

FROBOT

File Robot Facility

Version 2.0

By Craig Derouen  
1:343/22

April 29, 1991



## Contents

---

Requirements .....	3
Features .....	4
What is File Robot? .....	5
How does File Robot make the requests? .....	6
Installation .....	7
Managing the FROBOT control file .....	8
Managing the FROBOT text instruction file .....	10
Frget.exe, script edit utility.....	14
Ready, Set, Go! .....	15
Warranty .....	16
History .....	17
Dedication .....	18



## Requirements

---

Operating System ....: DOS 3+  
Video .....: Generic MSDOS  
Keyboard .....: Generic MSDOS  
Memory .....: Less than 150k  
Net Mail .....: Fido format message system  
Mailer type .....: Fidonet standards such as Fido 12, Binkley  
Opus, Dutchie, Seadog.



## Features

---

- o Time schedule control of File Requests
- o Time schedule control of File Sends
- o Complete control and automation of File Request retries
- o User specified Fidonet attributes for File Requests and File Sends
- o Accurate reporting of File Request and File Send activity
- o Acknowledgment receipt of File Requests
- o Courtesy netmail to destination of File Requests
- o Courtesy netmail to destination of File Sends



## What is File Robot?

---

File Robot is a Fidonet utility I designed to ease the sysop management of file requests and file sends. It allows you to 'schedule' file requests and sends so they are only processed by your Fidonet Mailer after a certain time & date. File Robot will also test to see if a file request has been received after shipping out request. If file(s) were not received, it can automatically retry the requests on sysop specified maximum number of times, or indefinitely.

How many times have you tried to place a file request to some bbs and only told to `FREQ` during some certain time, often late at night or after Zone Mail hour? Did you actually stay up until that time then manually send out the request? I got tired of doing that! Also I wanted to know if my file request had been received, so thus Frobot was born. There are certain limitations as to how much detection Frobot can do for managing file requests. I have tried to develop the best possible logic for this management within the constraints of current Fidonet technology, in a generic fashion supported by most if not all mailers.

Frobot will notify you in 3 different ways when a file request has been received:

- 1) Optional logging of FROBOT actions to a file. This can be your normal `OPUS`, `BINKLEY`, `QBBS` or whatever log. Or alternately it can be a special separate log file of your choosing. You specify where and if a log file is to be maintained. If the log file does not exist, it will be created for you automatically, of course.
- 2) Optional message reply to you, the `SYSOP`, in your netmail directory. I find that most sysops will read (or should) their netmail directory before any other area. Frobot will create a standard FIDONET style message giving time, date, when and which files were received. Quite handy!
- 3) Frobot automatically tracks the script instruction file, where you specify which files are to be requested/sent. When a file has arrived that was requested, it will `DISABLE` that instruction line and put received date information on that line. This methodology forces FROBOT not to make `Freqs`'s after a file has been received.



## How does FROBOT make the file request?

---

Your mailer still does most of the work. File Robot automatically manages your Fidonet outbound directory by creating the necessary REQ (File request) files and FLO files. These 2 file formats are Fidonet standard and honored by all Fidonet mailers. Frobot will generate a FLO file for BOTH file requests and file sends, as recognized by Binkley and others.

Frobot is intelligent enough to scan outbound directory and know when REQ files need to be changed, edited, added to generate a file request. File sends require management of FLO files also, which Frobot handles quite nicely, making sure no file dupes occur. You specify the Fidonet attribute to apply to each file request/send. FLO files are created/adjusted accordingly.



OK, I'm excited!  
Now how do I install this and get started?

---

Copy Frobot.exe to any directory along your path. Copy Frobot.ct1 either to the directory you start Frobot from, the home directory of Frobot or your Binkley start directory, or in the worst case you may specify an ALTERNATE control file on the command line when you start Frobot. This could be used to manage multiple configurations such as zones (where you have multiple outbound dirs, etc). Frobot.exe will default search for the file "Frobot.ct1" in the following order:

1. A filename/path you specify on Frobot.exe command line
2. Directory where you execute Frobot from.
3. Directory where Frobot.exe lives (Likely in your path statement).
4. Looks for the "BINKLEY" environment statement and goes there looking for it.

Now that you have that straight, edit FROBOT.CTL with your proper bbs configuration parameters. This is a HEAVILY commented file which should tell you everything you need to place in it. We describe all the tokens again in this document, but you could just skip that and go straight to editing.

