

082a7b10-0

Alessandro Pedretti

| |
|----------------------|
| COLLABORATORS |
|----------------------|

| | | | |
|---------------|------------------------------|--------------|------------------|
| | <i>TITLE :</i> 082a7b10-0 | | |
| <i>ACTION</i> | <i>NAME</i> | <i>DATE</i> | <i>SIGNATURE</i> |
| WRITTEN BY | Alessandro Pedretti | May 28, 2025 | |

| |
|-------------------------|
| REVISION HISTORY |
|-------------------------|

| NUMBER | DATE | DESCRIPTION | NAME |
|--------|------|-------------|------|
| | | | |

Contents

| | | |
|----------|--|----------|
| 1 | 082a7b10-0 | 1 |
| 1.1 | xFX - Guide | 1 |
| 1.2 | 1. Introduction | 2 |
| 1.3 | 2. System requirements | 2 |
| 1.4 | IEC cable | 3 |
| 1.5 | 3. Installation | 4 |
| 1.6 | 4. Use of xFX | 4 |
| 1.7 | 4.1 Shell use | 5 |
| 1.8 | 4.1.1 Examples | 8 |
| 1.9 | 4.1.2 AmigaDOS Script | 8 |
| 1.10 | Workbench use | 9 |
| 1.11 | 4.2.1 Tool Types | 9 |
| 1.12 | 4.2.2 Drag & Drop | 10 |
| 1.13 | 4.3 Utilizzo tramite porta ARexx | 10 |
| 1.14 | 4.4 Hints | 11 |
| 1.15 | 4.5 Error messages | 11 |
| 1.16 | 6. Note | 16 |
| 1.17 | 7. Bugs | 18 |
| 1.18 | 8. Futuro | 18 |
| 1.19 | 9. Development tools | 18 |
| 1.20 | 10. Thanks | 19 |
| 1.21 | 11. Copyright | 19 |
| 1.22 | 12. History | 20 |
| 1.23 | 5. GuiX | 21 |
| 1.24 | 5.1 GuiX: Introduction | 22 |
| 1.25 | 5.2 GuiX: Menu | 22 |
| 1.26 | 5.3 GuiX: The Main Window | 23 |
| 1.27 | 5.3.1 GuiX: Read | 24 |
| 1.28 | 5.3.2 GuiX: Write | 25 |
| 1.29 | 5.3.3 GuiX: Copy | 26 |

| | | |
|------|---|----|
| 1.30 | 5.3.4 GuiX: Check | 27 |
| 1.31 | 5.4 GuiX: Device requester | 28 |
| 1.32 | 5.4.1 GuiX: Mounted | 30 |
| 1.33 | 5.4.2 GuiX: Unmounted | 30 |
| 1.34 | 5.4.3 GuiX: Device | 31 |
| 1.35 | GuiX Menu: Start | 32 |
| 1.36 | GuiX Menu: Delete Files | 32 |
| 1.37 | GuiX Menu: Save Prefs | 32 |
| 1.38 | GuiX Menu: About MUI | 33 |
| 1.39 | GuiX Menu: About GuiX | 33 |
| 1.40 | GuiX Menu: Iconify | 34 |
| 1.41 | GuiX Menu: Quit | 34 |
| 1.42 | GuiX Menu: Read | 34 |
| 1.43 | GuiX Menu: Write | 34 |
| 1.44 | GuiX Menu: Copy | 34 |
| 1.45 | GuiX Menu: Check | 34 |
| 1.46 | GuiX Menu: MUI Prefs | 35 |
| 1.47 | GuiX Menu: Help | 35 |
| 1.48 | 5.5.1 GuiX: Checksum calculator | 35 |
| 1.49 | 5.5.2 GuiX: Diz Toolkit | 36 |
| 1.50 | 5.5.4.1 GuiX: New Diz | 36 |
| 1.51 | 5.5.3 GuiX: File Splitter | 37 |
| 1.52 | 5.5.4 GuiX: UnPacker | 39 |

082a7b10-0

[illegible]

```

Release 1.2
© 1997-98, Alessandro Pedretti

English documentation by
Alessandro Pedretti & Steve Peruzzi

~1.~~Introduction.~~~~~
~2.~~Requirements.~~~~~
~3.~~Installation.~~~~~
~4.~~Use~of~xFX.~~~~~
~5.~~GuiX~~~The~GUI~of~xFX.~
~6.~~Note.~~~~~
~7.~~Bugs.~~~~~
~8.~~Future.~~~~~
~9.~~Development~tools.~~~~~
~10.~~Thanks.~~~~~
~11.~~Copyright.~~~~~

```

```
#                                     # ~12.~History.~::~::~::~::~
```

1.2 1. Introduction

xFX is a program that has some features of an old and very usable utility called DMS. Using some emulators (ShapeShifter, AmiMSX, fMSX, Speculator, etc.), the Author found many disk problems that have been solved with this utility. In other words, xFX is a tool to create disk images (packed or unpacked) from whatever storage medium in order to transfer (modem, E-Mail, etc.) or simply to archive. A little problem is the manipulation of Macintosh HD floppies. xFX is able to use many device drivers that are compatible with trackdisk.device (e.g. mfm.device, scsi.device, fmsdisk.device, ramdrive.device, etc.). This feature is usable to read and write file-disks, and so, to obtain security-only disk copies. This performance is very useful to copy the Mac HD disks, because xFX is much faster than ShapeShifter (with most probability the very slow disk copy operation is a problem of MacOS and not of the wonderful ShapeShifter). All file-disks generated with xFX (included packed files) do not make usage of a proprietary format and thus they can be used with other software without any limit. This feature allows to unpack a file-disk with any utility that fully supports XPK format.

1.3 2. System requirements

- CPU 68000 or better.
- Kickstart 2.0 (V37) or better.
- XPK libraries (optional).
- rexsyslib.library (optional).
- reqtools.library (optional).
- powerpacker.library (optional).
- Any storage driver, like:
floppy driver (3.5" DD, 3.5" HD, 5.25"), hard disk, ZIP, JAZ, floptical, CD-ROM, LS-120, etc.
- Correspondent device drivers.

If you want use the graphic user interface (GuiX), you also need:

- Kickstart 3.0 (V39) or better.
 - Magic User Interface (MUI) V 3.3 or better.
 - GZip (optional).
 - asl.library.
-

If you want access to Commodore 64 devices, you also need:

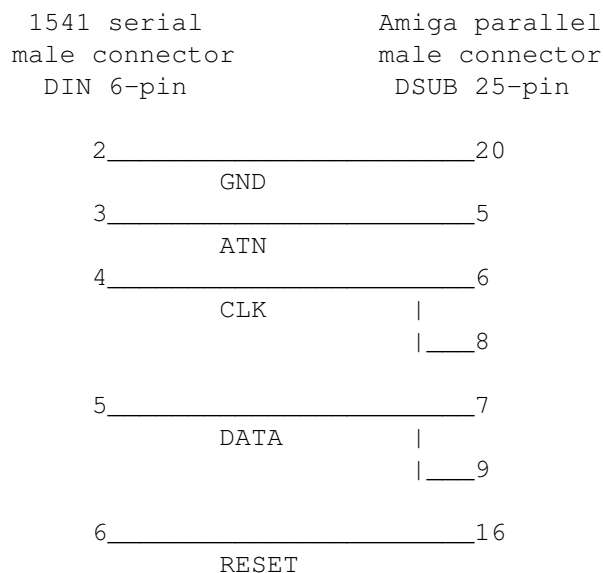
- iec.library (Easy1541 Package).
- IEC~compatible~cable (see FRODO or The 64 Emulator).

xFX allocates dynamically all resources but only when needed. If the file-disk compression is not used, the XPK libraries aren't used (and not opened). If rexsyslib.library is not installed in your Amiga, the ARexx port is not usable when xFX is running from Workbench.

