
32.2 MUIFind

'MUIFind' is a GUI based file finding utility for use on IceBBS systems and for regular users who have an AMUC CDROM (see Section 44.8.3 [AMUC], page 193) and its cross reference database. 'MUIFind' uses MUI for its user interface, so you will need to get a copy of MUI installed to use it.

32.2.1 Installation

There are three cases for installing this utility:

1. if you run an IceBBS system just run 'MUIFind' and it will read your 's:icebbs.find' file to discover the location of your databases.
2. if you have AMUC CDROM 1 you need to get a copy of the database files for that CD, these are present on AMUC CDROM 2 or they may be downloaded. There will be 5 database files named something like:



Searching by File Contents

A combined search on several keywords can also be done. The search string will be parsed by 'MUIFind' and each word within the string will be searched for in the database(s). This produces a table of files for each word. The finder can apply several operators to combine these tables and hence allow you to search for files which must, may or cannot contain several words. The operators are the same as those described previously in the 'find' command (see Section 32.1 [Find], page 123).

When using the lower area to search for keywords by using AmigaDOS patterns you enter the pattern into the string gadget and hit return. For example the pattern:

```
delay#?
```

will cause it to search for all the keywords that start with *delay*. If you enter:

```
#?delay#?
```

'MUIFind' will look for all the keywords that contain *delay*. You can use the full capabilities of AmigaDOS pattern matching here.

32.2.2.2 Search by file name

When the *Search by file name* mode of 'MUIFind' is selected then there is a single string requester at the bottom of the window. Into this one types an AmigaDOS wild card string and upon hitting return 'MUIFind' will search the databases for any file names that match the pattern. Full AmigaDOS pattern matching is available so that typing:

```
mui#?(1|2|abc)#?
```

will cause a list of all the file whose name starts with 'mui' and follows that by a '1', '2' or 'abc' later in the name. This gets particularly

- The software should support a fully-multiplexed, error-free, communications channel,
- The user should be able to do more than one thing at a time,
- The BBS should not require an accelerated machine to run more than one line,
- The BBS should not eat huge amounts of RAM,
- The BBS should easily support a mixed set of serial ports and modems, as well as providing support for the new fixed baud rate modems (between computer and modem) at the same time as the old style modems,
- The BBS should provide the basics (messages, sysop chat, conference mode, email and a file system),
- The BBS and terminal programs should offer some means of expansion, this is provided by the doors mechanism which allows new clients and servers that behave in the same way as the built in ones to be added,
- The BBS should take advantage of the capabilities of advanced modems (error correction and compression) to off-load work from the Amiga's CPU,
- The BBS should allow messages to be imported from and exported to the major networks (FidoNet and USENET),
- Support CDROMS,
- Support some access to the Internet.

At this point the software meets all of these goals.

2.3 Custom terminal

To realize the multiplexed serial connection (and the BBS multitasking capabilities it provides) it was necessary to develop a special terminal program. This terminal program currently is only available for Amiga computers. As this poses a problem for non-Amiga users an *ASCII interface* to IceBBS has been developed; of course this lacks some functionality and it has not been as extensively tested at this point in time.

The terminal program is also copyrighted software; however, it is freely distributable, so all your users can get a copy of it for free.

baud, 24 hours a day (V32bis and HST) on (403) 284-2048 and (403) 284-5625 and in the evenings (mountain standard time) by voice at (403) 282-7990.

files need to be sorted so that 'MUIFind's keyword searching feature works better. The syntax is just:

```
mysort input output
```

where 'input' is the name of the file to be sorted and 'output' is the name of the file to place the results in. Since 'mysort' does all its sorting in memory you need to have enough RAM for it to load the whole file, plus some overhead to sort in.

32.5 FindStat, trimming the database

This program is used to analyse the word count information produced by 'FindScan' so that you can select certain parameters that prune the data base. There are two reasons you might want to do this. The first is that pruning the database will save you some disk space without imparing its usefulness much. The second is that words which have a high occurance count can cause problems for the users of your system. This is because the system sends the user the full path and file name for each match it finds, so if the user specifies a word that appears in a thousand files he's going to get a lot of output. As it happens these very common words are not very useful for a searching point of view anyway, so it makes sense to delete them from the system (they are a sort of *information noise*).

32.6 FindBuild, building the database

This program is responsible for building the occurance file, which contains the tables that list which files each word in the database index occurs in. This takes very similar command parameters to the 'FindScan' program, in fact the parameters provided should be identical with the exception of the flags.

32.7 The list of databases

Both the user interface to the file finding system through IceTERM and the sysop interface through the 'FIND' command allow for automatic searching of more than one database. The list of databases is stored in the file:

```
S:IceBBS.FIND
```

This file is a white space separated list of the path and root names of all the databases in the system. For example if you have three databases the file might look like:

once, or if they have selected a large editor (such as a commercial word processor⁴) to edit their messages with.

The IceBBS program will also run on a 512K machine, although more memory may be required when your users start multitasking its functions. At a minimum you should budget about 200K for the first line and about 100K per additional line.

⁴ I had one user who used WordPerfect to do this.

the dictionary that the sysop built the database with. With a smaller dictionary (like 10000 words) there will be many more collisions than a larger dictionary like 100000 to 1000000 words). Note that larger dictionaries only cost the sysop file storage space, they do not slow the searching process significantly.

32.8.1 Memory usage

The routines that perform the finding of words in a completed database use quite a small amount of memory, so should not pose a problem. The 'FindScan', 'FindStat' and 'FindBuild' routines all require a single chunk of memory which is 4 times the number of words in the database's dictionary (ie. the same size as the completed '.hsh' file). So for the default 100000 word dictionary you need at least 400K of free memory to run. If you want to produce a million word dictionary you will need at least 4Megs of free RAM, this usually means you must be running on a system with at least 8Megs of RAM installed.

3.2 The Install Script

If you are setting up a single line IceBBS system and you are using a normal 2400 baud modem (ie. one that does not support MNP or higher baud rates) you can install IceBBS by just typing the commands:

```
CD your_directory_for_icebbs
lharc -m x IceBBS54
Install
```

Once the script has finished there are a few things left to do. First you should add a line to your startup sequence file to mount the 'VAUX:' device, for example:

```
mount VAUX:
```

You will usually want to start IceBBS via your startup sequence, so add another line to run IceBBS, you should probably add the 'BBS_C' directory to your command search path by using the AmigaDOS 'path' command. You probably will want to modify the 's:user-startup' file rather than the startup sequence file, as this is the safer and system-preferred way. Note that if the initial shell window will not disappear after you have made your changes you should add some '>NIL:' and '<NIL:' entries to the new commands between the command name and the first parameter that the command takes.

Next you must configure IceBBS for your modems (see Chapter 6 [Register settings], page 25). To do this you may need to make a few edits to the 'S:IceBBS.DIRS' (see Chapter 16 [IceBBS.DIRS file], page 63) and 'S:IceBBS.BINIT' (see Chapter 15 [IceBBS.BINIT file], page 55) files. The 'make_binit'¹ command will make an 'S:IceBBS.BINIT' file for you taking into account the root directory you specified, it will also make an 'S:IceBBS.DIRS' file which lets the users see the contents of the directory²: '<dir>BBSFiles'. The script file will also set up three SIG areas for the SIGs tool.

3.3 Configuring Your Modem

If you have chosen to just use the default installation, the next thing to do is to configure your modem. The basic commands that must be sent to the modem are:

```
AT&D3      ;hang up using DTR
AT&C1      ;pass carrier detect
ATV1      ;command results as text strings
ATX4      ;report most modem events
```

¹ This command is not yet available, sorry

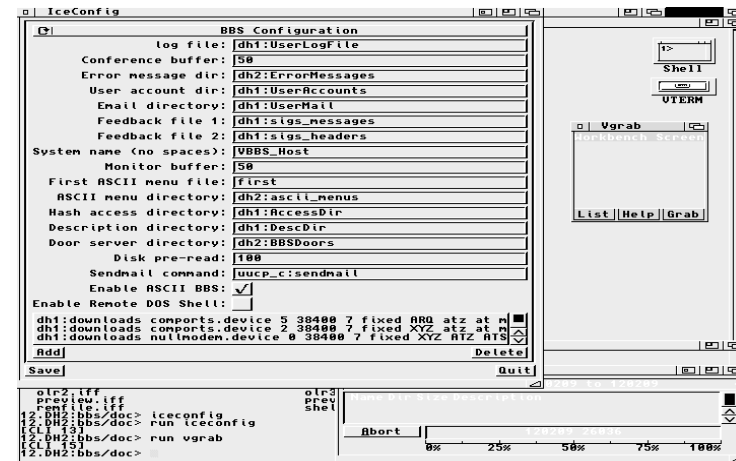
² The default 'VBBS.DIRS' file will make several directories accessible

```

#define FH_CRCVALID (1 << 14) /* the CRC is valid, ie.
    the file uploaded OK, Pre-89 versions of TERM
    and anonymous uploads do not set this bit
    */
#define FH_LOCN_MASK 0x03
    /* these bits are used to specify where the file is
    */
#define FH_INPLACE 0
    /* a regular IceBBS file, the file follows
    the long description
    */
#define FH_INDIRECT 1
    /* indirect access, the data following the
    long description is really the full path and
    name of the AmigaDOS file to be sent to the
    user. This is useful for CDRoms
    */
#define FH_SOMETIMES 2
    /* like FH_INDIRECT except the if the media is
    "offline" (maybe on tape or something) so
    the file request should be logged to
    a file for future processing (not implemented
    yet)
    */
#define FH_OFFLINE 3
    /* the file is not in online storage, but the
    "file" area contains the requesting info
    that is to be written to the request file.
    For example this might include FTPing or
    FREQuing instructions.
    */

USHORT CRC; /* 16 bit CRC of the file */
ULONG Downloads; /* the number of times
    downloaded */
ULONG DescLength; /* the length of the long
    description */
UBYTE User[64]; /* name of the user who
    uploaded it */
UBYTE ShortDesc[64]; /* the short description */
UBYTE LongDesc[2]; /* the long description...
    variable sized */
};

```



The IceConfig Tool

3.6 The S:IceBBS.HAIL File

This file (which is required to be in your 'S:' directory) is used by IceBBS to send an ASCII text welcoming banner to all who successfully connect to the system. This welcoming text may be as long as you wish. It is sent before the IceBBS protocol is started, so that non-Amiga users can see they have connected to a very special kind of BBS they cannot make full use of, and to give them instructions on how to start the ASCII side of the BBS.

3.7 The S:IceBBS.BINIT File

This file (which is required to be in your 'S:' directory) is used by IceBBS to configure its serial lines and to specify the various special directories and files that IceBBS requires. This is a line oriented file, so each line in it has a fixed purpose. There can be a variable number of serial port configuration lines, and these appear at the end of the file to allow for this. For more information see Chapter 15 [IceBBS.BINIT file], page 55.

3.8 The S:IceBBS.DIRS File

This file (which is required to be in your 'S:' directory) is used by IceBBS to determine which directories you are going to allow users to download files from. You may list any number of directories here, by providing complete paths to each directory and separating the directory names by "white space" (ie. spaces, tabs or new lines). Because of this

AmigaDOS file and a second file containing a long description and combine them into a single IceBBS format file. This program takes a number of parameters.

33.3.2 BBSSplit

To allow the sysop to extract long descriptions and regular AmigaDOS files from IceBBS format files the 'BBSSplit' utility is provided. This utility will take an existing IceBBS file and extract from it one or both of the AmigaDOS and long description files.

33.3.3 UJoin

'UJoin' was written to automatically add descriptions to files. A more recent attempt at the same thing is 'AmiJoin' (see Section 33.3.4 [AmiJoin], page 134). 'UJoin' will scan the contents of its input directory looking for files that end in '.desc'. If it finds one of these it looks for the matching file without the '.desc' and joins the two together. The resulting combined file is copied into the output directory and the two input files are deleted. Credit for the upload can be assigned with the '-c' switch.

33.3.4 AmiJoin

The 'AmiJoin' command was made to join '.readme' files from Aminet with their corresponding AmigaDOS files to make VBBS format files. Like the 'UJoin' command it processes an entire input directory, looking for pairs of files with the same base name but where one file ends with '.readme'. When it finds a pair it extracts what information it can from the '.readme' and places that information into the VBBS header of the file in the output directory, it then deletes the input files. The '-c' flag can be used to set the uploader credit line. The command's syntax is:

```
amijoin -iINPUTDIR -oOUTPUTDIR [-cCredit_Where_Due]
```

The format of the '.readme' files is quite a bit more flexible than in the case of 'ujoin'. Aminet uses a header that's rather line the RFC-822 specification for email on the Internet. The 'amijoin' command looks for a line starting with 'short:' and uses it for the short file description. It then places a copy of the full '.readme' file into the VBBS long description.

messages it finds into RAM as soon as the user starts reading⁵, and unless you are using 9600 baud (or faster) modems the modem will not be able to keep up with the disk. Of course if you are using high speed modems, why do you not have a hard drive?

The feedback system has a number of major limitations, these include:

- The users can only backup by using the ASCII terminal mode,
- Threaded reads are not implemented,
- Old messages cannot be deleted from the message base (apart from deleting the entire message base).

3.11 Enabling the Remote Shell

In order for your remote sysops to use the remote AmigaDOS shell (see Chapter 40 [DOS shell], page 177) you will need to do a few additional things. First copy the file 'VAUX-Handler' into your 'L:' directory. Second, add the small 'mountlist' file that is found in the 'DEVS' directory of the distribution archive to your 'DEVS:mountlist' file. This will put the proper mounting information for the handler into your system. Next add to your startup sequence script file a command like:

```
mount VAUX:
```

to bring the handler online sometime before the command to run IceBBS. And last, create a file in 'S:' called 'IceBBS.SHELLACCESS'⁶. See the chapter on the remote DOS shell for more information.

3.12 Running the BBS

Once the previous configuration steps have been finished you run IceBBS just like any other CLI only program, type: 'RUN IceBBS'. Depending on what other software you intend to run at the same time as IceBBS you may need to either run IceBBS at a higher priority or run the other software at a lower priority. If you are often running compute bound programs like a raytracer or an archiver like 'Lharc', or you make heavy use of the WorkBench, the easiest thing to do is to run IceBBS at a priority of 4, using the AmigaDOS command 'ChangeTaskPri 4' in the CLI or Shell window before starting IceBBS. It is recommended that you set the CLI stack to about 10K before starting IceBBS.

⁵ This only applies to the current version of the IceTERM reader, the ASCII reader does not work this way and a future version of the IceTERM program will not either.