Frobot needs to know where your outbound, inbound, netmail directories are. Also other special options are specified in this file. If Frobot can not find this ctl file, it will complain and abort to DOS with an errorlevel of "1", which you can test for in a batch file. All FROBOT ERRORS will exit with a code of "1". Errors will be logged to the console as well as the log file if possible. Here's the basic steps to setting up Frobot and getting things going:

- 1) Edit Frobot.Ctl.
- 2) Edit Frobot.scr (script instruction file), it should be located where specified in Frobot.Ctl, or optionally run the lazy FrGet.exe program (see details).
- 3) Execute Frobot several times a day during scheduled events.

Really that's all there is to it! I'm a big beleiver in simplicitcy.



## Managing the FROBOT control file

---

As said above, FROBOT gets its various information for operation from a control file. Some of this information can be found thru other various BBS files such as MAIL.SYS or CONFIG.DOG, but not all of it. Plus it makes for a lot of additional coding to find and parse those files! Mail.Sys is useful but tells you very little and Config.DOG tells more, but many folks don't have one or know how to create such, so I had to establish my own control file. It also allows me add additional options without having to put them on the command line. I really dislike multitude of command line switches, remember Fido 11?!

Frobot.ctl is basically a text record file. Each record consists of one text line bounded by a carriage return/linefeed. Tokens within that line are delimited by any combination of spaces and tabs, only position is dependant. Any records beginning with a ';' is considered a comment line and ignored by FROBOT, thus easy to add comment lines. Also comments may be added to end of record with a ';' as shown in example.

First token item is parameter type. Case is not significant. Each parameter is described in detail below:

- (NODE):                   Your current Fidonet node number. No Zone support yet so place only net/Node. You should use your primary Node number, though Frobot doesn't really care. It uses this number when sending out mail messages to callee.
- (OUTBOUND):               Directory of outbound files, mail. Should have full drive and path qualifier. You may or may not add a trailing '\' to directory. Frobot will append one if necessary.
- (INBOUND):                Directory of inbound net files. Frobot will scan this directory for any incoming files that you requested.
- (NETMAIL):                Directory of where standard Fidonet netmail messages are placed. Frobot will send netmail to yourself and place optional netmail to callee here.
- (SCRIPTLOC):              Fully qualified filename of Frobot script instruction file used for processing. See following chapter for full discussion of format of this file.
- (LOGFILE):                If given, fully qualified filename of log file where to place any errors or incoming files tracked. If logfile exists, Frobot will append to it standard OPUS/BINKLEY log format information, else it will create the file on first invocation.
- (DISPLAY):                Logical option to display activity on local console. Doesn't really make much difference but most folks like to see what its doing, so Frobot will tell you how each line is processed as it scans, if enabled here.



- (SELFMAIL) : Logical option to send yourself netmail when a filerequest has arrived. Netmail message will be marked as local, from Frobot, addressed to SYSOP, so it is easily distinguishable.
- (SYSOPMAIL) : Logical option to send courtesy mail to calling bbs. When file requests are send out, Frobot will generate a netmail message also (in netmail dir) telling callee what files you are requesting. Highly reccommended. Sometimes when we request files, the name is slightly wrong or such, if we send out a message, sysop can then easily see what was wrong and correct it. Quite useful! Also makes sysops feel less uneasy about file raping. Wwhen sending files out a message is generated to destination, Frobot will exit with an errorlevel of '2' when this option is turned on and netmail was created, allowing you to run your packer after Frobot, in batch file.
- (BELLS) : Logical option to turn on fancy bell sounds if an error occurs. If your run Frobot in the middle of the night, this may wake you up!

That's it! I may add additional options in the future and using a control file format such as this makes it easier. BTW, the actual options are not dependant upon any specific line number in file, i.e. in the current example in the archive options are listed in above order, but this is not really required if for some crazy reason you want to move them aound.

When Frobot starts up it will process this control file and display the various options and their respective tokens on the local console, using standard DOS redirectable i/o. You can use this for any possible debugging assistance. If there is an error in this file (missing token or non-existant directory), Frobot will immediately abort to DOS with errorlevel of '1' and NOT process the text instruction file.



## Managing the FROBOT script instruction file

---

Now take your Frobot.txt file and place it in the directory where you plan to invoke Frobot. This will usually be your home bbs directory such as "C:\Opus" or such, but of course it can be anywhere. The exact location of this file must be specified in the Frobot.Ctl if you wish to use a different name. As an alternative you can use the added utility Frget.exe which can be invoked from batch files or prompted. See separate section for details.