1.4 IEC cable

This is a simple scheme that allows to make the IEC cable needed to connect C=64 peripherals to Amiga parallel port. xFX supports only Commodore 1541 disk drivers (or compatibles), but the iec.library (developed by Fabrizio Ferenga) is able to control virtually all C=64 devices. The 1570 and 1571 disk drivers are supported only in 1541 mode and the 1581 is unsupported, mainly because I don't have all those devices.

The following scheme is compatible with ReadySoft's "The 64 Emulator II" cable. With very old peripherals (e.g. the old VIC-1541 disk drive), this cable can't work correctly, because 8520 output signals are too low to control the old TTL integrated circuits.



In the documentation of the C=64 Emulator called Frodo (developed by Christian~~Bauer) is present the following compatible scheme. According to the author, this scheme reproduces exactly the original circuit that interfaces the 8520 of C=64 with IEC devices. This cable can work with very old devices.

Amiga parallel
male connector
DSUB 25-pin

1541 serial
male connector
DIN 6-pin

| Pin | Nome | | Nome | Pin |
|-----|------|--|------|-----|
| 14 | +5V | -----+ 7406 - - - - - (74LS05) 1k each - - - - - | | |
| 5 | PB3 | 1 \ 2 ---- O-----*--+--+----- / | ATN | 3 |
| 6 | PB4 | 3 \ 4 ---- O-*-----*--+----- / | CLK | 4 |
| 7 | PB5 | 5 \ 6 ---- O-+-*-----*----- / | DATA | 5 |
| 8 | PB6 | -----+ | | |
| 9 | PB7 | -----+ | | |
| 25 | GND | ----- | GND | 2 |

The integrated circuit can be without doubt a 7406 (that is not simply to find) or a 74LS05. You must remember to connect the IC power (pin 7 to ground and 14 to +5V).

I have built this cable with a 74LS05, because the simplified version doesn't work with my old 1541.

IMPORTANT NOTE:

The construction and the use of such cables is only at your risk. I am not responsible for any damage to your computer.

1.5 3. Installation

To install xFX, you can copy manually all files on your hard disk, otherwise you can use the included installer script.

If you want use any C= 64 device, you can build the special~cable that connects the Amiga parallel port with IEC port of this peripherals.

1.6 4. Use of xFX

At the present time, xFX can be invoked from shell or Workbench (via tooltypes, drag-&-drop, and GuiX). A built-in ARexx port can be used to control xFX from other programs.

~4.1~Shell~use.~~~~~

~4.2~Workbench~use.~~~~~

~4.3~Use~of~ARexx~port.~

~4.4~Hints.~~~~~

~4.5~Error~messages.~~~~~

1.7 4.1 Shell use

Using xFX from the shell is the easiest way to let it show all its power. All commands available from a shell, are also available through ARexx port with identical syntax.

Synopsis:

```
xFX COMMAND/A,DEVICE/A,TO/K,FILE,MODE/K,DISKBUF/K,PACK/K,PASSWD/K,
  START/K/N,END/K/N,MOUNTLIST/K,ADDICON/S,VERIFY/S,FORCEEXT/S,
  FORCEWRT/S,NOEJECT/S,VERBOSE/S,QUIET/S
```

You can call the extended-help to have a small command list, this is the output:

```
xFX V1.2 - © 1997-98 Alessandro Pedretti
```

```
COMMAND    = COPY, READ, WRITE, CHECK, INFO
DEVICE      = Device name (AmigaDOS or DEVICE.device:UNIT)
TO          = 2nd device for copy
FILE        = Input/output file for READ/WRITE commands
MODE        = Device mode (Std, SCSI, IEC)
DISKBUF     = Disk buffer path for COPY
PACK        = Xpk compression for WRITE (SUBLIB.METHOD)
PASSWD      = Password for Xpk en/decoding
START       = Start cylinder (default 0)
END         = End cylinder (default last)
MOUNTLIST   = Creates a mountlist with fmsdisk.device (READ)
ADDICON     = Adds an icon to disk image (READ)
VERIFY      = Verifies cylinders during write operations
FORCEEXT    = Always adds the .xfx extension
FORCEWRT    = Allows piping with WRITE
NOEJECT     = Inhibits auto disk eject
VERBOSE     = Explains all operations
QUIET       = Suppresses messages
```

* COMMAND: select the xFX run mode.

CHECK = surface disk error checker.
READ = reads a disk and save it into a file (you must specify it).
WRITE = Writes a file-disk to disk.
COPY = Copies a disk.
INFO = Displays disk geometry.

* DEVICE: device name to execute the selected command.

You can use the AmigaDOS device name or directly the device driver with unit number. In this case, the correct syntax is: DEVICE_DRIVER:UNIT. The extension ".device" is optional. You can use with identical result: DF0:, trackdisk.device:0, trackdisk:0

* TO: second device name.

This option is usable only with two-devices disk copy. In all other cases, this option is ignored. The correct syntax is identical to DEVICE option.

* FILE: file name to read or write.

This option is only usable with READ and WRITE commands.

* MODE: access mode.

This option allows to select the specific access mode for each device. The implemented modality are:

Std -> Access with trackdisk.device like commands (default).
SCSI -> Access SCSI direct commands.
IEC -> Access trough IEC cable with C= 1541 like commands.

* DISKBUF: temporary buffer path.

When the free memory is too little, this option allows to execute a single device copy. All temporary data are stored in a file buffer instead of memory.

* PACK: name of XPK packing and/or encrypting method.

This option is only usable with READ command. The correct syntax is:

SUBLIBRARY.METHOD

SUBLIBRARY is a four-characters algorithm name (e.g. NUKE, MASH, FAST, GZIP, etc) and METHOD is a number (from 0 to 100, default 100) that indicates the method efficiency. Examples:

xFX READ 0 Ram:MyFileDisk PACK MASH -> alghoritm MASH efficiency 100%
xFX READ 0 Ram:MyFileDisk PACK FAST.50 -> alghoritm FAST efficiency 50%

* PASSWD: XPK sublibrary password.

During READ or WRITE operations, this parameter allows to specify the en/decrypting password. Not all XPK sublibraries support this password.

* START: starting cylinder.

It's usable with READ and WRITE commands (default 0).

* END: ending cylinder.

It's usable with READ and WRITE commands. Default is the last disk cylinder (e.g. for a 3.5" floppy disk driver, default is cylinder 79).

* MOUNTLIST: mountlist file name.

At the end of file-disk writing operation (READ command), xFX generates a mountlist with source device geometry. Instead of the original device name, the mountlist contains fmsdisk.device. In this way the created file-disk can be simply mounted under AmigaDOS. The file-disk can't be packed. This option can't be used if the source device is specified like a device driver (e.g. scsi.device:0) and not like a AmigaDOS device (e.g. DH0:), this because the same device driver can control more partitions.

* ADDICON:

This switch adds an icon with tooltypes to the file-disk generated with READ command.

* VERIFY:

This switch turns on the verify option during write to disk operations (WRITE and COPY commands).

* FORCEEXT:

This switch forces xFX to add the ".xfx" extension to file name.