⁶ It can contain anything.

Create icons too

when 'fmt' is moving or copying it will create icons to go with the files it creates,

Output to dest

the files that are created will only be written to one destination directory,

Output to dest + NEW:

the files that are created will be written to both a destination directory and a new files directory. The new files directory is specified in the configuration editor. This makes placing a copy of the files into both the new files directory (which can be scanned by the 'newsince' command, see Section 23.3 [NewSince], page 90) and the proper permanent directory in the file system very easy.

One destination

if all the files in the to be processed list are to be placed in one directory then this setting will only ask you for the directory once,

Many destinations

otherwise, select this setting and you will be prompted to select a destination directory for each file in the list.

The various processing methods are described below:

- CRC Test** test the files against the internal CRC value to see if they suffered damage during the upload. Note that it is possible for someone either with an old version of IceTERM, by uploading with out description or by uploading from the ASCII side of the BBS to send a file without the internal CRC value,
- Browse** view what information is stored about each file, including its type and any short or long descriptions,
- Rename** rename the files in the list,
- Move** move the files in the list, that is copy them to a destination and then delete the originals,
- Copy** copy the files in the list,
- ARC Test** test the files, by using the appropriate archiver tool's test mode, to see if any are corrupt,
- Edit** edit the long or short descriptions of the files in the list,
- Delete** delete all the files in the list,

access

The ‘access’ command will then prompt you to enter the appropriate information.

‘Access’ needs to read the ‘S:IceBBS.MAXACCESS’ file (see Chapter 10 [IceBBS.MAXACCESS file], page 43) to determine the maximum number of access levels to the BBS, and if this file is not present it will default to 16 levels.

33.3.8 HashAccess

This is an alternative command to the ‘access’ command. It is a new addition intended to be used with CDROM (or other read only) devices to allow access control to be added. Since it is new, it has not been extensively tested yet so be careful.

33.3.9 FileList

The ‘filelist’ command is used to make listings of the files that are available for download. This command has a number of command line parameters to control how the listings are built, and it will only scan the directories to which you let it have access. Type ‘filelist’ by itself to get the help screen.

33.3.10 Store

The ‘store’ command is useful for debugging script files that produce temporary files that need to be saved for examination. By using a combination of ‘store’ and ‘delold’ one can create a rolling log of the last few days of activity, which might be useful when tracking down errors in FidoNet or USENET export trips.

33.4 Listing New Uploads

One result of IceBBS using the AmigaDOS directory system for its file storage is that it becomes difficult to provide a list of the recently uploaded files to users. The work around I have adopted for the moment is to set up an assign to a directory called ‘NewFiles:’. Add this assigned name to your ‘S:IceBBS.DIRS’ file so that it is visible to users as soon as they click on the Get Files tool in IceTERM.

Then, whenever you validate a newly uploaded file, place that file in it’s proper place in your directory trees and at the same time place a copy of the file into the ‘NewFiles:’ directory. Now a user can easily find

parts there is room here for an ambitious hacker to write a higher performance version³ of this driver software.

The A2232 card performs quite well, its only weakness is the lack of official baud rates higher than 19.2K baud (there is an undocumented rate of 115K baud, but that does not work with handshaking and with a direct wired IceBBS connection it seems to result in about 25K baud transfers). The driver software also seems to have problems when 6 ports are being run at 19.2K baud with RTS/CTS handshaking⁴. It handles 6 ports at 9600 baud fine, but the higher baud rate seems to cause some of the lines to go to sleep⁵ for extended periods of time during transfers.

The A2232 card’s driver software also has another bug, it does not drop the DTR line properly⁶ when the serial device is closed. Since this is the hardware method of resetting a modem (by using the ‘AT&D3’ setting) any system using an A2232 may have to use the software method of ‘++ATZ’ instead. The easy way to test your serial port for this problems is to fire up a terminal program, watch that the modem connected to a A2232 serial port shows its “TR” light go to on and then exit the terminal program, the “TR” light should now extinguish. If it stays on then the serial driver software has the bug. A more recent version of the driver exists that has fixed this problem, but it may not be shipped to end users for a while. There is another bug with the serial driver that can cause lines to stop responding. There is a software patch for this which has been added to IceBBS.

From browsing the A2232 manual’s schematic diagrams of the A2232 card it appears that the 6551 chips are being driven by a 1.79MHz clock (actually one quarter of the Amiga’s 7.1MHz clock). Now according to the only information⁷ I have on the 6551 chip the clock speed required for this chip should be 1.8432MHz. So it looks like the actual maximum baud rate might be 18646 baud⁸. Also it might be possible⁹ to modify

³ Please someone, do this!

⁴ As soon as you try running 3 lines at 19.2K baud the card starts to loose characters, this becomes worse as more lines are run.

⁵ Of course this could be due to a bug in IceBBS or IceTERM, but they work fine with ASDG boards at higher baud rates even at the same time that the A2232 has stalled.

⁶ At the time of writing of this manual this bug has been fixed in beta versions of the new driver and this driver has seen wide circulation.

⁷ *6502 Assembly Language Programming* by Lance A. Leventhal, Osborne and McGraw-Hill Inc. 1979

⁸ There is a 1.8432MHz crystal installed on the board so the schematics might be completely misleading.

⁹ This is a suggestion that may or may not work, I have not tried it with my board and I have no way of knowing if there will be some

33.5.2 BulkJoin

Since it would be a *lot* of work to reverse the effects of a `'bulksplit'` command the `'bulkjoin'` program was created. This command combines the separate description and file directory trees into one IceBBS file type tree.

Note that if you have any files in the source files tree³ which do not have descriptions corresponding to them in the desc tree these files will not be found and hence will be left behind.

If you have any descriptions for which the files cannot be located then these descriptions will be left behind and nothing will get written to the combined files destination.

If you have put a IceBBS combined file into the description tree it will be detected and just copied across to the combined file tree.

Finally, if you have put regular AmigaDOS files into the description tree, these will be detected and they will be written to the combined file tree with a default IceBBS header attached.

These last two allow you to take a description tree for a CDROM (or perhaps several!) and mix into it regular IceBBS and AmigaDOS files as you want, then to make a single combined image.

33.5.3 BulkTest

The `'bulktest'` program can be used to check files of the `'DMS'`, `'ZOM'`, `'LHA'`, `'LZH'`, `'ARC'`, `'ZIP'`, and `'ZOO'` types en-masse and to print a report of any suspect files it finds.

33.5.4 Example

Ok, so lets follow the process through step by step. Say you have a set of IceBBS type files (ie. the combined description and file in one) in a directory tree whose root is:

```
DH2:bbs/files/
```

and you want to process these so that you get the separate description files and AmigaDOS files. These will be placed in separate directory trees, we'll place the descriptions in:

```
DH2:bbs/desc/
```

and the actual files in:

```
DH3:rawfiles/
```

³ A file tree probably made by a `'bulksplit'` command.

4.3 ASDG - Dual Serial Board

The ASDG dual serial card is a nice piece of hardware but the software is in need of some bug fixes. This beast is also a Zorro II card and it has two 9 pin (joystick type) connectors on the back panel. You can use standard AT serial cables with this card. The ASDG driver software is up to version 1.6, I advise you to send in your registration and get the most recent copy from ASDG since they have fixed some significant bugs in the earlier versions which were related to use with accelerator cards.

This card, on an accelerated Amiga, is capable of running two lines with IceBBS at baud rates of 153.6K baud, with an unaccelerated Amiga two lines at 38.4K baud is achievable. This card has four problems which I know of:

- It does not coexist with TrapDoor 1.80 well. This is because of a bug in the way the card does either a `'CMD_CLEAR'` or a `'CMD_FLUSH'`,
- It drops characters from time to time. This only seems to happen when there is handshaking involved as is the case when talking to high speed modems,
- It suffers from some sort of `'mungwall'` incompatibility when it is first loaded from disk, if the serial port is opened before `'mungwall'` is started all seems to be fine.
- It has a bug some where that will cause your system to crash after a period of time, the time this takes depends on the activity level of your system and the number of lines you have.

This card is not currently being manufactured by ASDG any more, so you will probably only find it on the used equipment market.

Summary: for lightly loaded one and two line systems this card will be fine; however, if you're trying to run a large system or stability is a concern then this card probably will not be satisfactory.

4.4 Multiface II and III dual serial cards

These are Zorro II cards that contain 2 serial ports and either one or two parallel ports. The cards work well in all Amigas (2000, 3000 and 4000). They run flawlessly with both IceBBS and IceTERM, even in heavily loaded environments. The only drawback with these cards is that they limit the total number of serial ports in a single machine to 10 (assuming you don't need a slot for a hard drive controller).

There may be a 4 port version of these cards but I have yet to see it.

4.8 Direct Line Hookup

It is also possible to run a direct serial connection between IceBBS and IceTERM. All you need is to make up a standard null-modem cable and connect it between the two serial ports. Next setup the 'S:IceBBS.BINIT' file to specify that the BBS serial port is a fixed rate line. Now to start the IceBBS protocol over the direct wire line, just type 'CONNECT' and hit return in the terminal program's ASCII mode window, in a few seconds the BBS will send the hailing message and start the protocol. If you want a direct wired connection to run with the MNP/V42 protocol you should type a string that contains the MNP or V42 connect string, for example: 'CONNECT ARQ' or 'CONNECT REL'.

4.9 Simulating a null modem in software

Rather than using two physical serial ports connected by a null modem cable it is possible to simulate the connection by software called the null modem device. This is a device that is placed in the 'DEVS:' directory. It was written by Ian Hibbert who can be reached at:

UUCP: plunky@closet.wizdom.royle.org
 FidoNet: 2:255/171.33
 AmigaNet: 39:136/1.33

A typical configuration line for the 'S:IceBBS.BINIT' file would be:

```
dh1:downloads nullmodem.device 0 38400 7 fixed XYZ +
      ATZ ATSO=1 mail:line2 -1 -1 -1
```

Note the two strings between the 'XYZ' and 'mail:' fields. The first string is the command to send to reset the modem after the BBS hangs up the line and sends an '+++' sequence. Normally one puts an 'ATZ' or an 'ATH' command here. The second command is the string to send the modem to re-initialize it. In the case of the 'nullmodem.device' one specifies an 'ATSO=1' which causes the device to auto-answer.

This makes it very easy to use nullmodem.device with IceTERM and IceBBS. Just configure the BBS line as shown above, and then use a dialing script like the following:

```
send ATD
cr
exit
```

with your IceTERM. If the BBS is using nullmodem.device unit number 0 as shown in the example, then IceTERM needs to be configured with the following tooltypes:

```
DEVICE=nullmodem.device
UNIT=1
```

```
};
```

The `'messagenumber'` is the reference number for a particular message. These numbers must be sequential (there cannot be gaps or duplicate message numbers), although the first message in the file can start with any number from zero on up. The message server software looks at the first and last record in the header file to determine where in the file is the first new message that should be sent to the user. The `'messagelen'` field is the length of the text part of the message in bytes. The `'fileposn'` is the location in the data file at which the text of the message resides. The message server loads each message to be sent to a user by seeking to the `'fileposn'` and then reading `'messagelen'` bytes from the file. A crude and simple approach.

The `'time'` field is the time that the message was posted by the user, this is stored as the number of seconds since the start of the Amiga's clock (which is January 1, 1978). This is the same integer that you would obtain from the C linker-library `'ctime()'` function.

The `'replyto'` field contains the number of a message that this message was a reply to. If this message is a new posting then this number is `'-1'` which should be considered to be an illegal message number.

The `'username'` and `'topic'` fields are null character terminated ASCII strings. The `'username'` is the name of the user who posted the message, and the `'topic'` field is the subject heading for the message. On new postings the system obtains the subject heading by extracting the first line of the text file on the user's machine. On replies the `'topic'` is just a copy of the original subject line.

To set up an old style message system all that needs to be done is to specify the names for the index and message files in the `'S:IceBBS.BINIT'` file (see Section 15.6 [Line 6 and 7], page 57). As soon as the second message is posted (either from a user's terminal session or via the sysop's `'mail'` utility) the two files will be created. The complete set of steps that must be fulfilled are:

- edit the sixth and seventh lines of `'S:IceBBS.BINIT'` file to have the correct file names for the index (header) file and the data (text) file for your message base and restart IceBBS,
- copy a `'TERM.info'` file which has your user name and password in its tooltypes array and name it `'MAIL.info'`, place a copy of this either in the directory you start the `'mail'` command from or in the `'S:'` or `'C:'` directories if you prefer to be able to run `'mail'` from anywhere,
- run the `'mail'` command and post one (1) new message. Run the mail command a second time and post one more message.
- check the user accounts directory it should now contain a user ac-

The linkages between messages in the SIG are stored in a file named: 'signame.lnk' where 'signame' is the name of the SIG. The linkage file contains:

```
a single struct FirstLinkItem
a LinkItem
a LinkItem
a LinkItem
and so on...
```

34.2.2 The FirstLinkItem structure

The 'FirstLinkItem' structure is a special case, it will always contain some padding to make it the same size as the 'LinkItem' structure. There is only one 'FirstLinkItem' structure, and it is always located at the beginning of the SIG's data file. This structure serves as the root node of the message tree for a particular SIG.

```
struct FirstLinkItem
```

```

{
  ULONG MsgNo; /* the number of this message
               (actually a date of posting) */
  ULONG Parent; /* number of this message's
                parent */
  ULONG Child; /* number of the first reply to
                this message */
  ULONG Next; /* next message at this level in
                the tree */
  ULONG Prev; /* previous message at this level
                in the tree */
  ULONG Offset; /* location of the message in the
                 data file, this is the offset
                 from the beginning of the file
                 to the DataItem structure for
                 this message, the data item will
                 be read to determine the length
                 of the message in the file, and
                 its type (compressed or
                 uncompressed)
                */
  ULONG Subject; /* crc of the subject string */
  ULONG Author; /* crc of the author string, except
                 for the EMAIL SIG where this is
                 the CRC of the user the message
                 is addressed to, if they are not
                 on this BBS then it is set to
                 zero.
                */
  ULONG Flags; /* special flags pertaining to this
                message, these are shared by the
                BBS and the TERM systems, so if
                you change one of these flags be
                certain to preserve the other 31
                bits. Note these are the bit
                values.
                */
#define LI_READ (1L << 0L) /* off line reader
                           sets this when the user
                           has read a message */
#define LI_DELETED (1L << 1L) /* message is
                               marked as deleted */

```

than polling the Ring Indicator pin like some BBS packages do because polling is a bad thing. Note that I have set my system to answer on the second ring because if you answer on the first ring the phone may not have rung on the caller's end and his modem may not respond correctly (the only modem I have seen do this is an old Packard Bell 2400 baud modem),

'AT&W' This saves these configuration settings into the modem's memory.