This is the file you will be working with daily to control Frobot. Both you and Frobot will be maintaining this file. Edit the file with any standard ascii text editor such as Pc Write or Qedit. The file as it is shipped is heavily commented to help guide you thru it, but I will clarify it all here in the next section. Frobot maintains this file by creating a temporary file with the same filename and an extension of ".\$\$\$" in the directory where it found the file. As it processes the file, it copies each 'record' to that temp file and if it needs to modify the record it will do so before copying out the record. All records are copied regardless of whether they are comments or not. When Frobot exits, it deletes the old file and renames the temp file to the old extension.

Thus, you have an exact copy of the old file but with possible changes in it. This file can be browsed easily to see what changes have occurred and how many times Frobot processed a request if needed to. Quite useful. Old records processed and completed are commented out and left intact otherwise, thus allowing only you to manually remove the records with your text editor.

The FROBOT script instruction file, default name of Frobot.scr, is the most critical part of Frobot. This file, standard ASCII, is maintained both by you and by Frobot. You will be editing and deleting records in this file. Frobot will only be editing and 'disabling' records, it will NEVER delete any records. A record consists of a free form text line ended by a carriage return, line feed combination. The ';' semicolon is used to delimit the record. Any text placed after a ';' is considered a comment and will not normally be touched by Frobot. Thus, you can place any kind of information suitable to you in keeping track of file requests. Naturally a line that begins with a ';' is considered a full comment record by Frobot and is totally ignored. I suggest at this time you print out the default FROBOT.SCR that came with this archive while you read the rest of this of this section.

A standard FROBOT script record consists of the following information:

```
OPTION  FILENAME  NODE   SDATE   STIME   ATTR   MAX
R FILES.ARC   343/1 12/01/88 13:45 C 50
```



All tokens in the text record are separated by any number of spaces/tabs. The first 6 token items are NOT optional and must be present in the record. Thus a record may consist of line length of up to 128 characters. I had to come up with some kind of reasonable limit. I will describe each token as they are looked at by Frobot here. Tokens are not case sensitive. Tokens are position dependant though! 1st token is considered to be the OPTION, 2nd the FILENAME, etc.

(OPTION): Currently only 2 options, 'R' for File requests and 'S' for file sends. Not case sensitive. Any 1st key other key in this line will generate an error and Frobot will disable line so it won't be processed again. Frobot will however tell you why it disabled the record. A record is disabled by placing a ';' in front of the OPTION field. This happens automatically. The 'S' token for file sends can have 1 of 2 additional options for specialized things. You can append a 'D' for deleting the file(s) after sending. Or append a 'T' for truncating the file to 0 size after sending.

(FILENAME): This is any legal DOS filename. For file requests this can be a MAGIC filename such as ABOUT or FILES, otherwise it may only contain 8 legal DOS filename characters, a '.', plus 3 chars for extension. Frobot will automatically detect if it is a MAGIC filename by seeing it has no extension. For File Sends, this filename must be the full qualifying pathname of the file, otherwise it will assume the file is from current directory, which may not exist there. If sending a file request, you may be required to supply a password, this processed in FROBOT by tacking a "!" and the password name after the filename (No space in between), such as:

"Files.arc!Password"

Passwords are only used in file requests.

Instead of a single filename you may use a file include list. Just prepend the filelist name with a '@'. Frobot will recognize this and parse the list for internal filenames for sending or receiving. Each name in the list is handled individually. You may use standard filenames, magic filenames, etc. Lines prepended with a ';' in listfile are ignored. Blank lines are stripped. For file requests, append a count number after each filename, (sep by space) if you want individual counts for the filerequests. Elsewise it will use the default count passed on script line. Frobot will not disable the script line until ALL files in the list are sent/requested, received or disabled for whatever reason. See example file list included for details. This feature can give you a lot of flexibility. If you send/request to multiple nodes you'll still have to create separate script lines and filelists, as each one is tracked individually by Frobot.

(NODE): Standard Fidonet net/node. Sorry, but Zones are NOT yet supported! It is assumed it will be in same zone. Zones will be supported in future release. This is the address of where you wish to request the file(s) or send file(s) to.



(SDATE): Starting date for file request/send. Standard US format with MM/DD/YY. Optionally you may specify NOW instead which says today's date (at invocation of Frobot), for immediate processing, else Frobot will not process line until system date reaches this starting date. There are 7 other special macros for specifying certain days. They are:

SUN, MON, TUES, WED, THU, FRI, SAT for obvious days of the week This means that if today matched one of those macros, then the sdate is ok for processing. Should be obvious. Only first 3 characters of day macro is significant.

