

????

COLLABORATORS

	<i>TITLE :</i> ????	
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>
WRITTEN BY		July 8, 2022

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	???	1
1.1	Preliminary	1
1.2	Introduction	1
1.3	Distribution	2
1.4	Arguments	2
1.5	How does it work ?	4
1.6	Program control	4
1.7	Author	5
1.8	Program history	5

Chapter 1

????

1.1 Preliminary

=====

NoDelete 1.10

Copyright (C) 1999 by Denis GOUNELLE

=====

Introduction

Distribution

Arguments

How does it work ?

Program control

Author

Program history

1.2 Introduction

NoDelete purpose is to offer a global trashcan to the system. ←

This means

that when a file is asked to be deleted, it is moved to a special directory rather than being deleted, so giving you a chance to get back the data if you discover later that you need it.

Because NoDelete replaces the "delete" function of AmigaDOS, it will work for ANY delete request, from ANY program you use. The program is automatically declared as a Commodity, and also has an AREXX port.

NoDelete is fully configurable (see the Arguments section) and installation is very easy: you just have to copy the program in the WBStartup directory of your boot partition, and to check the program arguments. Then NoDelete will be automatically started next time you will boot your Amiga.

1.3 Distribution

This program is made available to you as a freeware program. Without my written authorization, any commercial usage or selling is forbidden. You can copy and spread this program under the following conditions:

1. All the files must be provided
2. No file has been modified, in any way
3. You can't ask for more than \$5 for this

In spite of several tests, no warranty is made that there are no errors in NoDelete. YOU USE THIS PROGRAM AT YOUR OWN RISK. In no event I shall be liable for any damage, direct or indirect, resulting of the use of NoDelete.

Because AmigaOS does not offer a function able to patch a dos.library vector, NoDelete has to do this by it's own. Although this is done in the cleanest possible way, some future version of the system may forbid such manipulation. Note that this program is compatible with AmigaOS release 3.5

1.4 Arguments

The list of possible NoDelete arguments is shown below. For each argument, the first line shows the Shell syntax, while the second one shows the Tool Types syntax.

CR=CONFIRMBYREQ/S
CONFIRMBYREQ

If this argument is specified, you will be asked what to do when a file is larger than the CONFIRMSIZE limit: either to delete the file, or to move it to the trashcan anyway. If this argument is not specified, the file will be deleted without warning (default behaviour).

CS=CONFIRMSIZE/K size
CONFIRMSIZE=size

Specifies the maximal size (in bytes) of files to move to the trashcan: files below this limit will be moved, files above will be deleted (with or without confirmation, see CONFIRMBYREQ above). Default value is 131072 bytes (128 KBytes).

CXK=CX_HOTKEY/K hotkey
CX_HOTKEY=hotkey

Specifies the Commodity's hot key. Default value is "ctrl lalt del". This argument is quite useless for this version.

CXP=CX_PRIORITY/K/N priority

CX_PRIORITY=priority

Specifies the Commodity's priority. Default value is 0. This argument is quite useless for this version.

LF=LOGFILE/K filename

LOGFILE=filename

Specifies the name (with path) of the optional NoDelete's logfile. If the argument is specified, the file will be used to log details about the program options and results of the trashcan checking.

MD=MAXDAYS/K/N days

MAXDAYS=days

Specifies the maximal age (in days) of files in the trashcan. Older files will be definitively deleted each time the trashcan is checked. Default value is 30 days.

MS=MAXSIZE/K/N size

MAXSIZE=size

Specifies the maximal size (CAUTION: in mega-bytes !) of each trashcan directory. Each time the trashcan is checked, files in the directory will be definitively deleted, from the older to the newer, to lower the directory size to this limit. Default value is 1 MByte.

REXEXPORT/K name

REXEXPORT=name

Specifies the name of the AREXX port to open. Default value is "NoDelete".

TD=THRASHDIR/K directory

THRASHDIR=directory

Specifies the name of the trashcan directory, where deleted files will be stored. Default value is "SuperThrashcan". Take care that the trashcan directory is considered to belong to NoDelete: never copy a file to this directory by yourself, because it might be deleted without warning. You may however delete one or several files in a trashcan directory, if you want to. But note that these files will be definitively lost.

You can specify either a basic directory name (like "MyThrashcan") or a full qualified name (like "Tmp>LastChance"). In the first case, one trashcan directory will be created on every writable DOS partition. In the second case, there will only be a single trashcan directory for all your disk partitions.