* FORCEWRT:

This switch forces xFX to use a FIFO compatible devices like PIPE for the interprocess communication. It's analogous to GZip -f option.

* NOEJECT:

This option inhibits the automatic disk ejection during copy operation. This function is performed by trackdisk.device TD_EJECT command and thus works only with devices that support this command.

* VERBOSE:

This option displays more informations that explain all operations.

* QUIET:

This switch suppresses all messages (except error messages).

~4.1.1~Examples~~~~~

~4.1.2~AmigaDOS~Script~

1.8 4.1.1 Examples

```
> xFX info scsi.device:0
displays the hard-disk geometry.

> xFX check mfm:1
executes a surface verifying of floppy disk in drive 1 using mfm.device.
This disk must be in Ms-Dos-like format (e.g Mac, MSX, ecc).

> xFX read DF0: RAM:MyDisk
creates "MyDisk" file-disk in Ram: with the disk image in Amiga format
(trackdisk.device) not packed/encrypted. The source disk is inserted
into the drive 0.

> xFX read DF0: RAM:MyDisk MOUNTLIST Ram:FF0
As upper, plus mountlist generation.

> xFX read PS2: RAM:MyDisk PACK MASH START 20 ADDICON
creates a file-disk called "MyDisk" in Ram: from cylinder 20 to 79, using
"MASH" compression with 100% efficiency and adds an icon.

> xFX write mfm.device:1 RAM:MyDisk START 10 END 20 VERIFY
writes (with verify) the disk-image of "MyDisk" file starting from cylinder
20 to 79. If the source file size is incompatible with the selected
cylinder number, an error is displayed. The unpacking is automatically
performed with crunched files.

> xFX copy mfm.device:2 MODE Std
executes a copy of disk inserted into the drive 2, using the ram buffer.

> xFX copy DF0: TO trackdisk.device:1 VERIFY
copies (cylinder by cylinder) form unit 0 to 1 with verify. In this way,
the required ram amount is very small.

> xFX copy DEVICE scsi.device:3 DISKBUF DH0: MODE SCSI
copies the entire disk from scsi.device unit 3 (removable media like ZIP)
with temporary buffer in DH0:

> xFX read 8 RAM:MyDisk MODE IEC
creates a file-disk called "MyDisk" in ram disk reading from the 1541 disk
driver with device number 8.

> xFX write DF0: PIPE:MyDisk.xfx/10240/1 FORCEWRT
using the PIPE: device, creates a temporary file-disk that can be reade
by another process.
```

1.9 4.1.2 AmigaDOS Script

Starting from 1.2 release, xFX can be used with fifo devices (like PIPE:, FIFO:, IXPIPE:, etc). In this way you can redirect input and output to other CLI programs (e.g. packers and unpackers) making simple AmigaDOS scripts.

4.1.2.1 zFX

This is a simple interface script between xFX and GZip. It's very useful with .adz file-disks that are normal images of AmigaDOS disks packed with GZip and used with UAE. This feature is also implemented into GuiX. The syntax of this script is:

```
zFX COMMAND/A,DEVICE/A,TO/K,FILE,MODE/K,DISKBUF/K,PACKEFF/K,START/K,END/K,
    VERIFY/S,NOEJECT/S,VERBOSE/S,QUIET/S
```

```
COMMAND    = COPY, READ, WRITE, CHECK, INFO
DEVICE     = Device name (AmigaDOS or DEVICE.device:UNIT)
TO         = Destination device for copy
FILE       = Input/output file for READ/WRITE commands
MODE       = Device mode (Std, SCSI, IEC)
PACKEFF    = GZip packing efficiency (1...9)
DISKBUF    = Disk buffer path for COPY
START      = Start cylinder (default 0)
END        = End cylinder (default last)
VERIFY     = Verifies cylinders during write operations
NOEJECT    = Inhibits auto disk eject
VERBOSE    = Explains all operations
QUIET      = Suppresses all messages
```

For the correct work of this script, you must have GZip in your default command path, the PIPE: device and then the correspondent mountlist and handler (queue-handler). zFX is able to mount automatically this device if not already mounted.

1.10 Workbench use

Under Workbench, xFX can't work if the reqtools.library is not present. If you want to use the graphic user interface, the Magic User Interface (MUI) is needed. You can find it on Aminet archive.

xFX can be used under Workbench in three ways: (1) trough tooltypes, (2) Appicon drag-&-drop, (3) GuiX - the graphic user interface of xFX.

4.2.1~Tool~Types~~

4.2.2~Drag~&~Drop~

4.2.3~GuiX~~~~~

1.11 4.2.1 Tool Types

In this way, a file-disk can be simply restored into the original support. A double click on file-disk project icon (that can be automatically added by xFX) can start the xFX write process. This icon must have xFX as default tool. The tooltypes actually usable are:

Non optional tool types:

```
COMMAND    -> Name of command to execute. At this time, the only supported
```

command is WRITE, that restores a file-disk to the original storage medium.

DEVICE -> Name of device to use. For the correct syntax into Shell~use section. If the device name is omitted or a question mark ("?.") is placed instead of it, the reqtools device requester is opened in order to simplify the selection.

Optional tool types:

START -> Starting cylinder (default is 0).

END -> Ending cylinder (default is the last and is dependent on the output device).

GEOMETRY -> It's a group of five values separated with one or more spaces: total sectors, size of each sector, sectors per cylinder, number of heads (surfaces). These informations are used from xFX to check if the file-disk is compatible with selected output device. If this tooltype is not set, xFX use another way (but less reliable).

VERIFY -> Executes a verify during write operations.

NOEJECT -> Disables automatic disk ejection (if it's possible).

QUIET -> xFX runs quiet displaying only error messages.

1.12 4.2.2 Drag & Drop

When xFX is running under Workbench (double click on icon), an AppIcon appears. If a file-disk is dragged on this icon, this is automatically used to regenerate the starting disk. To quit xFX, you can use the "Kill xFX" item into the tools menu. In another way, you can double click on xFX AppIcon. A file requester asks you confirm to proceed with this operation.

All read and write operations are graphically displayed in a window containing the "Abort" button. If this gadget is pressed or you type ESC or "A" (upper or lower case) on your keyboard, all operations are stopped.

1.13 4.3 Utilizzo tramite porta ARexx

If in your system is installed rexxsyslib.library and xFX is running under Workbench (double click on icon), an ARexx port called "xFX.N" is created. N is a progressive number that is used to avoid the duplication of name port if two or more xFX copies are running at same time. Usually, if only one xFX copy is running, the port name is "xFX.1".

Each command sent through ARexx port, sets correctly the return code (RC ARexx variable), the RESULT variable for the return value and a special variable called XFX.LASTERRO with the error message that can be generated during xFX run. The supported ARexx commands are:

```

NOVIEWERR      -> Disables the displaying of error requesters, but all er-
                  rors are still reported in XFX.LASTERERROR variable.
QUIT           -> Exits from xFX.
REQERR "TEXT"  -> Displays a text in an error requester.
VERSION        -> Returns the current xFX version in RESULT.
VIEWERR        -> Activates the displaying of all errors (default mode).

```

Other commands follow the same syntax of shell ones:

```

> info "DF0:"
> read "diskspare.device:0" "MyDisk.dsk" pack SHRI NOEJECT

```

1.14 4.4 Hints

4.4.1 Packing method choice.

The packing method choice can be very important to obtain much compact filedisk and to waste a minimal amount of time to process a disk. A very good and fast algorithm is MASH. With slower Amigas, NUKE is sufficient for most applications but the efficiency is not the same of MASH. The best compression ratio (but with slow speed) can be obtained with SHRI and can be used to transfer disk images with a modem. This packing method is very useful with MSX disk, because can reduce a 720 Kb disk in 160-230 Kb file.