The modem is now be set up for use with IceBBS.

6.2 Advanced modems

These are the modems which you specify with a 'fixed' baud setting in the 'S:IceBBS.BINIT' file. This type of modem usually supports MNP or V.42 and it supports a fixed baud rate link between the computer and the modem. The general approach to configuring an advanced modem for use with IceBBS is to lock the baud rate of the linkage between IceBBS and the modem, enable CTS/RTS³ flow control on that linkage and tell the modem to negotiate with the calling modem to establish the best protocol between them. IceBBS does not need to know or even care⁴ that the link is an MNP link at 300 baud or an HST line at 14400 baud the protocol will still work fine. The following are the common settings that should be set for use with IceBBS:

ATL0	This selects the lowest volume for the modem's speaker,
ATM0	This turns the modems speaker off at all times,
ATQ0	This turns on result codes, IceBBS needs these so it can see when someone has connected to the modem, the baud rate of the connection, and the error correction used in the connection,
ATV1	This makes the result codes be the English text strings,
ATX4	This makes the modem report all ⁵ events to us, the ones that IceBBS looks for at are the 'CONNECT', the baud rate string and the error correction protocol name,

³ The importance of flow control cannot be over emphasised.

⁴ IceBBS can make good use of this information though, it will allow a faster version of the protocol to be used which will result in better throughput and less CPU loading on both ends of the connection.

⁵ Users of US Robotic's Dual Standard and HST modems can use 'X6' here.

```

struct MessageHeader
{
    ULONG Type; /* a 4 character identifier for
                the file type == VMSG */
#define VMSG (((ULONG) 'V' << 24L) \
             | ((ULONG) 'M' << 16L) \
             | ((ULONG) 'S' << 8L) \
             | ((ULONG) 'G'))

    ULONG BBSDate;
    ULONG CreationDate;
    UBYTE Author[64];
    ULONG DelBy;
    ULONG ReadCount;
    ULONG Archive;
    ULONG Expansion[5];

    /* the following six length fields are used
       to indicate the number of bytes of data in
       each field.

       Any or all of these can be zero bytes long.

       The actual data appears in the file in the order:

           Subject, Address, Destination,
           OldSubject, Keywords, Network.

       */
    ULONG Keywords;
    ULONG Network;
    ULONG Subject;
    ULONG Address;
    ULONG ReplyTo; /* this used to be: Destination */
    ULONG OldSubject;
};

```

where:

VMSG a 4 character identifier for the file type, if the SIG server encounters non-IceBBS files in its wanderings it will treat them as a straight ASCII message,

BBSDate the date at which this message was posted to the system. The date is in conventional C-library (ctime) format,

AT%C1 Turn on compression (MNP level 5),

AT\J0 Turn off the automatic baud rate adjustment. This keeps the modem to computer connection at a fixed baud rate regardless of the actual modem to modem link speed. The GVC manual documents this incorrectly in one place and correctly in another so who knows what it is really supposed to be,

AT\N3 This enables the modem to negotiate for an auto reliable link, it first tries to get an MNP class connection and if that fails, it falls back to the old style connection,

AT\V1 Tells the modem to send extra result strings to describe the connection it got. This is not really necessary for IceBBS but it can be useful as a diagnostic,

AT\Q3 Tells the modem to use full CTS/RTS flow control between it and the computer, essential if you are wanting to go faster than 2400 baud,

AT\G0 This is a mysterious command, it seems that the 'G0' setting is required at least at the caller's end (when IceBBS is being run on old-style modems and the caller has an MNP modem), for this reason I recommend that MNP modems on the IceBBS end also be set to 'G0' (this is usually the default factory setting).

With these settings stored into the appropriate register set you should be ready to run IceBBS with an MNP type modem.

6.4 USR HST Modem

IceBBS has now been tested with a pair of USR HST Modems, performance was good on unidirectional transfers (1630 CPS) but became poor when bidirectional transfers took place (about 350 CPS). The performance drop from the expected 800 CPS bidirectional can probably be attributed to the HST modem's asymmetric nature, it appears that the HSTs do a line retrain everytime that the high speed direction changes. It looks like the retrain process takes a significant amount of time to complete and hence we see a large drop in performance.

Note that the above figures are obtained when IceBBS and IceTERM are running in the *error-corrected modem mode*, if this protocol is disabled then some additional performance will be lost since IceBBS and IceTERM use a smaller packet size than the HST modem likes best.

For a BBS system I would strongly recommend not buying an HST modem, but rather getting a USR Dual Standard modem, as these now

These fields are each the indicated number of bytes long, there are no null termination bytes, but there could be. Any or all of them could be zero.

Subject The subject line that the author has provided – in original postings this is the subject, but in replies this becomes a secondary line. This is filled in by the NewsIn and Import tools when USENET and FidoNet messages are received.

Address this is the name of the user the message is addressed to. The FidoNet utilities will copy this into the ‘To:’ field of a FidoNet message header, if it is not supplied then the default of ‘A11’ will be used. On USENET messages this field is ignored if present,

ReplyTo this is the full Internet RFC-822 ‘Reply-To:’ line, if it is present this will be used as the address to which replies are sent rather than the ‘author’,

OldSubject
the length of the previous subject in bytes.

Keywords the size of the user defined keyword space, some reader programs allow the user to enter a number of keyword topics, these are stored in the message header for future searching,

Network the size of a network information block. This is primarily intended for FidoNet support, the import program will strip the FidoNet control lines from the message and place it in this data area, then on export it will move it back into the body of the message so that it is Fido compatible. Note that the reader should not copy the contents of this block into replies, instead it should create a zero length field for any fresh postings.

The complete USENET header is also removed from the body of the message and placed here by ‘newsin’, in this way we can be assured that the user does not tamper with it (unless he writes a special tool).

Note that several useful features are provided in the header. The first is that the author also has an email address, this is to allow for networking. The ‘DelBy’ field allows for automatic message purging on a message by message basis. Several possible uses for this exist, a moderator might read a bunch of garbage messages (or superfluous responses) and this allows him to immediately (or at any later date) have these messages deleted so that the quality of the discussion is maintained. Note that setting the ‘DelBy’ field causes a *soft delete* of the message, the SIG server just no longer sends the message to unprivileged users,

AT&F2 Select factor default settings for hardware handshaking,

ATW2 Should get the modem to report the connection speed rather than the speed of the serial port link (which is going to stay fixed),

AT&C1 Returns the current state of the carrier so the BBS can tell when the connection has been broken,

AT&D3 Get the modem to do a full reset when the BBS drops the DTR serial port line,

AT&K3 Enable the hardware handshaking,

AT\A3 Allows use of full MNP block sizing (this may not be too important),

AT\N3 Negotiate for the best connection, starting with V42, dropping back to MNP and then falling back to a regular non-corrected connection,

ATS0=2 Answer the phone on the second ring (so that all callers hear at least one ring),

ATS95=3 This convinces the modem to send the connection string as only one line (like other modems do),

AT&Y0 Make register set number 0 the default one,

AT&W0 Save these settings into register set 0, so that when the modem is reset or powered up again it remembers these settings.

These modems work pretty well now that Supra has upgraded the ROMs several times. There is a command to print out the ROM version, you might want to do this to check that you have the latest ROM version, and to upgrade if you have an older ROM. Their only failings are that older V32 modems have trouble connecting to them and the long distance callers do not always connect at the highest rate.

6.7 V.32 and V.32bis Modems

V32 is the international standard for 4800, 7200 and 9600 baud communications. V32 modems are available from a number of manufacturers, most V32 modems will also feature MNP and V42 error correction protocols. Note that V32 modems will only connect at 2400 baud when an HST modem is calling them. V32bis is the new international standard for communications at 12000 and 14400 baud, it will also fall back to the V32 speeds if necessary. The advantage a V32 or V32bis modem has over an HST modem is that it allows data to be sent in both directions


```

if (MsgNo == 1L)
{
    Seek(file, 0L, OFFSET_BEGINING);
    Read(file, link, SIZE(LinkItem));
    return(TRUE);
}

if (!fli)
{
    Seek(file, 0L, OFFSET_BEGINING);
    Read(file, &localfli, SIZE(LinkItem));
    fli = &localfli;
}

lower = 1L; /* lower limit of the search */
upper = fli->Number;

/* before searching we should examine
   the lower and upper bounds to see if
   we are sitting right there...
*/

Seek(file, lower
      * SIZE(LinkItem), OFFSET_BEGINING);
if (SIZE(LinkItem)
    != Read(file, link, SIZE(LinkItem)))
    return(FALSE);

if (link->MsgNo == MsgNo)
    return(TRUE); /* found it */
if (link->MsgNo > MsgNo)
{
    /* seek to the right spot... */

    Seek(file, lower
          * SIZE(LinkItem), OFFSET_BEGINING);
    /* requested message too old */
    return(FALSE);
}

```

6.10 High Speed Protocol

IceBBS makes special provisions for V.42, MNP and HST connections. If such a connection is detected when the user logs in, IceBBS and IceTERM will start up a different version of the protocol that does not do the error correction (since the modems are doing this). When this is done the overhead on file transfers is reduced to about 2% and less CPU at both ends is required.

The error correction is almost entirely handled by the modems, but not quite. It appears that not all serial ports run without loosing bytes occasionally. Also, when a phone connection is broken some MNP modems return a brief burst of garbage to the computer, instead of falling silent. Both of these cases can be disastrous to a protocol like that used by IceBBS which requires that the data gets through intact.

As a result of this the high speed protocol still has to check for errors; however, at this point in time it does not have the ability to recover from errors. Therefore when an error is detected it has to break the connection. In the future a retry capability will be added to get around this limitation.

If you find that users complain a lot about IceTERM reporting an **MNP Error** and then hanging up the connection, you will have to turn off the high speed connection. This can be done by editing the string that the BBS looks for in the modem connection strings to decide if there is an error free link in place between the modems. This string is set in the 'S:IceBBS.BINIT' file, all that needs to be done is to change this from its normal value (which might be 'ARQ' or 'REL') to something that the modem does not return (like 'XYZ').

```

while (lower < upper)
{
    target = (upper + lower) / 2;
    Seek(file, target
        * SIZE(LinkItem), OFFSET_BEGINING);
    if (SIZE(LinkItem)
        != Read(file, link, SIZE(LinkItem)))
    {
        return(FALSE);
    }
    if (link->MsgNo == MsgNo)
    {
        /* got it! next read will read
           the following message
           */
        return(TRUE);
    }
    else if (link->MsgNo < MsgNo)
    {
        /* because of integer math we can end up
           with the target equal to the lower bound,
           this is the time to stop since further
           iterations will produce the same result
           (as lower and upper are adjacent and we
           have already inspected both the seek
           position is correct too)
           */
        if (lower == target) return(FALSE);
    }
}

```

7.2 Mixing access levels

It is also possible to mix and match the various levels so that a code of '1 0 0 1' would give a user access to the unique sysop functions but not to the areas that members and executive only have access to.

The access control system gets more versatile (and complicated) when you remember that each of the groups (there are four groups in the above examples) can each take on 256 different levels. The rule to remember is that if a user's account has not been set to a level equal to or greater than the level the sysop has set for the function² the user is trying to access then he cannot use that function.

For example if we add a fifth group (group number 4) to the previous example, this being used to control access to the doors. And if there are four doors on the BBS we might assign the following levels to each door:

'0'	for the IFF picture preview door,
'1'	for the voting door,
'2'	for the GPChess door,
'10'	for the remote sysop file maintenance door.

As everyone must have at least a level zero access in all the security groups this makes the preview door automatically available to everyone. Next consider a member who is given the access levels: '1 1 0 0 1'. He will have access to both the voting door and the IFF picture previewer. If his level in group 4 is raised to 2 ('1 1 0 0 2') then he will also get access to the GPChess door. However, he will not get access to the remote sysop file maintenance door ('RemFiles') until his access level in group 4 is raised to at least 10.

7.3 Access levels in different groups

It is also possible to give a user access to particular functions (or file areas or SIGs) in several ways. If, for example, we continue the previous example on the doors system. It is possible to tell a particular door that access to it can be granted based on more than one security group. For example we could automatically give every sysop access to the 'RemFiles' door by setting an alternate access level of one in group 3 in addition to the above level 10 in group 4. This way, whenever someone is given *sysop status* by setting their group 3 level to 1 they automatically get access to the remote sysop door. Consider the following accounts:

² This includes: doors, individual SIGs and directories within the file system.

then to find each subsequent message after that (although its not really necessary since they are in fact sequentially stored).

To find the first real message in the file, just ask to find message number 1. This works because message number 1 is permanently reserved by the 'FirstLinkItem'; and hence, the message that follows number 1 will be the first real message in the file.

```
int FindMessageAfter(BPTR file,
                    struct LinkItem *link,
                    ULONG MsgNo,
                    struct FirstLinkItem *fli)
{
    FindMessageNo(file, link, MsgNo, fli);
    if (SIZE(LinkItem)
        != Read(file, link, SIZE(LinkItem)))
        return(FALSE);

    /* check to see if the message that was
       returned is number 0, if it is we have
       reached the logical end of file, so
       again return false
    */
    if (link->MsgNo == 0) return(FALSE);
    return(TRUE);
}
```

34.2.6.3 Finding the previous message

This function is used to find the message that occurs before a particular message. This would be used in a sequential reverse read.

```
int FindMessageBefore(BPTR file,
                     struct LinkItem *link,
                     ULONG MsgNo,
                     struct FirstLinkItem *fli)
{
    FindMessageNo(file, link, MsgNo, fli);
    Seek(file, -(2*SIZE(LinkItem)), OFFSET_CURRENT);
    if (SIZE(LinkItem)
        != Read(file, link, SIZE(LinkItem)))
        return(FALSE);
    return(TRUE);
}
```

could just give themselves this level anyways so this is what the shell server looks for. So on AMUC's system a full sysop would have '255 255 255 255 255'. Note that if you have no 'S:IceBBS.MAXACCESS' file set you will have to specify 255 as the access level for all 16 groups before a user can get access to the DOS shell.

7.6 Setting access levels

To set the access levels for a particular user the sysop uses the 'edituser' command or the 'eu' program.

See Section 33.3.7 [Access], page 137 and Section 25.1 [Edituser], page 95 for more information on tools to control user access.