(STIME): Starting time for file request/send. Standard 24 hour for your LOCAL area. If you specify "NOW" instead, it will process at ANY invocation on passed date, regardless of time of day. Elsewise, Frobot will not process record until current time is > or equal to start time.

(ATTR): Standard Fidonet Attribute. 'C' for crash request/send, which means mailer will always try, 'D' for normal direct request/send which says try only during ZMH. (Note that under Version 6 of the Nodelist, Binkley will send out normalized mail to a node regardless of non crash-attribute if the node supports Continuous mail). 'H' for hold sends/requests. This is would be used when you want callee to poll and pickup on his nickel rather than yours.

(MAX): Maximum # of times to try requests. Frobot will keep trying file requests until the specified files are in your inbound Net directory or until the MAX count is reached. Frobot will keep track of this count by appending a '=x' to MAX value. X will be the number of times it has currently tried to send out the request. You can later look into this file and see easily how many times Frobot has polled the callee. Handle this with care! A large number can run up a good fone bill. If you specify a MAX of '0' then Frobot will ALWAYS try to send out request UNTIL it sees the file in the inbound dir. There are some exceptions to this:

Magic filename receipts can not be detected. They will only be processed once by Frobot. After the request is processed 1 time, FROBOT will disable by commenting record, and tell you so in the comment field. Remember that MAGIC filenames are determined by a filename with no extension. I know of no other portable way to detect this. Any suggestions?

File sends are only processed 1 time. Of course there is no real portable way to tell if the file(s) were truly sent so we have to assume 1 time. Thus, for file sends (OPTION=S) the MAX token field is optional and will be ignored. Of course the starting date and time are still honored. When the record is processed, the record will be disabled and marked with a comment. This saves wear and tear on your fone bill!



As you can see, the script file is the heart of the operation! All you really need is this to muck with to operate Frobot once things are setup. When records are disabled, you might periodically delete them from the file if desired. You will find however that Frobot is quite fast at processing this file so size of file is not too important. Admittedly though I haven't tested Frobot with a script file bigger than 64K, so it probably shouldn't get larger than that.



## FRGET.EXE

### The lazy sysop's Frobot utility

If you don't care to spend all your time editing the Frobot script file then you can use Frget.exe, an add on utility, instead. Place Frget.exe somewhere on your DOS path, where Frobot.exe is usually. Frget is a special utility that will automatically APPEND file requests/sends to the script file, based upon prompts from the sysop or command line. It makes life easy for us lazy sysops!

Frget.exe will look for Frobot.ctl under the same conditions as Frobot.exe, except you can not specify an alternate ctl file on the command line. Elsewise it will look for teh ctl file in same directory, then in directory where Frget.exe exists, Binkley environment string, etc. Ot only needs a couple of tokens, the script file name and bells option.

On the command line for Frget you can specify the complete script function, identical in format to the script file:

```
"Frget [Option] [Filename] [Node] [Sdate] [Stime] [Attr] [Maxtimes]
```

[Maxtimes] is optional and a "1" will be the default if not given. If you do not specify the full number of options (6) then Frget will automatically switch into the friendly prompt mode asking you for input for each parameter with a little bit of explanation. Its very friendly, try it sometime. Prompt mode is also invoked if NO parameters are specified, of course. While in the prompt mode, you can press control-c or control-break to abort, the script file is not altered. The prompt mode will not allow you to enter blank parameters of course.

Frget will do a fair of amount of parameter checking to see if things look ok and then, if things are ok, appends a new script line to the file. That's all there is to it. Of course it's designed to be run from a batch file using command lines as well as interactive mode. This utility will likely be your preffered method for mucking Frobot commands.



## Ready, Set, Go!

---

Now that you have the Frobot control and script file under control you are ready for operation. All that's left is to set up schedules so that Frobot is automatically executed occasionally. This should be done at least once daily and obviously more often if you want to get finer control over time schedules. I run mine whenever I do an echomail export, or scan, and also at beginning of events such as ZMH or ld fone calling charges change.

If you have Frobot setup to send courtesy mail to the callee (HIGHLY RECOMENDED!) then you will need to run a message packer after Frobot is run to pack up messages to the outbound directory. Frobot will determine if it needs to have the external packer called by exiting with an errorlevel of '2' . If other errors it will exit with code of '1', elsewise exit code of '0'. I always run Frobot in my EXPORT function just prior to running Qmail anyways so the exit code detection is not really needed in that situation. Frobot will also display a message to the console if the packer needs to be run. In any case, this is not real important as eventually the packer probably will be run anyways and the message(s) will go out.