4.4.2 xFX and ShapeShifter.

The performances that xFX offers to ShapeShifter users, are very useful: HD disk copy, conversion of a disk into a file-disk for direct use with the emulator and the conversion of a entire ShapeShifter device-disk into a file-disk. This last possibility is very interesting, because it allows to create a Mac file-disk (e.g. to insert in a CD-ROM). Trough Consultron's mro.device, it's possible to convert 800 Kb A-Max disks into file-disks for ShapeShifter direct use. Sometimes, using a xFX file-disk, ShapeShifter does not recognize it immediately and asks for format. If you won't allow this operation, ShapeShifter will rebuild automatically the desktop and the file-disk icon will appear.

4.4.3 xFX and fMSX.

xFX can be used to archive MSX disks in simply way using mfm.device. The XPK SHRI packing method allow to obtain the best compression ratio. File-disks, created with xFX, can be directly used with fMSX.

1.15 4.5 Error messages

xFX can returns all error messages in three different ways: Shell error messages (if it does run under shell), reqtools message requesters (if does run under Workbench) and "XFX.LASTERERROR" ARexx variable. Here we report, alphabetically ordered, all possible error messages:

* Aborted.

It's not a real error message. It allows to inform the user that xFX run is

aborted (e.g. CTRL+C).

* Can't add icon

It's not possible to add the file-disk icon , when the ADDICON option is selected. A disk access problem can generate this error.

* Can't allocate crunch/decrunch buffer !

The crunch/decrunch allocation failed, because your system has not enough free memory.

* Can't allocate device requester

Reqtools.library is not able to allocate the file requester. Maybe the memory run out.

* Can't create AppIcon

During Workbench run, xFX can't add the AppIcon. With most probability, your chip memory run out.

* Can't create ARexx port

The program can't create the ARexx port. Maybe your memory run out.

* Can't create MenuItem

During Workbench run, xFX can't add items to tools menu.

* Can't create mountlist

It's not possible to create the mountlist (MOUNTLIST option), because xFX can't recognize the file system of selected device driver. Usually this problem appers when you access directly a device driver. To solve this problem, you can use the correspondent AmigaDOS device,

* Can't find disk object

It's not possible to find the disk object selected with drag & drop on AppIcon.

* Can't open DEVICE unit N

The device DECICE unit N can't be opened, because the device is temporarily busy and allocated by another program, or simply it's not installed in your computer.

* Can't open progress window

It's not possible to open the progress window.

* Can't read drive geometry

The selected device driver does not respond to TD_GETGEOMETRY command. This problem can be bypassed using the correspondent AmigaDOS device.

- * Can't use an assign
Instead of a normal AmigaDOS device, you have used an assignment.
 - * Command COMMAND require option OPTION
The specified command (COMMAND) requires the option OPTION.
 - * Device name too long
The size of specified device (DEVICE and TO options) is too long.
 - * Disk buffer full
On your disk, there is not enough free space to create the buffer.
 - * Disk buffer not usable
The disk buffer is not usable. With most probability, the specified disk is write protected.
 - * Disk buffer path too long
The disk buffer path is too complex (long).
 - * Disk has errors !
This message remembers only that the processed disk has read or write errors.
 - * Disk write protected in DEVICE unit N
The disk inserted into DEVICE unit N, can't be used, because it's write protected.
 - * DOS Error: XXX
It's a dos error message. For more informations, please consult AmigaDOS manual.
 - * File too large
The selected file-disk can't be used with WRITE command, because the size is not compatible with geometry of destination device.
 - * File too short
The selected file-disk can't be used with WRITE command, because the size is not compatible with selected range of cylinders (START and END options).
 - * IEC Error: IEC device N not present
The IEC device with unit number N is not present. Maybe the device is switched off or the IEC cable is not correctly inserted.
 - * Illegal character in PACK option
-

You have used an illegal character to specify the efficiency of packing method.

* Illegal combination of START/END cylinders

The starting cylinder is upper to ending.

* Illegal command

Unknown command (COMMAND option).

* Illegal device name

The selected device is incompatible with xFX device syntax.

* Illegal ending cylinder

You have specified the END option with a cylinder number greater than top cylinder or lower than zero.

* Illegal pack method

The packing efficiency is not in 0-100 range.

* Illegal starting cylinder

You have specified the START option with a cylinder number greater than top cylinder or lower than zero.

* Illegal unit number

You have used a not valid unit number.

* Incompatible options TO and DISKBUF

The two-devices disk copy is not compatible with disk buffer.

* IORequest not allocated

It's not possible to allocate IORequest. Maybe the memory is not enough.

* KickStart 2.04+ needed !

Your system is not compatible with xFX.

* LIBNAME.library VXX.XX or higher needed !

xFX can't open the LIBNAME library with specified version. Maybe, the library is not installed in your system.

* Memory fragmentation is too high !

The memory fragmentation is too high to make a single-swap copy. You can reset your Amiga, or use the disk buffer.

* MsgPort not allocated

xFX can't allocate MsgPort. Maybe the memory is full.

* No device selected
You don't have specified the DEVICE tooltype.

* No disk in DEVICE unit N !
No disk is present in device DIVICE unit N.

* Not enough disk space for buffer
Your disk space is not sufficient for disk buffer (DISKBUF option).

* Not enough memory for single swap copy
The single swap copy can't be performed, because your system memory is not enough. You can use disk buffer, or a second device (TO option).

* Only WRITE command is supported through Tool Types
Instead of WRITE, you have used another command via tooltypes.

* Option OPTION ignored
You have specified a option (OPTION) that is not used by command. This option is ignored.

* Out of chip memory !
The chip memory is not enough.

* Read error at cylinder N
During read operation, the cylinder N reports an error.

* SCSI Error: XXX.device is not a SCSI compatible device
You have selected the SCSI access with a non-SCSI device. Please use the standard access mode (default).

* SCSI Error: Type of SCSI device not supported
The specified device is SCSI but xFX is not able to control it (e.g. scanner, tape, etc).

* SCSI Error: Sense Error
The device responds in faulty way. xFX shows a synthetic description of the problem, all bytes of Sense message and, if possible, a complete description (ASC and ASQ). This information is included into "SCSI_Errors.dat" file that must be placed into the directory where xFX is located or into directory pointed by XFX: assign.

* Screen too small
The progress window can't be opened, because the screen is too small.

* Selected device is not a disk or is not mounted
The selected device does not correspond to device disk, or it is not mounted.

* SOURCE and DESTINATION are not compatible !
The source and destination disks have incompatible geometry.

* SOURCE and DESTINATION are the same device and unit
The source and destination are the same device (COPY command).

* Specify the input file-disk
With WRITE command, you need specify the input file.

* Specify the output file
With READ command, you need specify the output file.

* Too many objects !
You have dragged more the one object on xFX AppIcon. xFX supports only one object at a time.

* Tool type COMMAND required
You need COMMAND tooltype.

* Verify error at cylinder N
xFX has found a verify error at Nth. cylinder.

* Write error at cylinder N
xFX has found a write error at Nth. cylinder.

* XPK Error: XXX
It's a XPK library error. Check the XPK documentation.