7.7 Privileges

Starting with version 47, IceBBS includes several forms of system privileges. The first are the controls to enable or disable the various built in clients and servers in the user's IceTERM program. Each of the buttons of the main control pannel of IceTERM can be selectively enabled or disabled, making it possible for you to set up a default 'IceBBSNEWUSER' account so that new users cannot do *anything* until you have validated them. The 'edituser' command is used to adjust these settings, to adjust the settings for an individual user just type:

```
edituser -mBITS "user name"
```

and the 'edituser' command will walk you through the various functions that can be enabled or disabled.

The second form of privilege control is the number of login sessions per day for a given user. Once a user has reached his account's login limit the BBS will not allow him to connect for more than 30 seconds until the next day is reached.

The third form of privilege control is the maximum process quota (set by 'NPROC' from 'edituser'). This is the maximum number of simultaneous tasks the user can perform on the BBS. Once he has reached this limit the BBS will not allow him to start doing something else until one of the current processes completes.

The fourth form of privilege control is in the download ratios. Through 'edituser' you can assign maximum ratios of downloads to uploads for each user based either on byte counts or file counts. Additionally you can choose to disable these ratios entirely or to suspend a particular user's ability to download anything at all.

34.2.8 Support Utilities

Currently there are a number of support utilities for the IceBBS SIG system. These are 'trim', 'newsin', 'export', 'import', 'uuexport', 'inmail', 'outmail' and 'updatesigs'. some of which are described in the USENET (see Chapter 30 [UUCP], page 113) and FidoNet (see Section 29.1 [FidoNet], page 105) chapters. At the user's end there is the IceOLR off-line reader.

34.2.8.1 Trim

'trim' is a tool to selectively delete old files from the IceBBS message tree on your system. This is commonly referred to as an aging process.

This program first builds a list of the SIGs to be processed and then trims each sig in the list. The list is built by:

1. first loading the 'S:IceBBS.NEWSIGS' file,
2. if the names of specific SIGs are included on the command line then it will mark only those (out of all the SIGs in the SIG list) for processing.

34.2.8.2 Updatesigs

This program is used by the sysop to create the necessary files for all the SIGs in the system. Typically he runs this once when he has changed the 'S:IceBBS.NEWSIGS' (see Chapter 14 [IceBBS.NEWSIGS file], page 51) file to include and new sigs. Old sigs should just be deleted when removed from the 'S:IceBBS.NEWSIGS' file. The command takes no parameters.

called 'IceBBSNEWUSER', if this exists then the BBS copies the contents of the new user account into the newly created account file. If the 'IceBBSNEWUSER' account does not exist then the BBS uses its own built in set of default settings. If you wish to have complete control over what a new user can and cannot do you should set up a 'IceBBSNEWUSER' account, which you can do by logging in under that name or by using the account creation facility of the 'mail' program.

- `-uUSERNAME` Can be used to specify the user name,
- `-eEDITOR` Can be used to override the editor name that was set by the icon,
- `-pPASSWORD` Can be used to specify the password,
- `-?` Displays a help screen.

The mail command can also be used to create new accounts as needed³. One use for this is to create the 'IceBBSNEWUSER' account. Another is to setup accounts for users (under a pre-arranged password) before they call. To do this just issue a command like:

```
mail "-uNew User Name" "-pHis password"
```

note that if there are spaces in the user's name or password then the command line switch and parameter must be enclosed in quotation marks as shown above.

³ This functionality should really be part of the 'edituser' program.

type files and is really the vestigial remanant of the email subsystem's original file transfer method, so it may be deleted in future releases.

The system email directory is also used to store the users' email profile files, these are files that the individual users create and then place online for other users to view. These files are typically used to place a description of the user, his interests and skills in a readily accessible place. They are also useful places to publish a public key for an encryption program such as 'PGP'. These profile files are formed from the account's file number by appending a letter 'P'. These files are plain ASCII, no special care needs to be taken with regard to their contents.

36.2 Between users on the BBS

When a user selects the email tool in the IceTERM program the BBS will start sending whatever mail is currently in the user's mail directory. The messages are sent one at a time to the terminal program and are displayed in a scrolling list window. Any files (of the junk mail type) that are waiting to be sent will get sent simultaneously via the regular file download server.

When the user attempts to send email or files to another user on the BBS he will be presented with a list of users to choose from. Initially this list is empty, so the user will have to request the current list (or a part of it) from the BBS before being able to pick a name.

Since the emailed files directory for each user is accessible via the regular get files tool of IceTERM it is possible for a sysop to just copy files directly into the directory. These files can be in any format that the get files tool understands, plain AmigaDOS, regular IceBBS format, or even the indirected IceBBS format. The easiest way of doing this is to use the 'email' tool as this will take care of finding the correct directory name for you; however, with care the AmigaDOS copy command could be used to perform a mass mailing, for example if the system email directory is 'dh1:usermail' then one might be tempted to try:

```
copy junk_mail.LZH dh1:usermail/#?F
```

to copy the file 'junk_mail.LZH' into every user's email files directory. This, does not work because the copy command does not allow a wild card on the destination. However, a simple script file to do the same could be built and executed:

```
list >temp dh1:usermail/#?F dirs +
    lformat="copy junk_mail.LZH %p%n"
execute temp
```

Note that only those users who have received emailed files in the past will actually get this mailing.

In a system that uses the ‘TrapDoor’ software an additional program called ‘TrapPoll’ should be used to handle automatic delivery of mail to other BBS systems.

See Section 29.1 [FidoNet], page 105 for further information.

36.4 Internet mail

The IceTERM program’s email facility can also be used by a user to send, receive and reply⁷ to Internet email messages. As far as the user is concerned these are much the same thing as mail between users on the same BBS, with the exceptions that he must type in the name and Internet address of the user he is sending mail to (for replies this is done for him), and he can optionally enter a subject. In order for the BBS to supply this function the ‘inmail’ support program (see Section 30.3.3 [InMail], page 115) must be run at various times of the day and UUCP must be installed and the appropriate ‘sendmail’ command must be specified in the ‘S:IceBBS.BINIT’ file, see Section 15.15 [Line 16], page 59.

‘Inmail’ scans the ‘UUMAIL:’ directory, which is where inbound Internet mail ends up, and any mail it finds addressed to accounts on the BBS will be placed in the email SIG.

‘Outmail’ is also needed to process the contents of the email SIG and forward on any new messages it finds there to the appropriate destination on the Internet or FidoNet.

An additional command: ‘fwdemail’ is provided (see Section 30.3.4 [FwdEmail], page 116)so that email that arrives in an IceBBS mail box can be automatically passed on to another Internet mail box, anywhere on the Internet. This was added at the request of a number of users who preferred to have their mail delivered to work or to their machine at home.

36.5 Support Commands

These are the additional IceBBS commands for supporting the email system.

36.5.1 CheckMail

This command is used to check if there is any email waiting on the system for a particular user. Syntax is:

⁷ As was the case for FidoNet email a user can still send messages directly from IceTERM; however, any replies to these messages will be placed in the email SIG only.

- a Will cause the message (or file) to be sent to all users, any users who are named on the command will receive two copies (not implemented yet),
- f The data file the command is to send is not a message and should be sent via file transfer, a copy of the file will be placed in each user's emailed files directory so they can download it at their leisure with the get files tool,
- F The data file the command is to send is not a message and should be sent via file transfer, this sends the file as soon as the user accesses the email tool. This is useful for sending junk mail files. The user cannot resume the download of files sent in this way so its not a good idea to send something important this way,
- m The data file the command is to send is a file containing a message so send it as an email message,
- r Use this directory for the email (overrides 'S:IceBBS.BINIT'). This is intended for testing (for example you might want to copy the entire email directory and then work with a copy of it so that you do not distroy any user's unread mail).
- s Use this to set a different name for the UUCP 'sendmail' command, if not set this will default to 'sendmail',
- u Use this as the user name for reading and sending mail (it overrides the icon file's 'USER=' field). This is useful if you want to test mail and you want to have several sysop accounts.

When the command is run *without* specifying one of the '-f', '-F' or '-m' flags then it will display any waiting mail for the given user account. This is something to avoid doing if you are running the 'email' tool through the remote sysop shell since this will open the editor which will be impossible to exit.

As a sysop you may actually need to receive email under several different names. The easiest way to do this is to run the 'email' command from a script file similar to this:

```
email
email "-ustephen vermeulen-newsletter editor"
email "-uStephen Vermeulen"
email "-uthe sysop"
email -usysop
```


HAILSTRING=

sets the hailing string that will be printed to the conference at the beginning of each of your messages,

TAILSTRING=

sets the tailing string that will be printed to the conference at the end of each of your messages.

38.3 Potential Problems

There are several potential problems with the conference system. None of them are dangerous, but you should be aware of them:

- If an ASCII mode user enters the conference but does not leave it and his modem is not set to hang up when he exits his terminal program, the result can sometimes be lots of garbage in the conference,
- An ASCII conference user may experience some line noise and this may just happen to turn everyone's postings to greek letters. This can be fixed by someone typing a 'CTRL-0' into one of their conference postings,
- Regular IceTERM users may sometimes mess things up by setting up strange options (like making both the text colour and the background colour the same) in their hail and tail strings.

These access levels work the same way that the access levels for the file system work.

The last two numbers specify a security group number and a level within that group that a user must have before he can use the *moderator* functions in the terminal program's SIGs tool. The moderator functions are things like message move and delete.

14.2 The Email SIG's Name

If you want your BBS to provide email you *must* have a SIG called 'EMail'. This name is special and it cannot be changed as this SIG is structured slightly differently internally. Also all the other sysop tools that create and process email must be able to locate this SIG to do their work.

In the example I have given everyone access to the email SIG (by setting at least one access level to zero). This is not strictly necessary, if you want to run a BBS where access to email is only available to certain groups of users you can just adjust the access levels.

14.3 SIGS Access Control

The IceBBS SIGs system has a multilevel access control facility that works on a SIG by SIG basis. Each SIG has a variable number of access level numbers which should be between zero and 255. These are the access levels needed to download messages from this SIG. The access control is the same scheme used for controlling access to file directories, that was discussed earlier (see Chapter 7 [Access control], page 33). Briefly, the BBS SIG system will compare the access levels set in the user's account file to the levels defined for a particular SIG in the 'S:IceBBS.NEWSIGS' file, if there is at least one case where the user's level is greater than or equal to the corresponding level in the 'S:IceBBS.NEWSIGS' file then the BBS will allow the user to read and write to that SIG.

In the above example 'S:IceBBS.NEWSIGS' file the message areas: *Amiga* and *General* will be available to everyone, since all users have at least access level zero in all their groups. If a user has been given access '6 1 3 0' then he will also be able to read the 'Test_SIG' but not the 'News' SIG since none of his levels are high enough to access that group. Note that the 'General' group is available to all users, regardless of what their access levels are set to, since the third security group's level has been set to zero and all users will have at least that level.

The above example shows only four access levels per SIG. For versions of IceBBS prior to 47 only four levels were allowed for SIGs. Versions 47

for each user connection—and will contain the time and user name name of each login and logout event along with the connection modem string and the baud rate. It will also identify ASCII mode login events. And the commands that ASCII users issue may also be logged.

15.2 Line 2

This line contains a single number which specifies the number of conference messages that will be kept on the system. These messages are kept in RAM so if you are tight on RAM keep the number down. The maximum space taken by a single conference message is 256 bytes. See also Section 38.2 [Confer], page 173.

15.3 Line 3

This line contains the name of a directory where the various IceBBS error message files are kept see Chapter 22 [Error messages], page 85. If you wish to customize the error messages you can edit these files.

15.4 Line 4

This line specifies the name of the directory in which the user account files are kept. There will be one file created per user, but since these files are only² accessed by name the number of files in this directory is not important. Note that it is not a good idea to store other files in this directory since the ‘edituser’ program (see Section 25.1 [Edituser], page 95) will try to process them.

15.5 Line 5

The fifth line contains the directory which will be used to hold the user email directories. The BBS software will create the individual user directories as it needs them, but this directory must be in existence when the BBS is run. The individual user directories are given the same names as the user account files. The ‘edituser’ and ‘checkmail’ utilities can be useful when trouble shooting user email problems.

² With the exception of the ‘edituser’ program.

AREXX

Door icons *must* have both the ‘MENU’ and ‘SERVER’ tool types present for the door client to be displayed in the user’s list for selection. The ‘STACK’ will default to 8000 bytes if not present and the ‘PRIORITY’ will default to 2 if not present.

‘STACK’ sets the number of bytes that the door server or client’s stack will be set to when the door server process is spawned by the IceBBS package. ‘PRIORITY’ is the *communications channel* priority (not the AmigaDOS task priority) that the door’s client and server will use to communicate. The priority is user selectable in case the user decides at doors are more or less important than another BBS task.

The ‘MENU’ is used by the user to specify the name of the door that will be displayed in the selection menu. You should set an informative default string here (eg. *Multi User Dungeon, Abandon Hope All Ye Who Enter Here* etc.), the user can change this to something else if he prefers.

The ‘SERVER’ tooltype is used by the sysop and user to define the name of the server that is to be launched on the BBS when this client is started. This allows the user and sysop to change the actual file names (in case there is a name conflict) while still allowing the correct server to be launched. This tooltype is only required at the user’s end of the connection.

The ‘ACCESS’ tooltype is only needed in a server icon, the BBS checks for this tool type to determine the access control group and the minimum level required in that group before the user is allowed to use the door. Each ‘ACCESS’ tooltype defines the access level required for one particular group, but as many ‘ACCESS’ tooltypes as are needed by the sysop may be included. Note that any groups that do not have access levels defined for them default to requiring level 255 (full sysop access) before access is granted.

The ‘AREXX’ tool type is used to indicate that the server or client is an AREXX program rather than a compiled C door.

41.3 Theory

Once the user selects a door client from the list on his system his terminal program will ‘LoadSeg()’ and ‘CreateProc()’ launch that client. IceTERM sends the client a startup message that emulates the message that WorkBench sends any programs that it launches. This was done so that you do not have to do anything special to compile a IceBBS door program. Once the client is running it sends the BBS a message telling it what server to load and start.

15.11 Line 12

Line twelve specifies the directory used for auxillary *hashed access* control files (see Section 33.3.8 [HashAccess], page 138). These can be used to add access restrictions to directory trees located on read-only media. The ‘bulksplit’ and ‘bulkjoin’ tools can be used as an alternative to this.

15.12 Line 13

Line thirteen specifies the directory used to store additional file descriptions for files that are located on read-only media. The ‘bulksplit’ and ‘bulkjoin’ tools can be used as an alternative to this.