Take care that the single trashcan mode can have a dramatic effect on performances, because each deleted file will have to be copied from the original partition to the one where the trashcan is located.

XP=XPATTERNS/K patterns

XPATTERNS=patterns

Specifies a list of patterns of filenames that will never be moved to the trashcan, but immediatly deleted. This is useful, for instance, to prevent temporary files to fill up the trashcan. It is also possible to exclude a complete directory or partition. Default value is an empty pattern (no exclusion).

You can specify as many patterns as you like, using a comma as a separator. For example, an argument value of "#?.o,Tmp:#?" specifies

that files which name end by ".o" or which are located on the "Tmp:" partition must not be moved into the trashcan.

If you specify a volume name, take care to use the "logical" volume name as it appears on the Workbench screen and not the "physical" name (unit name shown by the "Info" command). For instance, write "Ram Disk:" instead of "ram:", or "Work:" instead of "HD1:".

1.5 How does it work ?

Each time NoDelete is started, the trashcan directories are ←
checked: files

older than the MAXDAYS parameters are deleted. Then, if any trashcan directories is larger than the MAXSIZE parameter, files are deleted (from the older to the newer) until the size of the directory is below the limit.

Once the trashcan checking is finished, the Delete() vector of the dos.library is patched so that not only the Workbench, but any program you use, will call NoDelete to remove files.

A file will be immediatly deleted if:

- NoDelete has been disabled (see the
Program control
section)
- it is already located in a trashcan directory
- it's name matches one of the patterns specified by the XPATTERNS parameter
- it's size is above the CONFIRMSIZE parameter (and the user as agree to delete the file if the CONFIRMBYREQ parameter has been specified)

Otherwise, the file will be moved to the trashcan, from where you will be able to recoved it later if you need. Files will be stored with their original names (so that you can find them easily) and with a comment giving their original location.

1.6 Program control

Because NoDelete automatically declares itself as a Commodity, you can control the program using any commodity handler like "Exchange" and:

- terminate the program ("Remove" button of the Exchange utility, this is equivalent as sending a CTRL-C break to the program)
- temporarily disable the program ("Disable" toggle of the Exchange utility)
- re-activate the program ("Enable" toggle of the Exchange utility)

CAUTION : as soon as you terminate or disable NoDelete, files will be actually deleted and not moved to the trashcan anymore.

NoDelete also automatically opens an AREXX port named "NoDelete" (this name can be changed using the REXXPOR argument) that accepts the following commands:

QUIT

Terminate the program.

DISABLE

Temporarily disable the program.

ENABLE

Re-activate the program.

GETSTATE

Return 0 if the program is currently enabled, 5 if it has been disabled.

DELETE files...

Delete the specified file or files (CAUTION: the files are really deleted, not moved to the trashcan !). You can specify as many files as you want. Patterns (like "ram:fil#?") are not allowed. You must provide fully qualified filenames (for example "ram:mydir/tmp/foo" instead of just "foo") and use quotes or double-quotes around names containing spaces (for example "ram disk:file"). The return code will be 0 if all the files were successfully deleted, or 5 if at least one file couldn't be deleted.

If an unknow command is received, or if some error occurred (not enough memory or the like), you will get a return code equal to 10 or 20.

1.7 Author

NoDelete was written in C++ by:

Denis GOUNELLE
27, rue Jules Guesde
45400 FLEURY-LES-AUBRAIS
FRANCE

Email: denis.gounelle@wanadoo.fr

Homepage: <http://perso.wanadoo.fr/denis.gounelle>

Don't hesitate to send me your comments or suggestions.

1.8 Program history

v0.00, 21-Aug-99, 31568 bytes

- First released beta version

v1.00, 28-Nov-99, 39356 bytes

- Is now a Commodity and has an AREXX port

- Renamed from SuperDelete to NoDelete

- Bug fixed: couldn't handle a space character in a tool types value (XPATTERNS for example)

- Bug fixed: didn't always set the DOS error code properly, so the caller knew that deletion failed but couldn't tell why

v1.10, 04-Jan-00, 39624 bytes

- Creates the files in the trashcan with the same protection bit as the original files
 - Removes a partially copied file in the trashcan, when the copy fails
 - The length of XPATTERNS, LOGFILE and THRASHDIR arguments is now unlimited from NoDelete's point of view (the Shell or the Workbench may have some limit here)
 - Added a few precisions in the documentation
-