1.16 6. Note

6.1 General notes.

- xFX does not need a large memory segment to allocate the copy-disk buffer. Thus, it have not problems to run even if memory fragmentation is very high.

- xFX works correctly with this tested devices: trackdisk.device, mfm.device, ramdrive.device, fmsdisk.device, scsi.device, atapi.device (Oliver Kastl), diskspare.device, mro.device (anche con disco A-Max), floppy.device 4.1 (Orhan Dagistanli), icddiskide.device.

- When any device receives the TD_GETGEOMETRY command, rather than giving a negative answer, crashes the system (this is not a xFX bug), because does not support this command. A little list of these devices: fmsdisk.device, multios.device (Jim Drew). This problem is solved mounting the correspondent AmigaDOS device and using that with xFX. In this way, xFX does not use the trackdisk.device command TD_GETGEOMETRY, but uses the DosList.

- atapi.device (old 113.1 version) and icddiskide.device don't support TD_GETGEOMETRY (without crashing the system). To use these devices, you can mount and use the correspondent AmigaDOS devices.

- xFX is capable to recognize the AmigaDOS device without file system (e.g. SER:, PAR:, etc), but this is not true for device drivers. If you call xFX with SER:, the program displays an error, but if you call xFX with serial.device:0, your Amiga crashes !

6.2 Floppy disk formats.

Computer = computer name.
 Format = format name.
 Drv = size of disk (inches).
 Dsk = disk type SS = single side single density.
 SD = single side double density.
 DS = double side single density.
 DD = double side double density.
 HD = high density.
 ZP = Iomega Zip-100.
 Size = size in Kbytes.
 Sec = total number of sectors.
 SecSize = size of each sector (bytes).
 TotCyl = total number of cylinders cilindri.
 SecCyl = number of sectors by cylinder.
 Surf = number of sides.
 Dev = device driver for xFX use:
 1 = trackdisk.device
 2 = diskspare.device
 3 = floppy.device
 4 = mfm.device
 5 = mro.device

| Computer | Format | Drv | Dsk | Size | Sec | SecSize | TotCyl | SecCyl | Surf | Dev |
|----------|-----------|-----|-----|-------|--------|---------|--------|--------|------|---------|
| Amiga | trackdisk | 3.5 | DD | 880 | 1760 | 512 | 80 | 22 | 2 | 1 |
| Amiga | diskspare | 3.5 | DD | 984 | 1968 | 512 | 82 | 24 | 2 | 2 |
| Amiga | floppy | 3.5 | DD | 1148 | 2296 | 512 | 82 | 28 | 2 | 3 |
| Amiga | trackdisk | 3.5 | HD | 1760 | 3520 | 512 | 80 | 44 | 2 | 1 |
| Amiga | diskspare | 3.5 | HD | 1968 | 3963 | 512 | 82 | 48 | 2 | 2 |
| Amiga | floppy | 3.5 | HD | 2050 | 4100 | 512 | 82 | 50 | 2 | 3 |
| Amiga | Zip-100 | 3.5 | ZP | 98304 | 196608 | 512 | 384 | 512 | 1 | 1\$^1\$ |

| | | | | | | | | | |
|-----------|------|----|------|------|-----|----|-------|---|---|
| Atari ST | 3.5 | SD | 360 | 720 | 512 | 80 | 9 | 1 | 4 |
| Atari ST | 3.5 | DD | 720 | 1440 | 512 | 80 | 18 | 2 | 4 |
| C= 1541 | 5.25 | SS | 171 | 683 | 256 | 35 | 17-21 | 1 | 6 |
| C= 1571 | 5.25 | DS | 342 | 1366 | 256 | 35 | 17-21 | 2 | 6 |
| C= 1581 | 3.5 | DD | 800 | 3160 | 256 | 80 | 40 | 2 | 6 |
| Macintosh | 3.5 | DD | 800 | 1600 | 512 | 80 | 20 | 2 | 5 |
| Macintosh | 3.5 | HD | 1440 | 2880 | 512 | 80 | 36 | 2 | 4 |
| Ms-Dos | 5.25 | SD | 160 | 320 | 512 | 40 | 8 | 1 | 4 |
| Ms-Dos | 5.25 | SD | 180 | 160 | 512 | 40 | 9 | 1 | 4 |
| Ms-Dos | 5.25 | DD | 320 | 640 | 512 | 40 | 16 | 2 | 4 |
| Ms-Dos | 5.25 | DD | 360 | 720 | 512 | 40 | 18 | 2 | 4 |
| Ms-Dos | 3.5 | SD | 360 | 720 | 512 | 80 | 9 | 1 | 4 |
| Ms-Dos | 2.5 | DD | 720 | 1440 | 512 | 80 | 18 | 2 | 4 |
| Ms-Dos | 3.5 | DD | 720 | 1440 | 512 | 80 | 18 | 2 | 4 |
| Ms-Dos | 5.25 | HD | ? | ? | 512 | ? | ? | 2 | 4 |
| Ms-Dos | 3.5 | HD | 1440 | 2880 | 512 | 80 | 36 | 2 | 4 |
| Ms-Dos | 3.5 | ED | 2880 | 5760 | 512 | 80 | 72 | 2 | 4 |
| MSX | 3.5 | SD | 360 | 720 | 512 | 80 | 9 | 1 | 4 |
| MSX | 3.5 | DD | 720 | 1440 | 512 | 80 | 18 | 2 | 4 |
| Spectrum | 3.5 | SD | 400 | 800 | 512 | 80 | 10 | 1 | 4 |
| Spectrum | 3.5 | DD | 800 | 1600 | 512 | 80 | 20 | 2 | 4 |

=====
 \$^1\$ ATAPI version automatically mounted by IDEFix '97@.

1.17 7. Bugs

xFX is a very young software and, with most probability, contains some little bugs. For suggestions, questions and bug reports, please you refer to the Author~address.

1.18 8. Futuro

- DMS depacking.
- Support of single sided disks.
- PowerPacker compression.
- Direct access to 5.25 inches floppy driver to read C= 64 floppy disks.
- GuiX tool to recognize the file-disk type.
- Read and write function for boot-blocks.
- Toolkit for .d64 files.
- Support for SCSI CD writers (data and CDDA mode).
- Others ...

1.19 9. Development tools

- Development hardware:
 A1200-T 030/882 @50 MHz, 32 Mb Fast Ram, HD 1080 Mb ATAPI, CD-ROM 4x ATAPI, Zip-100 ATAPI, Kick 3.0.

- Hardware for beta-testing:
A4000 040 @25 MHz, 16 Mb Fast Ram, CV 64, HD 3.2 Gb, CD-ROM, Kick 3.0
A3000 030/882 @25 MHz, 2 Mb Chip/12 Mb Fast, Merlin, HD 2 Gb, Kick 3.1
A500 512Kb Chip/1.8Mb Fast, HD 20 Mb, Kick 2.04
- Software:
Gcc V2.7.2, Libnix, Cygnus Editor Professional V3.5 , GoldED 4, XPK User & Developer Archive, Enforcer V37, MungWall.