15.13 Line 14

Line fourteen specifies the directory that is used to store the door server programs (see Section 41.1 [Doors], page 179) and their icons. It is a good idea to make this directory accessible from the WorkBench (as a drawer icon), this way a new door can be installed at any time by just dragging its icon into the drawer and the tool types that allow you to control the doors³ are easily accessible.

15.14 Line 15

Line fifteen contains a single number, which is the number of blocks that the BBS reads at a time from disk when sending files to users. The blocks are 200 bytes (or so) each, so a value of 50 will cause the file sender to load about 10K bytes at a time. If you are in a heavily loaded system and have spare memory, increasing this number will help. Where increasing this number really helps is when you have a multi-line BBS with CDROMS attached (see Chapter 44 [CDROM drives], page 189), since CDROMS are such slow devices (when seeking) you want to minimize the number of seek and read operations. This is especially true when you are using a *CDROM jukebox* type drive⁴ where the disk that is currently in the drive may have to get swapped often.

³ Particularly the access control tool type.

⁴ The DRM-600 and DRM-604X from Pioneer are examples of this drive type.

archiver does not stop and wait for a confirmation from the user² as this will cause a locked up process on the BBS machine. If you want to allow the co-sysops to use several archivers you just supply additional 'ARC=' tooltypes. If you leave a particular option empty the user's door will not allow him to use that feature of the archiver.

Note that some archivers do not have a list option (like those that pack entire disks) and their extract option may not be useful either since they usually want to unpack to a floppy drive.

Here are some sample settings for the 'ARC=' tooltypes:

```
ARC=lz      |t|l|-m x|
ARC=lharc   |t|l|-m x|
ARC=zoo     |-test|-list|x//|
ARC=arc     |t|l|x|
ARC=unzip   |-t|-l|-x -o|
ARC=dms     |test|||
ARC=zoom    |check|||
```

It appears that the server part of 'remfile' can only run an archiver if the archiver is present in the 'C:' directory. Under AmigaDOS 2.0 this may be different if the WorkBench was started after your path commands.

41.5 The Preview Door

The 'preview' door allows users to preview pictures before deciding to download them. Since full pictures take a long time to download the only feasible way of providing a preview capability is to reduce the size of the picture before sending it to the user. If a picture is compressed both in height and width by a factor of 4 this will result in a factor of 16 savings in transmission time.

As this image reduction process can take a lot of time (especially on an unaccelerated Amiga) and not all pictures can be compressed the same amount and still be recognizable it was decided that the images should be precompressed. A tool such as ASDG's 'AdPro', the shareware 'HamLabPlus' or the freely distributable 'Wasp' and 'FBM'³ can be used to prepare the minaturized pictures for previewing.

The only restrictions on the previewable pictures is that they should be smaller than 320 pixels wide and 200 pixels high and they cannot use HAM mode (they can use the 64 colour EHB mode). In order to

² As 'lharc' does by default when it must create a subdirectory.

³ The Fuzzy Bit Map image processing tool package that is available on the Fish Disks.

modems. The importance of using handshaking with the higher speed modems cannot be overemphasized,

- this selects the type of modem that is being used, it may be either 'auto' for the normal auto-baud rate changing modems (ie. when the modem gets a 'CONNECT 1200' it expects to talk to the computer at 1200 baud), or else it is 'fixed' for the more advanced modems which can keep the computer to modem serial link at the same speed regardless of the actual modem to modem connection speed,
- this specifies the modem result code that is used to enable the faster IceBBS protocols for MNP or V42 connections. The 'REL' is for a GVC or older Supra modem, and the 'ARQ' is for a US Robotics Dual Standard or newer Supra modem. To disable the high speed protocol all that must be done is to set this field to some string (such as 'XYZ') that the modem does not return. Currently⁵ it is recommended that you do not use this high speed protocol, so set this field to 'XYZ',
- this is the first of two strings that are to be sent to the modem to reset it, no spaces are allowed, up to 63 characters long,
- this is the second of two strings that are to be sent to the modem to reset it, no spaces are allowed, up to 63 characters long,
- this is the name of the directory where FidoNet email messages sent by users will be placed until the 'email2cut' program is run,
- this is the name or other identifier of this line. This will be displayed by the monitor tool. Normally one would put either the phone number or type of modem here. This can be up to 31 characters long, no spaces are allowed.

15.16.2 Additional fields

The remaining fields on the configuration line depend on the additional functions that this line must perform. There are currently three distinct modes of operation available as illustrated in the example.

15.16.3 Example line 17 (Use with TrapDoor)

Line 17 is the case for a system that is using 'TrapDoor' (or 'WPL'⁶) to answer that line:

```
10200 4800 -2 IceBBS2
```

The first two numbers are the time and duration of a *bbs not available period*, this starts at 10200 seconds past midnight (02:50) and runs for

⁵ v54 of IceBBS.

⁶ WPL replaces the now obsolete 'welmat'.

41.9.2 Installation for sysops

Drag the ‘FTPserver’ icon into your BBS doors directory. And that’s about it. You may want to adjust the access levels to the server if you only want certain groups to have access to it. This is done using the ‘ACCESS=’ tool type that is standard to all IceBBS doors.

Of course, you will also need to have AmiTCP 4.0 (or higher) installed and running. And you will need some sort of connection to the Internet running (perhaps through PPP), see Section 31.4 [Internet tools], page 122 for more information.

41.10 Internet relay chat

41.10.1 What is IRC?

IRC – Internet Relay Chat – is a real-time, text-based, conferencing system for the Internet. It allows a large number of people to chat at once about a large number of topics. It is real-time because within a few seconds (usually) of you typing a new message into a channel it is visible to all the other people who are viewing that channel.

Typically there might be five to ten thousand people connected into the IRC at any one time. There might be several thousand different topics being discussed, some might only have one or two people, some might have fifty.

IRC has been extended to provide lots of other services (like file transfer) over time. The current version of IceIRC only does the basics, but it’s enough to start.

41.10.2 Installation for IceBBS sysops

Drag the ‘IRCserver’ icon into your BBS doors directory. And that’s about it. You may want to adjust the access levels to the server if you only want certain groups to have access to it. This is done by editing the ‘ACCESS=’ tool types in the icon.

Of course, you will also need to have AmiTCP 4.0 (or higher) installed and running, and some sort of connection to the Internet running (perhaps through PPP). Section 31.4 [Internet tools], page 122 has more information about this.

The last number is the time of the day at which the script file *must* be run and any user on that line will be preempted to do so. This has been set to 00:15, as it was not important in this application.

The last parameter is the name of the script file that is to be run.

15.16.5 Example line 19 (a normal line)

Line 19 is an example of an *ordinary line* which does not do any running of scripts and for which the BBS (not a front door program) manages the modem.

```
-1 -1 -1
```

in this case the last three numbers are just ‘-1’.


```

assign _Audio: AUDIO: defer
assign _Graphics: KM_GRAPH_01: defer
assign _AMINET: AMINET_0693: defer
assign _FishAndMore1: Fish_and_More:fish defer
assign _FishAndMore2: Fish_and_MoreII:fish defer
assign _AB20: AB20: defer

```

where I setup assigns for 9 CDROMs (I only have one drive), then in my 'S:IceBBS.DIRS' files I list all nine disks, thus:

```

_Minis:
_GIFS:
_FishOnCD:
_Audio:
_Library:
_Graphics:
_AMINET:
_FishAndMore1:
_FishAndMore2:
_AB20:

```

and the BBS software takes care of only showing the one assignment for the one disk that is actually in the drive. At any time (appart from when someone is downloading from the disk that is currently in the drive) I can change CDROMs and the BBS software will pick this up and only show the assignment for the new disk.

The 'S:IceBBS.BINIT' file contains additional configuration controls that can be used to reduce access conflicts when multiple users are accessing the same CDROM at the same time. Line 15, see Section 15.14 [Line 15], page 58, allows one to specify the number of blocks that will be read from the disk at a time when a file is being downloaded. Also there is an arbitration system that is very useful on juke box style CDROM drives, see Chapter 18 [IceBBS.ARBITRATE file], page 67 for more information.

Currently there are six handlers that make CDROMs accessible to AmigaDOS. These are described in the following sections.

44.2 CDTV

Commodore's CDTV product (and the A590 add on for the A500) can read CDROMs. One might think that this is not a viable solution for a BBS since it would be difficult to run a BBS on a CDTV machine²;

² There is a company that makes a product called SCSI-TV that adds a SCSI port to a CDTV, hence it might actually be possible to run an entire BBS on CDTV.

(bundled with two CDs of freely distributable software sorted by subject). I have tested this on GVP Series II controllers with the 3.07, 4.0 and 4.5 FaaastROMs and two versions of the Sony drives and the Pioneer DRM-600 and DRM-602 and have found it to work flawlessly.

44.6 AsimCDFS

This is into its second major release, Ian Gunn has been running (and beta testing) this software for about a year now on his Great White North IceBBS. Currently he is running 7 CDROM disks off a GVP Series II controller, one disk is in a Chinon drive and the other 6 are contained in a Pioneer DRM 604 drive (this is a 6 disk changer type drive).

44.7 AmiCDROM

This is a freely distributable CDROM file system that is available through most good BBS systems. It comes with source code in C.

AmiCDROM is a CDROM disk filing system for the Commodore Amiga. It supports the ISO-9660 standard, the Rock Ridge Interchange Protocol and the Macintosh HFS format.

The author can be reached at the Internet address:

`ln_fm@pki-nbg.philips.de`

This package has been used on the AMUC Express BBS to drive a DRM-600 and a DRM-602⁴ with great success. In fact it seems to do the same job that Xetec did while using much less memory.

The commercial packages still have better support for some of the other uses of CDROM, such as for PhotoCD and for playing CD32 and CDTV games.

44.8 CDROM Sources

If you know of any good sources for CDROMs of public domain, freely distributable or other software that can be legally placed online please let me know and I'll add them to this section.

44.8.1 Walnut Creek

Walnut Creek specializes in publishing CDROMs that contain images of some of the large Internet FTP sites. Some of these are updated quarterly and reduced price subscriptions are available.

⁴ Both are 6 disk jukebox drives from Pioneer.

Aliases also have a primitive form of parameter substitution, any additional text that the user types after the alias command will be saved and then appended to the alias substituted command allowing you to alias the 'dir' command as 'd' and then still allowing the user to type 'd df0:' to effect a 'dir df0:' command.

The syntax is:

```
alias perm|temp user substitute
```

where, 'alias' is the name of the command, followed by either a 'perm' or 'temp' keyword to define whether this is a permanent alias or a temporary one. The 'user' string (a single word only) is a command name that the user will type to trigger the alias substitution process and the 'substitute' string is a IceBBS command that is to be substituted for the user command. Note that the 'substitute' string can contain white space, allowing you to specify any additional parameters that the command might normally take.

Note that any alias can only execute one IceBBS command, if you need to execute a series of commands (for example to change to a certain directory and then start an xmodem download of a particular file), you can put all those commands into one menu file and execute the file with a 'goto file' command in the alias.

21.3 bye

Causes the BBS to log the user out and hang up the line.

21.4 cd

The 'cd' command allows the user to walk around the BBS file directory tree, in a similar fashion to the AmigaDOS 'CD' command. The 'cd' command respects the access limits that the sysop has placed on the file tree. It only allows the user to start in the directories listed in the 'S:IceBBS.DIRS' file and will allow him to step into any subdirectories of those root directories, but only if he has the appropriate access level.

The current implementation only allows one level of directory change per command. It will also trap all attempts to use the ':' character and will prevent the user from accessing private parts of the system through excessive use of the '/' character.

21.5 chat

Not implemented yet.

the file (in bytes), and the long description (if any). If the file does not have any of this information attached to it, the BBS then displays the 'nofiledesc' error message file¹ to the user.

21.9 dir

The 'dir' command displays the list of files and subdirectories in the current BBS directory. This command takes five optional parameters (all can currently be abbreviated to their first letter),

time	shows the AmigaDOS date and time stamp for the file,
size	shows the AmigaDOS size (in bytes) of each file, note this may be slightly larger than the actual file size reported by the 'describe' command,
dirs	shows only subdirectories within the current directory, no information about files are shown,
files	shows only the files within the current directory, no information about subdirectories are show,
new	causes all the items to be sorted in date order (newer files are displayed first, allowing the user to use a 'CTRL-C' to abort the listing once old files are being listed.

21.10 display

This function displays a file to the user, it will only show files that are in the ASCII menus directory so the user cannot wander around your system displaying files at will. Note that this means that any user can display both the menu definition files and the files that just contain text for display purposes. If this poses a problem² then you can use the 'lock' command to prevent the user from executing any command except those that are currently defined by a temporary alias.

21.11 download

This command takes a file name as its sole parameter and starts an Xmodem³ download of that file to the user's machine. The user can

¹ Found in the ASCII menu files directory.

² It might be best for me to just create a separate directory for files that will be displayed by the 'display' command

³ Xmodem is the only file transfer protocol hard coded into IceBBS's ASCII mode. Support for other protocols (including ymodem and

first four correspond to the normal WorkBench colours³. To set the background colour to white the user would send the sequence: 'CSI 41m'.

Note that multiple effects can be selected with one escape sequence such as: 'CSI 1;33;40m' which selects boldface with colour 3 as foreground and colour 0 as background. And that to reset everything to the default settings you need to send a: 'CSI 0;31;40m' sequence.

47.2 Amiga ANSI Codes

Here is a little manual on Amiga ANSI codes that I got from a FidoNet message⁴, it may prove useful.

47.2.1 Cursor Positioning or Movement

To move the cursor to a specified position: 'ESC [#;#h' where the first '#' is the desired line number and the second the desired column. To move the cursor up without changing columns: 'ESC [#a' where '#' specifies the number of lines moved. To move the cursor to a specified horizontal and vertical position: 'ESC [#;#f' where '#' means first the line number and the second '#' the column number. To get a cursor position report: 'ESC [#;#r' where the first '#' specifies the current line and the second '#' specifies the current column. To move the cursor down: 'ESC [#b' where '#' specifies the number of lines moved down. To move the cursor forward: 'ESC [#C' where '#' specifies the number of columns moved. To move the cursor backward: 'ESC [#d' where '#' specifies the number of columns moved. To save the cursor position: 'ESC [s' and to restore it: 'ESC [u'.

47.2.2 Erasing the Screen and Erasing a Line

To do a CLS (erase screen move cursor to home position): 'ESC [2j'
To erase from cursor to end of line: 'ESC [k'

47.2.3 ANSI Colour Graphics

To set the colour/graphics attributes, enter 'ESC [#;#m' where the first '#' is the desired foreground colour and the second is the desired

³ Under AmigaDOS 1.3 these are: blue, white, black, and orange

⁴ My thanks go to Walt Sullivan for posting this message from: *Why a duck?* FidoNet node: (1:163/109.4).