Frobot does not require any command line options for normal operation. You could specify a filename for alternate control file perhaps to specify an different script file or alternate outbound directory for other zone, etc. Here's an example of my batch file to run Frobot.

```
EXPORT.BAT:
@echo off
timelog Exporting Mail
frobot
if errorlevel 2 qm pack
```

Again, this is run several times a day usually during schedule change or more idle times, when I want be sure echomail gets exported out in a timely manner. The more often you run Frobot, more fine tuning you have over time of day control, obviously.



Warranty  
Disclaimer

---

You must be joking! No? Well er,uh, ok, here it comes.....

Due to the wide range of possible target environments, and the free cost nature of this product, the author of this software disclaims all responsibility for any and all damages or liabilities resulting from its possession or use, including larger, costlier telephone bills!

If you send me netmail discussing problems or looking for solutions, odds are I won't answer. I simply can not afford it. It's possible I may answer, but don't count on it. I am welcome to netmail suggestions. But at this point I don't know if I will release any more versions or not.

If you want to see source for Frobot, you should be able to obtain it from the same location where you got this archive. Frobot was submitted in the SDS network system under the names:

```
FRBT_20.LZH : This archive
FRBTS_20.LZH : C Source for Frobot and utils
```

Naturally I won't support the source at all! Frobot was written in Microsoft C 6.01, but probably can be easily mucked to compile with Borland C/TurboC or others. Most of the routines are very generic. I welcome others to use the source code for studying and hope it will encourage others to generate fine Fidonet utils, source code released too! If you are an MS-DOS programming working with C or 8086 give my bbs a call, Cornerstone. (206) 362-3213, FIdonet 1:343/22. It is run by Trio Software and we have a HUGE collection of C,8086 source code online, in fact we are dedicated to that subject, exclusively, with 100 Mbytes online!



## History

---

Frobot Version 1.0 was originally written and submitted to SDS in December of 1988. It had some problems with it and I released an update 1.5 a couple of weeks later, also some new new enhancements and some add-on utilities fget.exe and fsend.exe. I have been using version 1.5 here religiously since.

Somewhere along the line unfortunately I lost the source to it, and decided I wanted to do some enhancements. So re-inventing the wheel, I sat down and wrote Frobot 2.0 from scratch. With a few more enhancements, and probably better code, I am now releasing this again to SDS, this time with full source so others may make use of the routines. It is not likely that I will be releasing any more major versions, Frobot does what it was designed to to do, unless some brilliant idea comes along. It's possible that I may add the "file update" feature to Frobot though I have no need for that here. If you decide you want to tackle that, welcome to it. I'd like to see any source code enhancements you make, as Frobot version should always be released by the original author, per copyright. To this day I have still not seen any other Fidonet mailer utility that does these exact functions.



## Dedication

---

This product is dedicated to the following individuals who have done much to bring Fidonet to its current advanced state:

- Tom Jennings: Grandfather of Fidonet, my original 'Net' coordinator back when I got started in Fidonet in 1984.
- Thom Henderson: Father of ARC, SEADOG, ARCMail, and other outstanding utilities that have vastly improved Fidonet standards!
- Wynn Wagner: Father of OPUS, which greatly popularized Fidonet with his outstanding and totally PUBLIC DOMAIN bbs system.
- Bob Hartman: An outstanding, tireless individual who has generously donated so many fine utilities such as Confmail, Binkley, Ommm, Parselist etc. How do you do it Bob?!
- Vince Perrillo: Another tireless programmer involved BOTH in the Binkley development trio as well as OPUS development.
- George Stanislov: Developer of AVATAR, bbs language system, and one of the initial developers of OPUS 1.1X.
- Tom Kashuba: Outstanding developer of high quality OPUS utils such as OOPS and ROBO, which inspired me to develop this program.
- Scott Dudley: Author of Maximus CBBCS, an outstanding bbs software program, which I am proud to be running. Scott has carried on the fine dedication of fidonet by not only providing a solid working bbs free of charge to all users, in the OPUS/BINKLEY tradition, but also gives away entire source code for it!

-----

NET343: My local net, whose put up with my doings for years now... Really, all Fidonet sysops deserve an applause for the thankless tasks they perform!

The list could go on and on! I have used so many fine utils for Fidonet that it would fill up too many pages. I hope that some others are inspired by this co-operative effort and continue to strive in making Fidonet an evolving humanistic technology.

Sincerely,

Craig Derouen  
Fidonet 1:343/22  
April 26, 1991

=