1.20 10. Thanks

- Carolyn Scheppner (Sushi).
- Christian Bauer (ShapeShifter & Frodo).
- Dave Jones & Matt Dillon (fmsdisk.device).
- Dietmar Eiler (GoldEd).
- Dirk Stöcker, Bryan Ford, Urban Dominik Müller, Christian von Roques & Co. (XPK libraries).
- Fabrizio Farenga (Easy1541).
- Fred Fish and his partners (ADE).
- Free Software Foundation (packages GNU).
- Jaun Antº Gómez (AmiMSX, AmiMasterGear, AmiGameBoy).
- Klaus Deppisch (DiskSpareDevice).
- Marat Fayzullin & Hans Guijt (fMSX).
- Michael Sinz (Enforcer).
- Michiel Plet (Professional File System).
- Nico François & Mangus Holmgren (reqtools.library).
- Oliver Kastl (IDEFix & atapi.device).
- Roman Patzner (icons).
- Stefan Stuntz (MUI).
- William James (Speculator).

Thanks to all xFX users and especially for your very useful advices:

- Nagilum (async I/O).
- Mikael Lund (danish transaltion of catalogs).

Special thanks to all my beta testers (Steve Peruzzi, Andrea Orsucci & IdeaTeam) and to all my friends. At last but not least, I want to thank all people (and are very much) that have contributed to make Amiga so wonderful.

All people unitentionally not thanked in this document, please don't offend.

1.21 11. Copyright

All trademarks and softwares directly or indirectly referred in this document, are copyrighted from legal owners. xFX is a program freeware and can be spreaded trough Internet, BBS, CD-ROM and each other electronic form. The Author of this program accepts no responsabilty for hardware/software damages resulting from the use of this package.

xFX
is a software developed in 1997/98
by Alessandro Pedretti
All rights reserved.

Alessandro Pedretti
via Enrico Mattei 16/D
20018 Sedriano (MI) - Italia
Tel. +39-2-90110528
Fax. +39-2-90110528
1st E-Mail: alex@indigo.farma.unimi.it
2nd E-Mail: apedretti@mail.est.it

1.22 12. History

0.10 (internal):

- First release.

0.11 (internal):

- Solved the access inhibition problem when device in use.
- Now the ARexx error messages are reported correctly.
- The device unit number is not more limited to 10.
- Added the IDCMP DISKINSERTED message to notify the disk swap.

1.0 (1st public release):

- AmigaGuide® documentation.
- Installer.
- ARexx examples.
- Removed a little bug concerning to two drive copy.
- Added automatic stack checker.

1.1 (2nd public release):

- GuiX: The graphic user interface of xFX.
- xFX is now localizable.
- Variables optimisation to reduce the code size.
- ".xfx" extension is added automatically to file name.
- VERBOSE and FORCEEXT switches.
- Cursor blanking to accelerate the text output.
- Progress bar under Cli.
- Asynchronous track reading during packing.
- All buffers are now allocated in public memory.
- The size of XPK chunks is totally used. In this way, the packing efficiency is more strong.

1.2 (3rd public release):

- SCSI direct commands support.
 - Eject inhibition of SCSI devices during disk access.
 - Complete management of SCSI errors.
 - Support for IEC devices (only in 1541 mode).
 - GuiX can view/add/remove a FILE_ID.diz to file-disks XPK packed.
 - GuiX can control IEC devices.
-


```

      ###      #####
      #####
      #####
      #####
#      #####      ###
#      #####      ###
#####      ### #
#####      ###
####      ##
#      #
#      #

```

1.24 5.1 GuiX: Introduction

GuiX is the graphic user interface of xFX. This software is based on Magic User Interface (MUI, © Stefan Stunz) and was developed to simplify as much as possible the xFX use. It has built-in tools that allow some functions not implemented in xFX. In-line help and bubble help allow a very intuitive use of GuiX and xFX.

Unlike some others graphic interfaces, GuiX communicates bidirectionally with xFX through ARexx port. This communication method allows a high integration of these two programs.

Unlike xFX, GuiX need the KickStart 3.0 or better to work.

1.25 5.2 GuiX: Menu

The structure of GuiX menu is:

```

Project
~Start~~~~~
~Delete~Files~
-----
~Preferences~~
-----
~About~MUI~~~~
~About~GuiX~~~
-----
~Iconify~~~~~
~Quit~~~~~

```

```

Command
~Read~~~~~
~Write~~~~~
~Copy~~~~~
~Check~~~~~

```

```

Extras
~Checksum~~~~~
~DizToolkit~~~

```

```
~Splitter~~~~~
~UnPacker~~~~~
-----
~MUI~Prefs~~~~~
-----
~Help~~~~~
```

1.26 5.3 GuiX: The Main Window

```

+-----+
|*| GuiX - © 1997-98 Alessandro Pedretti                                     |%|||
+-----+
|+----- Device -----+ /-----\ /-----\ /-----\ /-----\ |
||+-----++++++|| | Read | | Write | | Copy | | Check | | <- F
A -> ||| ||^||Std|| / +-----+-----+-----+
||+-----++++++|| | +-----+ +-----+
|+-----+ | | [] Add Icon | | BLFH |*|||
|+----- File -----+ | | [] Pack | | BLZW | |||
||+-----+++++|| | | [] MountList | | CBR0 | |||
B -> ||| ||^|| | | [] No Eject | | CRMS +--+||
||+-----+++++|| | +-----+ | DLTA |=|||
|+-----+ | +-----+ | DMCB +--+||
|+----- Cylinders -----+ | |^|| XPK | | DUKE |=|||
|| +-----+ | +-----+ +-----+
C -> || Start | 0| || | +-----+ +--+||
|| +-----+ | Efficiency | |100|||
|| +-----+--+|| | +-----+ +--+||
D -> || End | |79|| | +-----+ +--+||
|| +-----+--+|| | Password | |||
|+-----+ | +-----+ +--+||
| | +-----+ +--+||
|+-----+ | Mountlist | ||^|||
E -> || Start | | +-----+ +--+||
|+-----+ +-----+
| | /|
+-----+

```

Description:

A) Device name.

It's the device name used for all operations. You must use the same rules that are explained into the `Shell~use~section`. If you press the button localized at the right of this string gadget, you can activate the `device~selector`.

Using the special cyclic gadget, you can select one of this access mode:

```
Std      <- Access with trackdisk.device compatibile commands
SCSI     <- Access with SCSI direct commands
IEC      <- Accesso trough IEC cable with C= 1541 compatibile commands
```

B) File-disk name.

It's the file name that are used for reading and writing operations.

C) Starting cylinder slider.

D) Ending cylinder slider.

This sliders are equivalent to START and END options of Cli and indicate the starting and ending cylinders to execute all operations.

E) Start button.

When this button is pressed, xFX is activated with ARexx messages.

F) Function register.

With this register, you can select the command that xFX executes when the Start button is pressed:

~12.3.1~Read~~~~~

~12.3.2~Write~~~~~

~12.3.3~Copy~~~~~

~12.3.4~Check~~~~~

1.27 5.3.1 GuiX: Read

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|*| GuiX - © 1997-98 Alessandro Pedretti                                     |%|||
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|+----- Device -----+ /-----\ /-----\ /-----\ /-----\ |
||+-----++-++-++-++-+| | Read | | Write | | Copy | | Check | |
|||          ||^||Std|| /          +-+-----+-----+-----+-----+
||+-----++-++-++-++-+| | +-----+ +-----+-----+-----+
|+-----+-----+-----+ | | [] Add Icon | | BLFH          |*|||
|+----- File -----+ | | [] Pack      | | BLZW          | ||| <- A
||+-----++-++-++-++-+| | | [] MountList | | CBR0          | |||
|||          ||^|| | | [] No Eject | | CRMS          +-+|||
||+-----++-++-++-++-+| | +-----+ | DLTA          |=|||
|+-----+-----+-----+ | +-+-----+ | DMCB          +-+|||
|+----- Cylinders -----+ | ^||      XPK      | | DUKE          |=|||
||          +-+-----+ | +-+-----+ +-----+-----+
|| Start | 0|          || |          ^ Efficiency |          |100||| <- C
||          +-+-----+ | |          |          +-----+-----+
|| End |          |79|| | B          +-----+-----+
||          +-----+ |          Password |          ||| <- D
|+-----+-----+ |          +-----+-----+
|          |          |          +-----+-----+
|+-----+-----+ |          Mountlist |          ||^||| <- E
||          Start          | |          +-----+-----+
|+-----+-----+ +-----+-----+
+-----+-----+-----+-----+-----+-----+-----+-----+
|          |          |          |          |          |          |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Description:

A) Listview of XPK compression methods.