A typical set of aliases for the message system might be:

```
alias temp n feedback next
alias temp p feedback prev
alias temp g feedback goto
alias temp l feedback last
alias temp f feedback first
alias temp a feedback again
alias temp e feedback enter
alias temp r feedback reply
ctrl temp m feedback next
ctrl temp p feedback prev
ctrl temp f feedback first
ctrl temp l feedback last
```

This set binds all the various 'feedback' command functions to single ASCII characters, as well it implements a number of them as control keys, including read next message as the return key.

21.14 goto

This command takes one parameter, the name of the new menu file that should be run. The ASCII menu system allows the user to move between menus as he pleases, subject to the links between them that the sysop has defined.

If a 'goto' command appears in a menu file where it will be executed it causes IceBBS to immediately change to the next menu file. This can be useful when you want to establish a few default aliases and then switch to a common menu file. An example of this is file which is the first menu file executed upon ASCII login on my BBS:

```
alias perm logout bye
alias perm lo bye
alias perm quit bye
alias perm help display help
alias perm new goto new
alias perm main goto main
alias perm term goto getterm
display welcome
login
goto main
```

Additionally, the 'goto' command will often be found in 'alias' commands where it will be executed when the user decides to select a different menu.

- '44' blue background
blue background
- '45' magenta background
white background
- '46' cyan background
black Background
- '47' white background
orange background
- '48' unknown
unknown
- '49' unknown
unknown

NOTE: Please note that for both the foreground and background sections, the colours repeat after each four codes; ie: blue text = 30 or 34; blue background= 40 or 44. (unless you use a WorkBench or screen with more than 4 colours) Also, as long as you know the escape code for the foreground colour, the same colour for background will be 10 numbers higher.

47.2.4 Additional ANSI Attributes

To set additional attributes enter: `'ESC [#m'` where `'#'` is the number of the desired attribute. Select attributes from the list below: 0 all attributes off (defaults to original Preferences settings)

- '0' all ANSI attributes off
all ANSI attributes off
- '1' bold on
bold on
- '2' unknown
turns text black
- '3' Italics on
italics on
- '4' underscore
underscore
- '5' blink
does not work on amiga

21.18 password

This command takes no additional parameters and will allow the user to change his password. It will first prompt him for the old password and then prompt him for the new one twice. The password will only be changed if the old password was entered correctly and the same new password was entered twice.

21.19 prompt

This command takes one parameter, the name of a file that is |o be displayed to the user everytime he executes a command in the current menu. This is the file that you use to define the actual menu text that is displayed to the user. An example of a prompt file might be:

```
----- IceBBS Main Menu -----
Return here with "main"   Get help with "help"
      m = feedback messages
      f = files           term to get TERM
      ! = conference     bye to logout
      new = goto new files  quit to logout
-----
```

Note that you should keep these files short, since they are going to be sent to the user a lot of times, and if your editor only ends lines with line feed characters you should probably add carriage returns to the line ends as well.

Also note that if you wish to have multiple sets of menus, perhaps for new users, advanced users and special groups you can. All you must do is have an upper level menu that allows the user to select the set he wants to work with and then use a `'goto'` command to jump to the first menu in the appropriate set.

21.20 protocol

The new XPR upload-download system requires that you install some XPR libraries (of the protocols you want to run—such as zmodem) in your libs: directory. These libraries always start with `'XPR'` and include these:

- `'xprxmodem.library'` for XModem file transfers, this may work better with some PC terminal programs than the XModem that is built into IceBBS,
- `'xprymodem.library'` for YModem file transfers,
- `'xprzmodem.library'` for ZModem file transfers.

prev non-threaded read previous message,
high return to high water mark,
first jump to the first message in the current SIG,
last goto last message in the current SIG,
up threaded reading: go up a message (ie. to the parent),
down threaded reading: go down a message (ie. to the child),
thread threaded next message,
back threaded backup (reverse or previous) message,
goto goto a message by date (not yet implemented),
enter enter a new message,
reply reply to the current message,
again type the current message again,

To maintain the users subscription list these are the commands:

sub add signame
 add the named SIG to the subscription list,
sub drop signame
 remove the named SIG from the subscription list,
sub full shows the full list of SIGs the user has access to on this BBS,
sub shows the list of SIGs the user has subscribed to.

To move between sigs:

change next
 goto next SIG in subscription list,
change prev
 goto previous SIG in subscription list,
change to signame
 change to SIG by name.

It is recommended that you build a set of menus that bind these commands through a series of simple prompts to keys using the 'alias' (see Section 21.2 [ASCII alias], page 73) and 'ctrl' (see Section 21.7 [ASCII ctrl], page 75) commands.

21.22 term

Not yet implemented.

21.29 xup

This is the command to upload a file using the currently selected XPR protocol (see Section 21.20 [ASCII protocol], page 80) from the user to the BBS. Note that if the XPR protocol supports wildcards then they can be used to transfer a batch of files.

CRC	Cyclic Redundancy Code, a form of checksum (or a finger print if you will) that is computed from the contents of a file. This is used to detect errors in transmission of the file, by recalculating the CRC once the file has been completely received and comparing it to the original CRC that was transmitted. CRCs can also be used to detect virus infections and other forms of file damage.
CSLIP	Compressed Serial Line Internet Protocol, a version of SLIP that <i>compresses</i> the headers of the packets that it carries between the two computers it connects.
DS	The Dual Standard modem from USR. This is a modem that supports both USR's proprietary HST protocol and the international standard V.32 and V.32bis protocols. Hence its "dual standard" name.
DTR	Data Terminal Ready, a line that indicates to the modem when the serial port it is connected to is live.
Email	Electronic Mail, messages and files that are sent electronically from one person to another, usually by use of some kind of a network or BBS machine.
FidoNet	This is a file transfer protocol which is often used to network a number of BBS machines so that they can transfer files and messages. It is also the name of the oldest and largest network that is running the FidoNet protocol.
FTP	File Transfer Protocol, after more than 20 years of use this is still one of the most popular ways of moving files between computers on the Internet. Sort of the X or Zmodem of the Internet.
Internet	A large meta-network of computers, mainly main frames and workstations, that are linked together 24 hours a day through dedicated lines. Personal computers are starting to appear on the fringes of this network, usually they are linked in by dial up lines running UUCP type software.
IRC	Internet Relay Chat, a real-time conferencing system that allows users all over the world to chat between themselves over the Internet.
ISP	Internet Service Provider, an entity who provides others with access to the Internet, usually for a fee.
LF	Line Feed, this is ASCII character 10, it is a character that a computer sends a printer to tell it to advance to the next line. Its original meaning has become a bit garbled, and on

exceededbratio	this is sent when a user attempts to download a file that would exceed his bytes downloaded to bytes uploaded ratio, see Section 7.7 [Privileges], page 36,
exceededdaily	this message is sent to the user if he attempts to start a new download when he has already downloaded more bytes in this day than his account's daily download limit allows, see Section 7.7 [Privileges], page 36,
exceedednratio	this is sent when a user attempts to download a file that would exceed his number of files downloaded to number of files uploaded ratio, see Section 7.7 [Privileges], page 36,
exceededquota	this is sent when a user tries to start more tasks than he is allowed to, see Section 7.7 [Privileges], page 36,
filenotdeleted	this is sent when a user attempts to delete a file he is not allowed to. Normally users can only delete files that are in their private emailed files directory,
imposter	this message is sent to any user who attempts to access IceBBS under an existing user's name, but with an incorrect password, see Section 25.1 [Edituser], page 95,
linenotopen	this is sent when a user tries to connect to a line that is currently in its shut down state, for example when it is FidoNet mail hour, see Section 29.1 [FidoNet], page 105 and Section 15.16.3 [Use with TrapDoor], page 60,
motd	this message is sent to all users (except new users) whenever they log into IceBBS, use it to send a message of the day. See Chapter 23 [MOTD], page 89 which describes the process of producing a version of the motd which is tailored to the user,
newuser	this is a special welcoming message sent to all users new (first time connections), useful to send some helpful hints about the system,
nodlallowed	this is sent when a user who has been denied access to the get files tool attempts to use it,
nodooraccess	this is sent to a user who has tried to use a door to which he has not been given access to,

- Sysop** System Operator, a person who runs a BBS or other computer system, generally someone who looks after the day to day maintenance of the computer hardware and software.
- Tool type** The means by which the Amiga's WorkBench allows users to pass configuration related information into programs. The information is entered into the icon via the WorkBench's 'Information' menu item and is available to the program.
- TrapDoor** A FidoNet frontend mailer package for the Amiga, this is the software that handles the transmission of files between FidoNet systems.
- USENET** A loosely connected network of computers running the UUCP Protocol software.
- USR** US Robotics, manufacturers of high end modems for amateur use, typically the modem manufacturer of choice on BBS systems since USR has a very aggressive discount program for sysops.
- UUCP** Unix to Unix CoPy, strictly this is the name of a program that is used to copy files from one UNIX machine to another, it is also the name of the protocol (as in "the UUCP Protocol") that is used to do this copying. This is the basis for USENET.
- V.32** The international standard set by the CCITT for data transmission at 4800, 7200 and 9600 baud. This describes the means by which a modem encodes the digital data onto the analogue phone lines. V.32bis is an extension of this standard to higher speeds, typically to 14400 buad.
- V.34** The international standard set by the CCITT for data transmission at baud rates between about 21000 and 28800. This describes the means by which a modem encodes the digital data onto the analogue phone lines. V.34bis¹ is an extension of this standard to higher speeds,
- V.42** An international standard set by the CCITT for error detection and correction between modems (ARQ). This is commonly associated with V.42bis which adds a form of data compression to V.42 to allow faster transmission of previously uncompressed data. Note that V.42 and V.42bis can run at any transmission speed, so you will find them on 2400, 9600, 14400 and higher speed modems.

¹ Which does not exist yet, but is expected sometime.

unknownuser

this message is sent by the email subsystem when it finds that the user is trying to send mail (or a file) to a user who is not registered on the system. This can happen if the user is using an old version of the user list and the account that he is trying to send email to has since been deleted,

The only thing about these files that is at all tricky is that the first line of each file must only contain four numbers. These are used to specify the size of the viewing box window that will be opened on the user's end of the connection. These numbers specify the position on the screen of the top left corner of the window (in pixels from the left edge and pixels down from the top) and the width and height of the window also in pixels. An example message file follows:

```
120 10 400 180
#####
#                                     #
#           Welcome to IceBBS, New User!           #
#                                     #
#   As a new user of the system, your               #
#   access level is restricted and your              #
#   access time is limited to 1 hour per           #
#   call.                                           #
#                                     #
#   Also, your message counter has been            #
#   fully reset, so there are about 600            #
#   messages to read... If you just want          #
#   to skip them all, so you only see new         #
#   messages the next time you log in you         #
#   should select the message section, wait       #
#   for the first message to arrive and           #
#   then hang up the modem.                        #
#                                     #
#####
```

Note that the box formed by the '#' characters are actually part of the file that will be sent to the user, you don't have to include them in your own files. Also note that it may be possible to crash a caller's system³ by specifying unreasonable values for the initial viewing window's position and size, so keep those dimensions to within the standard workbench size which is 640 pixels wide and 200 pixels high.

³ This should no longer be possible, so let me know if you ever do this.

```
.key tfile,user
stack 10000
copy dh2:errormessages/motd <tfile>
dh2:bbs/util/motdinfo <tfile> <user> -1
dh2:bbs/util/newsince <tfile> <user> -2 Files:New
```

This script first copies the standard 'motd' file into the custom file, then it calls on the 'motdinfo' command to append some account limit information to the output file (the 'tfile'), and then it uses the 'newsince' command to scan the new files directory and add the names of any files that are new (to this user) to the output file. When this is done the script file will return and the BBS will read the output file, send it to the user (in the form of a standard scrollable text list) and then delete it.

It is certainly possible to write other utilities that add additional information to this dynamically built message of the day. For example a fortune cookie program or additional statistics on BBS usage.

23.2 Motdinfo

This command is intended to be called from a 'makemotd' script and to add some information about the user's BBS account to the message of the day.

23.3 NewSince

This command is intended to be called from a 'makemotd' script to add a listing of the files that are new on the BBS since the user last logged in. It does this by scanning a directory that you specify for files with an AmigaDOS time stamp newer than the last time the user logged in. Any the names of files it finds will be appended to the output file and depending on the level of detail desired file dates and sizes can also appear.

23.4 IsNetUp

This is the command you can use to check to see if a PPP internet link (or other device-type link) is up and running. It is designed to be part of the 'makemotd' script file so that it adds a message about the current status of the network to the message of the day.

Stephen Vermeulen
45 Butler Cr. N.W.
Calgary, Alberta
CANADA, T2L 1Y3

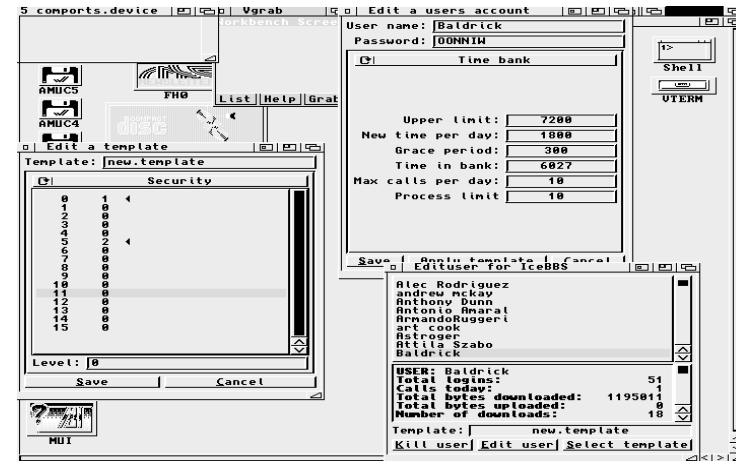
Voice: (403) 282-7990
Email: svermeulen@ragnarok.mtroyal.ab.ca
FidoNet: stephen vermeulen at 1:134/92.0
WWW: <http://www.cuug.ab.ca:8001/~vermeuls/>

50.4 No Warranty

THERE IS NO WARRANTY FOR THE PROGRAMS, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAMS "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAMS IS WITH YOU. SHOULD THE PROGRAMS PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY REDISTRIBUTE THE PROGRAMS AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAMS (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

The one thing that is a bit tricky is the security section—as this is a list. You edit the security area by clicking on the line you want to change and then editing the access level in the string requester at the bottom. To *check mark* the access level so that the apply template function will change that row of a user's account you double click the appropriate row in the list. This will cause a little *left pointing triangle* to appear on the right of the row indicating this item will be applied to a user's account. This is shown in the following figure, where access levels 0 and 5 have been make active in the 'new.template':



Editing an EU Template

The templates are stored in a single directory, if you start 'eu' from the command line you can specify the directory thus:

```
eu tmplt=some_directory_name
```

if you start 'eu' from the WorkBench then you need to include a 'TMPLT=dirname' tool type in the 'eu' icon.

M	
Mail.....	161
make_binit.....	8
makecoupon.rx.....	99
Makemotd.....	89
monitor.....	171
motdinfo.....	90
MUIFind.....	124
mysort.....	127
N	
news.....	79
newsin.....	114
newsince.....	90
O	
outmail.....	116
P	
pass.....	93
pkt2email.....	111
preview.....	182
prompt.....	80
protocol.....	80
R	
read.....	145
RemFile.....	181
S	
safewait.....	110
secho.....	111
send.....	169
sendmail.....	116
setaccount.....	93
setuser.....	93
sig.....	81
Store.....	138
T	
TELNET.....	183
term.....	82
toolkit.....	180
top.....	83
TrapDoor.....	105
Trim.....	160
U	
UnusedCoupons.....	99
Updatesigs.....	160
upgrade.....	83
upload.....	83
users.....	83
uucico.....	114, 115
uucp.....	116
uuexport.....	115
uusplit.....	116
uuxqt.....	114
V	
VAUX-Handler.....	177
VBBSReader.....	145
W	
who.....	83
WPL.....	105
X	
xpr.....	80

CD32.....	191
CDROM access control	189
CDROM drives, using with IceBBS	189
CDROM, and download counters...	189
CDROM, arbitration.....	67
CDROM, assign command	189
CDROM, file systems	191, 192
CDROM, making one.....	139
CDROM, security.....	138
CDROM, slow directories	63
CDROM, swapping.....	67
CDROM, via CD32	191
CDROM, via CDTV	190
CDROM-FS	191
CDROMs, and file format	133
CDROMs, source for.....	192
CDTV	190
Chaining ASCII menus.....	78
ChangeTaskPri	197
Changing error messages	85
Chat, multi-user	173
Checking mail	166
Client	179
Closed line.....	85
Codes, escape.....	199
Colours, in text	199
Commands, defining new.....	73
Comments	1
ComPorts, 8 serial port card	19
Conference, history depth.....	56
Conference, multi-user	173
Conference, problems	174
Conference, sysop's window to it ...	173
Configuration file	55
Connect limits	37
Control of user access to tools.....	36
Conventions.....	1
Cookies.....	217
Copying	213
Copyright.....	213
Coupons, for account upgrades.....	99
Coupons, unused.....	99
CRC, testing files	134
Cursor positioning, text	200
Customizing, MOTD.....	89

D

Data files, for SIGs.....	145
Data set ready	19
Database for files, building.....	127
Databases, building.....	128
Databases, list of	128
Databases, multiple.....	128
Databases, sizes.....	128
Debugging, commands for.....	138
Dedication.....	217
Default account	37
Default, FidoNet node number	71
Defining new ASCII commands.....	73
Defining, ASCII menus.....	80
Definitions.....	207
Deleting file, message	85
Deleting old files.....	137
Description, none for file	85
Descriptions, making lists of.....	138
Devices, null modem device	20
Devices, required.....	23
Directories, access control.....	137
Directories, for downloads.....	63
Directory in use, message	85
Directory, for ASCII menus	57
Directory, for doors	58
Directory, for email	56
Directory, for error messages	56
Directory, for file descriptions	58
Directory, for hashed access control..	58
Directory, for user accounts	56
Disclaimer.....	213
Disk reading.....	58
Distribution	213
Distribution archive.....	7
DMS	140
Dog's body, a.....	97
Door access not allowed, message....	85
Door server not installed, message..	85
Doors.....	179
Doors, ARexx	183
Doors, CircleMUD	183
Doors, picture previewer.....	182
Doors, programming examples	180
Doors, remote account editor.....	183
Doors, remote sysop file maintainance	181
Doors, server directory	58, 179

a time of 25200 (that's 7 hours times 3600 seconds in an hour). If you started running IceBBS at 6 AM then the script file would be called about 1 hour after you started IceBBS. However, if you started running IceBBS at some time after 7 AM the script file would not be run until 7 AM the following day. Essentially this is an offset to the first time the script is run after the BBS is started, beyond that point the second parameter is more important.

The second number is the time to wait between running the script file once and running it again. This number can be as low as 900 (wait 15 minutes between scripts) or as long as you wish. Typically you might wish to call a news feed node 4 times a day (a value of 21600, which is 6 hours times 3600 seconds per hour) or perhaps just once a day (use 86400 seconds). If you want the script file to be run just once a week² you would use 604800 seconds (which is 7 times 86400). This value will not pre-empt a user who is online at the time the script should be run again. The script will be delayed until the user logs out normally, because of this the third number is provided.

The third number is a time at which the line must be shut down and the script run. This is a time in seconds after midnight, so to run a script at 2:45AM you would use 9900. This value is pre-emptive and will cause the BBS to terminate the session of any user who is on that line when this time is reached. Note that this is done by adjusting the users session limit at the time he logs in, so if he is running IceTERM version 86 (or higher) his count down timer will reflect this. This is primarily useful for FidoNet applications where the Fido mail hour must be respected.

Strictly speaking the script name is simply a command name that you could enter at the AmigaDOS prompt. You may also wish to use the script facility to run a single AmigaDOS command, such as 'DIR' once a day. It should also be possible to include command line parameters, in which case you need to enclose the command name and the parameters with quotation marks like:

```
"dir dh2:downloads opt a"
```

Also note that for the script to run you need to be using a shell such as the AmigaDOS 1.3 shell, or ARP's AShell which recognize script files as executable files and you need to have the script files script bit set with the 'protect +s' command. If you do not specify the full path name to where the script file is stored it might not run, depending on your version of AmigaDOS or the presence of ARP³.

² Perhaps to call a remote site to grab new files...

³ Those who are running AmigaDOS 1.3 with or without ARP, or AmigaDOS 2.0 should have no problems.

File system	131
File systems, see also CDROM.....	191
filenotdeleted	85
Files, access control	137
Files, account file format	187
Files, appending	91
Files, BBS log	55
Files, bulk operations on.....	139
Files, CDROM.....	138
Files, configuration.....	55
Files, CRC error test.....	134
Files, for feedback.....	57
Files, joining descriptions	133
Files, joining descriptions and files..	140
Files, making listings of.....	138
Files, names of SIGs	51
Files, on removable drives.....	195
Files, PKT type	109
Files, remote sysop door	181
Files, saving temporary	138
Files, sending in email	163
Files, splitting descriptions.....	134
Files, splitting into parts.....	139
Files, testing.....	140
FILES.BBS, making and using.....	139
Find files, finds wrong thing.....	129
Finding a news feed.....	113
Finding files	123
Font, typewriter.....	1
Fonts, meaning of.....	1
Format, of IceBBS files.....	131
Forwarding email	169
Fred Fish	193
FreeNet	113
Front door programs.....	103

G

Games, multi-user dungeon	183
Games, see also doors.....	179
Get files not allowed, message.....	85
Glossary.....	207
GNU.....	1
Goals	3
Greg Pringle.....	217
Groups, access.....	33
Groups, number of.....	35

H

Hacker.....	85
Handler, for DOS shell.....	177
Handler, VAUX	23
Handlers, see also CDROM	191
Hayes modem.....	25
Help files, see news.....	175
Hibbert, Ian	20
High speed modems.....	26
Hot keys, for ASCII mode.....	75
Hourly time limits	37
HST modem.....	28

I

Ian Gunn.....	217
Ice, meaning of.....	3
IceBBS.FIND.....	128
IceBBS.MSGID.....	45
IceBBS.NEWS.....	175
IceBBS.SHELLACCESS	177
IceBBSNEWUSER file.....	37
IceOLR.....	159
IceTERM, icon problems.....	185
Icon library	23
Icons, problems with.....	185
IDs, see also sequence numbers.....	45
IFF, previewing pictures	182
imposter	79, 85
Indirected files	133
Information highway	119
Install script	8
Installing the BBS	7
Internal modem, Applied Engineering	19
Internal serial port	15
Internet mail not available, message	85
Internet, configuring Email for	59
Internet, forwarding mail to.....	116
Internet, testing status of.....	90
Internet, tools for.....	119
Italics	199

J

Join, or the append command.....	91
Joining files and descriptions.....	133
Jukebox, CDROM drive.....	67
Junk mail, see email	164

News files, format of.....	175
News item missing message.....	85
NewsGroups, see also SIGs.....	143
newuser.....	79, 85
No file description.....	85
No profile for user.....	85
Node number, default.....	71
nodlallowed.....	85
nodooraccess.....	85
nodoorserver.....	85
nofiledesc.....	85
nointernetmail.....	85
noserver.....	85
noshellaccess.....	85
Not an IFF picture, message.....	85
notaniffpic.....	85
nouserprofile.....	85
Null modem.....	20
Null modems.....	20
nullmodem.device.....	20

O

Offline reader.....	159
Old message base, from ASCII mode	77
Origin, FidoNet.....	107
Out of time message.....	85

P

Password, bad.....	85
PATH, FidoNet.....	107
Pausing a line.....	101
Performance.....	5
PGP.....	164
PGP, public keys.....	163
PGP, uses of.....	105
Picture not an IFF, message.....	85
Pictures, previewing before downloading	182
Pioneer DRM604.....	192
Pioneer, DRM-600, 602,604.....	67
PKT files.....	108
PKT files, viewing contents.....	109
Plus character.....	1
Polling script files.....	61
PPP device.....	90
Premastering, CDRoms.....	139
Preview door, not an IFF.....	85

priority, door tool type.....	179
Privileges.....	36
Profile, none available.....	85
Protecting against bad packets.....	105
Protocol, high speed.....	32
Pruning, databases.....	128
Public access sites.....	113
Public key, see PGP.....	163

Q

Quit.....	73
Quota exceeded message.....	85

R

Radio modems.....	119
Ratio exceeded.....	85
Real time chat.....	173
Register settings, for modems.....	25
Remote sysop, file maintainance.....	181
Removable media.....	195
Restart, menu item.....	101
Restricting file areas.....	39
Restrictions.....	213
Root directories.....	63
Root directory, moving to.....	83
RSA, see PGP.....	163

S

Script commands, time of day.....	110
Script file, debugging.....	111
Script files, debugging.....	138
Script files, running at certain times	101
Scripts, nstall.....	8
Searching, for files.....	123
Searching, multiple databases.....	128
Security.....	33, 105
Security groups, numbers of.....	43
Security, files.....	39
Security, of files.....	137
SEEN-BY, FidoNet.....	107
Sendmail, specifying command for....	59
Sequence numbers, for messages.....	45
Serial devices.....	23
Serial port, internal.....	15
Serial port, sharing with TrapDoor..	60
Serial ports.....	15

When the FidoNet 'import' command is run, one of its options is to check the packets before processing to see if they are marked as secure (see Section 29.5.2 [Import FidoNet], page 108). If this option is enabled then only packets that are marked secure will be unpacked, any others will be left alone. The reason for this is that other nodes can still call and send mail, even though they do not have session passwords defined; however, any such piece of mail could contain bogus packets so sysop intervention in the unpacking is advised. If the secure packet check option is disabled then all packets found in the inbound directory will be unpacked and imported into the various message bases.

Note that the combination of session passwording and only processing secure packets is not really enough to protect a system against importing bogus packets. The risk here is that one of the trusted systems with which you have set up a session password could send bad packets containing messages addressed to users on your system or to an echo that system does not normally exchange with you.

The IceBBS support software does not currently provide a way around this problem.

The interested sysop might want to explore other solutions to the security problem, perhaps by using a package like PGP to ensure authenticity. For example the remote system would encrypt any packet it sends with its private key, this ensures that packet can *only* be decrypted by applying the remote system's public key. This guarantees that the packet did indeed come from that remote system.

29.4 Multiple Nets

The 'import' and 'export' commands supply some support for acting as an echo-conference hub (either within the same net or between different zones). The primary limitation to this is that low priority mail can not be transmitted at this time.

29.5 FidoNet Commands

This section describes the commands that are provided with IceBBS to support FidoNet activity. Not all of these are needed for the basic installation (the minimum set is just 'export', 'email2cut' and 'import'); however, the others can be useful for diagnostics and day to day operations. To make a fully functioning FidoNet environment you will also need to install a front end mailer, such as TrapDoor or WPL.

toomanycalls	79, 85
Translator	23
TrapDoor, sharing serial port	60
Traversing links, in SIGs	153
Trim, last trim file	47
Trimming, old files	137
Trimming, databases	128
Trimming, messages	160
Type, ASCII command	76
Typewriter font	1

U

Underlined	199
Unique message IDs	45
Unknown user	85
unknownuser	85
Unused coupons, directory for	99
Updates	213
Updating SIGs list	160
Upgrades	213
Upgrading accounts	99
Uploads, listing new	138
US Robitics, see USR	28
USENET	113
USENET, connecting to	113
USENET, importing email from	115
User account information, displaying	90
User accounts, specifying directory ...	56
User activity, monitoring	171
User profile files	163
User unknown to email, message ...	85
Users, new	37
Users, see also accounts	97, 187
USR Dual Standard modem	29
USR HST modem	28
UUCP, batch directory	115
UUCP, configuring	114

UUNET	113
UUNews:Export, directory	115
UUSPOOL:batch, directory	115

V

V32 modems	30
V34 modems	31
V42 modems	31
VAUX handler	23
VAUX-Handler	177

W

Wait, replacement for	110
Walnut Creek	192
Warranty	213
Wemat, sharing serial port	60
Wireless modems	119
Wish list	205
WorkBench, icon bug	185
WPL, and UUCP	114
WPL, sharing serial port	60

X

Xetec	191
XModem	80
XPR libraries	80

Y

YModem	80
--------------	----

Z

ZIP	140
ZModem	80
ZOM	140
Zone routing	111
ZOO	140

be rescanned the same MSGID lines will be written (see Chapter 11 [IceBBS.MSGID file], page 45).

If there is a configuration problem that prevents ‘`export`’ from running it will return a result code of 5 which can be used to abort the script file. It returns 0 if the run was successful.