This list is disabled if the Pack option (section B) is not checked. If do you want use the GZip compression, this list is disabled.

B) Optional functions.

Cyclic gadget:

XPK -> Activates XPK libraries to compress.

GZip -> Uses GZip to compress.

Checkboxes:

Add Icon -> Adds an icon to file-disk.

Pack -> Activates the file-disk compression.

MontList -> Generates automatically the fmsdisk.device mountlist.
(see ~Shell~use~)

No Eject -> Inhibits the automatic disk ejection.

C) Slider of XPK efficiency compression.

This option is disabled if the Pack option (section B) is not checked.

D) XPK password (optional).

This option is disabled if the Pack option (section B) is not checked. It allows to specify an encryption password if needed.

E) Mountlist file to create.

This item is not activated if the Mountlist option (section B) is not checked.

1.28 5.3.2 GuiX: Write

```

+--+-----+-----+-----+-----+-----+-----+-----+-----+
|*| GuiX - © 1997-98 Alessandro Pedretti                                     |%|||
+--+-----+-----+-----+-----+-----+-----+-----+-----+
|+----- Device -----+ /-----\ /-----\ /-----\ /-----\ |
||+-----+-----+-----+ | Read | | Write | | Copy | | Check | |
|||                               ||^|| /-----++      +-----++|
||+-----+-----+-----+ |                                     ||
|+-----+-----+-----+ |                                     ||
|+----- File -----+ |                                     ||
||+-----+-----+-----+ |               A               B      ||
|||                               ||^|| |               |          ||
||+-----+-----+-----+ |               *               *      ||
||+-----+-----+-----+ | +-----+-----+-----+-----+ ||
|+-----+-----+-----+ | | [] Verify          [] No Eject | ||
|+----- Cylinders -----+ | +-----+-----+-----+-----+ ||
||      +--+-----+-----+ | |               +-----+-----+ ||
|| Start | 0|               || | Password |               | ||
||      +--+-----+-----+ | |               +-----+-----+ ||
||      +-----+-----+-----+ | |               *          ||
||      End |               |79|| |               |          ||
||      +-----+-----+-----+ | |               C          ||
|+-----+-----+-----+ | |                                     ||
|+-----+-----+-----+ | |                                     ||
||      Start               | | |                                     ||

```

```

|+-----+ +-----+
|+-----+
|                                     |
|+-----+

```

Description:

A) Verify checkbox.

This checkbox activates the cylinder verification.

B) No Eject checkbox.

If checked, this option inhibits the automatic disk ejection.

C) Password item (optional).

If needed, in this string gadget you can specify a decryption password.

1.29 5.3.3 GuiX: Copy

```

+-----+ +-----+
|*| GuiX - © 1997-98 Alessandro Pedretti |%||
+-----+
|+----- Device -----+ /-----\ /-----\ /-----\ /-----\ |
||+-----+ +--+| | Read | | Write | | Copy | | Check | |
||| ||^|| /-----+ +-----+ +-----+ +-----+
||+-----+ +--+| | | |
|+-----+ | +----- Destination Device -----+ |
|+----- File -----+ | |+-----+ +--+| |
||+-----+ +--+| | || ||^|| | <- A
||| ||^|| | |+-----+ +--+| |
||+-----+ +--+| | +-----+ |
|+-----+ | +-----+ |
|+----- Cylinders -----+ | | [] Verify [] No Eject | | <- B
|| +--+-----+ | | |
|| Start | 0| | | |+----- Method ----+- Buf. Path --+ | |
|| +--+-----+ | | | |
|| +-----+ +--+| | ||[] Memory || +-----+ +--+| |
|| End | |79| | | ||[] Disk Buffer || | ||^|| | <- C
|| +-----+ +--+| | ||[] Device || +-----+ +--+| |
|+-----+ | | |
| | |+-----+ +--+| |
|+-----+ | +---/\-----+ |
|| Start | | \ D |
|+-----+ +-----+
|
| |
+-----+

```

Description:

A) Destination device

If the radio button Device is checked, you can specify the name of destination device. You must use the same rules that are explained into the section Shell~use. You can use the device~requester pressing the special

1.31 5.4 GuiX: Device requester

With the GuiX device requester, you can select with no limitations mounted or unmounted AmigaDOS devices and device drivers. It has the following structure:

```

+-----+
|*| Device Selector                                     |%|||
+-----+
| /-----\ /-----\ /-----\ +----- Geometry -----+|
A -> | |Mounted| |Unmounted| |Drivers| |                               ||
|/          +-+-----+-+-----+-+ | Device Driver -          ||
|| +-----+-----+-----+ | |                               ||
|| | CC0:                |*| | |                               Unit - ||
|| | CD0:                | | | |                               ||
|| | DF0:                | | | | Size (KBytes) -              ||
|| | DF1:                | | | |                               ||
|| | DF2:                | | | | Tot. Sectors -               ||
|| | DH0:                | | | |                               ||
|| | DH1:                | | | | Sector Size -               || <- B
|| | DH2:                | | | |                               ||
|| | PC0:                | | | | Cylinders -                  ||
|| | PC1:                | | | |                               ||
|| | PC2:                | | | | Sectors/Cyl. -              ||
|| |                    +-+ | |                               ||
|| |                    |=| | | Surfaces -                   ||
|| |                    +-+ | |                               ||
|| |                    |=| | | Sectors/Track -              ||
|| +-----+-----+-----+ | |                               ||
| +-----+-----+ +-----+ +-----+ +-----+|
C -> |Device | | | | | Ok | <- D | Cancel | | <- E
|      +-----+-----+ +-----+ +-----+ +-----+|
+-----+
|                               | / |
+-----+

```

Description:

A) Register of functions.

~Mounted~~~ -> Mounted AmigaDOS devices.

~Unmounted~ -> Unmounted AmigaDOS device.

~Device~~~~~ -> Device drivers.

B) Geometry of selected device.

The geometry is showed automatically in Mounted e Unmounted mode, when you select a device from listview. In Device mode, you must press the Info button to display the geometry.

C) Name of selected device.

You must use the same conventions explained into Shell~use section.

D) Confirm button.

Confirms the selected device.

E) Cancel button.

Cancels the selection and closes the window.