29.5.2 Import

This program is designed to handle the FidoNet message importing process. It can unarchive arc-mail⁴ and can unpack ‘*.PKT’ files⁵ and sort through the resulting messages and place them in the appropriate places (either SIGs for echo mail, or a user’s email directory for netmail). Usually this program is run near the end of the FidoNet script (after the TrapDoor activity is finished) but before control is returned to IceBBS.

The ‘FIDOCFG’ file (see Chapter 20 [IceBBS.FIDOCFG file], page 71) needs to be configured for ‘`import`’ to work properly.

‘`Import`’ works by scanning the FidoNet inbound mail directory⁶ for ‘*.PKT’ files. It will parse the messages out of each ‘PKT’ file it finds and for each message that is extracted it will decide whether the message is private netmail (in which case it is placed into the IceBBS email SIG) or an echo mail conference message. If the message is part of an echo mail conference it will extract all the control information and store that in the message’s VMSG header, as well it will figure out the SIG area the message belongs to and put the message into that SIG’s message base⁷.

The program will strip out CR and *soft* CRs and convert them to LFs, as well it will break long lines into LF terminated short lines. Any LFs in the original will be discarded.

If the program finds any archived mail packets in the inbound directory it will unarchive⁸ them one at a time into the TEMPDIR and then will process the resulting ‘*.PKT’ files via the above steps.

All ‘*.PKT’ files and archive files that are successfully processed will be deleted.

If you are receiving an echo mail conference feed then there is another feature of ‘`import`’ that will be of interest. As ‘`import`’ unpacks messages and places them into the correct SIG directory it will check each message

⁴ Also called compressed mail.

⁵ Uncompressed mail bundles.

⁶ TrapDoor usually uses ‘MAIL:InBound’ for this.

⁷ SIG message bases are stored in two files, the actual text and header information goes into the ‘.dat’ file while the message linkage information goes into the ‘.lnk’ file.

⁸ Using the archiver specified by the ‘-a’ command line switch.

40	The Remote DOS Shell	177
41	The Doors System	179
42	IceTERM icon problems	185
43	Format of Account Files	187
44	CDROM Drives	189
45	Removable Hard Disks	195
46	Using ChangeTaskPri	197
47	ANSI Codes	199
48	Future Enhancements	205
49	Glossary	207
50	Matters Legal	213
51	Thanks	217
	Tool types and Buttons Index	219
	Command Index	221
	Concept Index	223

Since the out of net and zone routing makes use of what is termed *low priority net mail* it may not work in all FidoNets and other facilities may have to be developed later.

29.5.5 FidoTime

This program is designed to be called from a script file and return a level 5 AmigaDOS result (a warning level) if it is time for the FidoNet commands to run; otherwise, it returns a level zero. The AmigaDOS 'Warn' command can then be used to check to see if the current time was between the two times specified and the appropriate section of the script file can be executed.

The program takes two parameters, both in the 24 hour clock, the first is the hour when FidoNet mail starts, and the second is the hour when it ends.

So if your site accepts FidoNet calls between 05:00 and 07:00 (ie starting at 05:00 and ending at 06:59:59) then you provide a 5 and a 7. If your system starts processing at 23:00 and ends at 02:00:00 the next day you provide a 23 and a 2.

An example script file usage might be:

```
FidoTime 3 4
IF WARN
  RUN TrapDoor answer
  WAIT 5 secs
  TrapTell "rings 1"
  TrapTell "rings 1"
  WAIT until 04:00
  TrapTell quit
ENDIF
```

It can be used for other purposes besides FidoNet, I use it to schedule when it is a good time to run the SIGs aging utility 'trim'.

This command should probably be made to take minutes and seconds as well as the hours.

29.5.6 SafeWait

This is an alternative to the AmigaDOS 'wait' command. This command takes one parameter, a time in seconds past midnight that the command should wait until before returning. This can be useful in script files because it will return immediately if the current time is already past the time the command is supposed to wait for; whereas, if you were to use the AmigaDOS 'wait' command your script file would wait until the time arrived *the next day* before proceeding.

6.4	USR HST Modem	28
6.5	USR Dual Standard Modem.....	29
6.6	SupraFAXModem V32bis	29
6.7	V.32 and V.32bis Modems	30
6.8	V.34 Modems.....	31
6.9	V.42, V.42bis and MNP Modems.....	31
6.10	High Speed Protocol.....	32
7	Access Control.....	33
7.1	Access levels	33
7.2	Mixing access levels.....	34
7.3	Access levels in different groups	34
7.4	How many security groups?	35
7.5	DOS Shell access.....	35
7.6	Setting access levels.....	36
7.7	Privileges	36
7.8	Time Limits	37
7.9	Account Defaults	37
8	The IceBBS.ACCESS Files	39
9	The S:IceBBS.SHELLACCESS File.....	41
10	The S:IceBBS.MAXACCESS File	43
11	The S:IceBBS.MSGID File	45
12	The S:TRIM.LASTTRIM File	47
13	The S:IceBBS.SIGS File	49
14	The S:IceBBS.NEWSIGS File.....	51
14.1	New SIGs file format.....	51
14.2	The Email SIG's Name	52
14.3	SIGS Access Control.....	52
15	The S:IceBBS.BINIT File	55
15.1	Line 1	55
15.2	Line 2	56
15.3	Line 3	56
15.4	Line 4	56
15.5	Line 5	56

21.16	login.....	79
21.17	news.....	79
21.18	password.....	80
21.19	prompt.....	80
21.20	protocol.....	80
21.21	sig.....	81
21.22	term.....	82
21.23	top.....	83
21.24	upgrade.....	83
21.25	upload.....	83
21.26	users.....	83
21.27	who.....	83
21.28	xdown.....	83
21.29	xup.....	84
22	Error Message Files.....	85
23	Message of the Day.....	89
23.1	Customizing.....	89
23.2	Motdinfo.....	90
23.3	NewSince.....	90
23.4	IsNetUp.....	90
23.5	Append.....	91
24	Account.....	93
25	Edituser.....	95
25.1	Introduction.....	95
26	EU, the MUI Edituser.....	97
27	The Coupon System.....	99
28	Sharing Lines.....	101
28.1	Pause Line.....	101
28.2	Timed Script Files.....	101
28.3	Front door programs.....	103

are companies who will sell you access to the Internet⁵ In North America some public libraries are starting to provide access to the Internet, these services are usually provided under the name *FreeNet*.

30.2 Configuring for UUCP

Once you have a news feed and you are setting up UUCP⁶ there is one additional thing to note, you must⁷ run UUCP in a dial out mode if you are going to have it share one of your serial lines with IceBBS. Since you are running it in a dial out mode you do not need to use the 'Getty' program and your script file (which will run UUCP) will only need to call the 'Uucico' program. Nothing additional needs to be run in the background and any serial port (or device) may be used with 'Uucico'. The following⁸ is a portion of the script file⁹ that I use to obtain my USENET feed, this is run every two hours or so on my system through the IceBBS timed scripting feature on one of my lines.

30.3 UUCP Support Commands

The following subsections detail the various UUCP support commands that are provided with IceBBS.

30.3.1 NewsIn

'NewsIn' is a conversion utility designed to move USENET news that have been received by UUCP into IceBBS SIG link and data files.

'NewsIn', like the 'import' command, will automatically make copies of messages that are found addressed to users of the BBS. These copies are placed in the users' email SIG. As some users find this feature annoying the 'CC: Usenet' button in IceTERM is provided so they can turn it off.

⁵ Usually through PPP or SLIP connections, although some offer UUCP feeds too. The fees that ISPs charge have been dropping rapidly in the last few years.

⁶ It should be possible to use AmiTCP through a PPP (or SLIP) to an ISP who has an NNTP (Network News Transfer Protocol) server to obtain your USENET feed. This is a theory, let me know if it works.

⁷ Apparently WPL will accept incoming calls from both FidoNet and UUCP sources so this is not strictly true.

⁸ Remember that the '+' character is used to indicate that a command is too long to fit on one line and has been split.

⁹ An additional example is included in the distribution archive in the 'BBS_S' directory as the file: 'usenet_script'.

32.5	FindStat, trimming the database.....	128
32.6	FindBuild, building the database.....	128
32.7	The list of databases.....	128
32.7.1	Why use these databases?	129
32.7.2	How large are these databases?	129
32.8	What are the limitations?.....	129
32.8.1	Memory usage.....	130

33 File System 131

33.1	Format.....	131
33.2	Indirected Storage.....	133
33.3	File Tools.....	133
33.3.1	BBSJoin.....	133
33.3.2	BBSSplit.....	134
33.3.3	UJoin	134
33.3.4	AmiJoin.....	134
33.3.5	FMT.....	135
33.3.5.1	Processing modes.....	135
33.3.6	DelOld.....	137
33.3.7	Access.....	137
33.3.8	HashAccess.....	138
33.3.9	FileList.....	138
33.3.10	Store.....	138
33.4	Listing New Uploads.....	138
33.5	The Bulk Programs.....	139
33.5.1	BulkSplit.....	139
33.5.2	BulkJoin.....	140
33.5.3	BulkTest.....	140
33.5.4	Example.....	140

34 Message Bases..... 143

34.1	Old Style: Feedback	143
34.2	New Style: SIGs.....	145
34.2.1	The Storage and Retrieval Methodology ...	145
34.2.2	The FirstLinkItem structure.....	146
34.2.3	The LinkItem structure.....	147
34.2.4	The DataItem structure.....	149
34.2.5	The VMSG structure.....	149
34.2.6	Traversing links.....	153
34.2.6.1	Finding a message by number.....	153
34.2.6.2	Finding the next message.....	157
34.2.6.3	Finding the previous message.....	158
34.2.6.4	Finding the end of file.....	159
34.2.7	The Sysop's Message Reader.....	159

30.3.4 FwdEmail

This allows one to automatically forward email from the email SIG (and the old email system, and emailed files) to other accounts on other machines on the internet via UUCP.

30.3.5 Outmail

This program handles both the export of USENET (internet) email and FidoNet email from the email SIG. As such it can even cross route messages for you, for example: if you don't have a FidoNet feed yet you still want to be able to send messages to FidoNet sites via internet.

30.3.5.1 Info server

The 'outmail' command provides support for the information by email server (see Section 30.3.3 [InMail], page 115) with the '-mi' command switch. By using the '-s' mode of 'inmail' the contents of any messages that are sent requesting information will be placed into a SIG. Now if someone replies to one of these messages that reply will only be seen locally. But if 'outmail' is run on that SIG the reply will be forwarded to the person who requested the information in the first place.

30.3.5.2 Sharing a mailing list

The '-mp' command switch activates 'outmail's mailing list sharing mode. If you have several users on your BBS subscribing to the same high traffic mailing list you should consider using this¹⁰ as it can cut down the size of the email SIG. What you do is to add another SIG to the BBS which will be used to store messages from the mailing list (for example *lightwave_sig*). You then subscribe to the mailing list on behalf of the BBS using a convenient user name that is not in use on the BBS, say *common_account*. You then run 'inmail' to scan the 'UUMAIL:' directory looking for messages addressed to name you subscribed to the mailing list under (ie. *common_account*). You then have 'inmail' place these into the common SIG, thus:

```
inmail -s common_account lightwave_sig
delete uumail:common_account
```

This places the message received from the mailing list into a single SIG that your users can subscribe to. When a user makes a posting into this SIG, or when he replies to a message in the SIG you want those

¹⁰ Note: at the time of writing this has not been heavily tested so take care.

41.10.1	What is IRC?	184
41.10.2	Installation for IceBBS sysops	184

42	IceTERM icon problems	185
43	Format of Account Files	187
44	CDROM Drives	189
44.1	Using deferred assignment	189
44.2	CDTV	190
44.3	CD32 and AmigaDOS 3.1	191
44.4	CDROM-FS	191
44.5	Xetec	191
44.6	AsimCDFS	192
44.7	AmiCDROM	192
44.8	CDROM Sources	192
44.8.1	Walnut Creek	192
44.8.2	Fred Fish	193
44.8.3	AMUC	193
45	Removable Hard Disks	195
46	Using ChangeTaskPri	197
47	ANSI Codes	199
47.1	Escape sequences	199
47.2	Amiga ANSI Codes	200
47.2.1	Cursor Positioning or Movement	200
47.2.2	Erasing the Screen and Erasing a Line	200
47.2.3	ANSI Colour Graphics	200
47.2.4	Additional ANSI Attributes	202
48	Future Enhancements	205
49	Glossary	207
50	Matters Legal	213
50.1	Direct Updates	213
50.2	Distribution Restrictions	213
50.3	Support	213
50.4	No Warranty	214
50.5	Disclaimer	215

There is now a rapidly growing industry called *Internet Service Providers* (ISPs) whose business is to sell connections to the Internet. So finding a point of access is not difficult (at least in major cities in North America). However such access is not inexpensive, but the costs are dropping very rapidly. In North America I expect the cost of a 14400 or 28800 baud full time connection to the Internet to eventually drop to about \$70-100 per month, unless the phone companies start providing permanent connections, in which case² the prices could drop still further. However, I expect the phone companies will attempt to make fat profits for a while at least.

An ISP can provide you with varying levels of connection, each of these would be at a different cost. The most common is a direct modem connection using PPP or SLIP as the protocol. These are available at different speeds, as permanent or dial up on demand, and may be billed by the month, day or hour. At a higher price are ISDN and special dedicated line connections. These can take you to 64K or even 128K baud rates. Beyond this are special leased lines which are what the ISPs use to get their feeds. The cost of these is pretty high (perhaps \$1000.00 per month and up). Names like T1 and T3 are types of leased lines.

If you happened to live beside an ISP you could even just sling an ethernet cable over the fence.

If you live within about a half mile of an ISP you might be able to bypass the phone company altogether by using a pair of radio modems. One such pair was written up in the November 1994 issue of Boardwatch magazine (pp. 26-35). This modem is manufactured by:

Metricom
980 University Ave
Los Gatos, CA 95030.

Voice: 408-399-8200
FAX: 408-354-1024
Email: info@metricom.com

These *Ricochet* wireless modems use the FCC Part 15 spread spectrum radio band (902-928MHz) and with less than 1 watt of output they do not require licensing in the USA. They can be pretty fast, Boardwatch reported about 77000 baud, but a pair of them will cost about US\$1000.00, which should pay for itself in under two years through phone savings. And since they are replacing a pair of high speed modems (say about

² Since the connection would only need one phone line from you to the switchboard, rather than three: from you to the switchboard and then from the switchboard to to the ISP and then from the ISP back to the switchboard.