Menu items:

Device List

```
Rescan    <--- Repeats the scansion of devices (short key RAMIGA + R)
Close     <--- Closes the window (short key RAMIGA + C)
```

Mode

```
Mounted      <---  Shows the mounted devices (short key RAMIGA + M)
UnMounted    <---  Shows the unmounted devices (short key RAMIGA + U)
Drivers      <---  Shows the device drivers (short key RAMIGA + D)
```

If you activate the IEC access, it's not possible to use the device requester, because it's substituted by a slider that is showed into GuIX main window. This slider allows to select the IEC unit number (from 8 to 11), like into following figure:

```

+-----+
|*| GuiX - © 1997-98 Alessandro Pedretti                                     |%|||
+-----+
+-----+ Device +-----+ /-----\ /-----\ /-----\ /-----\ |
||+-----+-----+-----+| | Read | | Write | | Copy | | Check |
----> ||8|                ||IEC|| /          +-----+-----+-----+-----+
||+-----+-----+-----+| | +-----+ +-----+-----+-----+
+-----+ +-----+ | | [] Add Icon | | BLFH |*|||
+-----+ File +-----+ | | [] Pack | | BLZW | |||
||+-----+-----+-----+| | | [] MountList | | CBR0 | |||
|| |                ||^|| | | [] No Eject | | CRMS | +-+||
||+-----+-----+-----+| | +-----+ +-----+-----+-----+
+-----+ +-----+ | +-----+ +-----+ | DMCB | +-+||
+-----+ Cylinders +-----+ | ^|| XPK | | DUKE | |=||
|| +-----+ +-----+ | +-----+ +-----+-----+-----+
|| Start | 0| | | | Efficiency | | 100|||
|| +-----+ +-----+ | | | +-----+-----+-----+
|| End | |79|| | | +-----+-----+-----+
|| +-----+ +-----+ | | Password | | |||
+-----+ +-----+ | | | +-----+-----+-----+
| | | | +-----+-----+-----+
+-----+ +-----+ | | Mountlist | | ||^|||
|| Start | | | +-----+-----+-----+
+-----+ +-----+-----+-----+
+-----+-----+-----+
| | | |
+-----+-----+-----+

```



```

|+-----+ +-----+
|      +-----+ +-----+
|Device |           | |   Ok   |   | Cancel | |
|      +-----+ +-----+
+-----+
|                                           | / |
+-----+

```

Description:

A) Devices listview.

List of accessible device drivers. If you click on a device, the geometry is NOT showed.

B) Unit slider.

With this slider, you can select the unit number.

C) Info button.

Many device drivers don't support the trackdisk.device command TD_GETGEOMETRY, so crashing your system because this command is not recognized. If you press this button, you can force the geometry reading with some risk. Thus in device mode, the automatic geometry reading is disabled. (see the section~6.1).

Note:

If you activates the IEC~~mode, it's not possible to open the device requester. Into GuiX main window, the "Device" string gadget and the button to open this requester are sobstiuted by a slider that allow to select the IEC device unit (from 8 to 11).

1.35 GuiX Menu: Start

Short key: RAMIGA + S

Activates xFX and has the same function of Start button that is avaible in main window.

1.36 GuiX Menu: Delete Files

Short key: nessuno

Allows to erase one or more files using the reqtools.library file requester.

1.37 GuiX Menu: Save Prefs

Short key: RAMIGA + P

Opens the GuiX preferences window:

```

+--+-----+--+
|*| GuiX Preferences      |%|||
+--+-----+--+
|+-----+|
||          +-----+--+||
|| xFX Path |PROGDIR:xFX   ||^||| <- A
||          +-----+--+||
||          +-----+--+||
|| GZip Path |C:GZip       ||^||| <- B
||          +-----+--+||
|+-----+|
|+-----+|
||+-----+ +-----+||
C -> |||      Save      | |      Cancel      ||| <- D
||+-----+ +-----+||
|+-----+|
+--+-----+--+
|                               | / |
+--+-----+--+

```

Description:

A) Path to find xFX during GuiX startup.

If GuiX don't found the xFX executables because the path is incorrect, this window is autamoatically opened.

B) Path to find GZip.

C) Button to save preferences.

D) Button to close the window.

Menù items:

Preferences

Close <--- Closes the window (short key: RAMIGA + C).

1.38 GuiX Menu: About MUI

Short Key: RAMIGA + M

Shows the MUI copyright message.

1.39 GuiX Menu: About GuiX

Short key: RAMIGA + ?

Shows the GuiX copyright message.

1.40 GuiX Menu: Iconify

Short key: RAMIGA + I

This menu item iconifies GuiX.

1.41 GuiX Menu: Quit

Short key: RAMIGA + Q

Closes GuiX and xFX.

1.42 GuiX Menu: Read

Short key: RAMIGA + R

Displays the register page with READ options.

1.43 GuiX Menu: Write

Short key: RAMIGA + W

Displays the register page with WRITE options.

1.44 GuiX Menu: Copy

Short key: RAMIGA + C

Displays the register page with COPY options.

1.45 GuiX Menu: Check

Short key: RAMIGA + K

Displays the register page with CHECK options.

1.46 GuiX Menu: MUI Prefs

Short key: RAMIGA + A

Activates the standard program to configure MUI.

1.47 GuiX Menu: Help

Short key: RAMIGA + H

Activates help in AmigaGuide® format.

1.48 5.5.1 GuiX: Checksum calculator

Short key: RAMIGA + 1

This is a tool for multiformat calculation of file-disk checksum. The supported method are five:

```
Brick CRC-32  <-- It's the GNU command Brick
BSD           <-- Without polinomial table
MD5           <-- 64 bit with polinomial table
Posix         <-- With polinomial table
System V      <-- Without polinomial table
```

```
+--+-----+--+
|*| CheckSum Calculator  |%||
+--+-----+--+
|+-----+|
||      +-----+--+ ||
|| File |          ||^| || <-- File name for calculation
||      +-----+--+ ||
||      +--+-----+ ||
||Method ||^|  Brick CRC-32  | || <-- Method
||      +--+-----+ ||
|+-----+|
|+----- Checksum -----+|
|| Dec -          || <-- Decimal checksum
|| Hex -          || <-- Hexadecimal checksum
|+-----+|
|+-----+|
||+-----+      +-----+ ||
|||  Calc  |      | Cancel  | || <-- Control buttons:
||+-----+      +-----+ ||      Calc  = calculates the checksum
|+-----+|              Cancel = closes the window
+-----+|
|                      | / |
+-----+--+
```

Menu items:

```
Checksum
  Calculate    <-- Activates the calculation (short key RAMIGA + C)
  -----
  Close        <-- Closes the window (short key RAMIGA + Q)
```

1.49 5.5.2 GuiX: Diz Toolkit

Short Key: RAMIGA + 2

DizToolkit is a little utility to add, extract and remove comments (diz) from/to XPK packed file-disks. At the end of file is added a text file with special string markers.

```
+---- File name
|
|
|  +-+-----+-----+-----+-----+
|  |*| Diz Toolkit                               |%|||
|  +-+-----+-----+-----+-----+
|  |+-----File-Disk-----+ +-----+|
|  ||+-----+-----+-----+ +-----+||
+-> |||Ram:MyDisk.xfx          ||^|| || New    ||| <-- Add~a~comment
|  ||+-----+-----+-----+ +-----+||
|  |+-----+-----+-----+ +-----+||
|  |+-----FILE-ID.diz-----+ || Extract ||| <-- Extract the comment
|  ||+-----+-----+-----+ +-----+||
|  |||FileType:                |*|| |+-----+||
|  |||[NUKE] File-Disk         | || || Remove  ||| <-- Remove the comment
|  |||Name:                    | || |+-----+||
+-> |||My Disk                  +-+| |         ||
|  |||Author:                  |=|| |         ||
|  |||Alessandro Pedretti     +-+| |+-----+||
|  |||Generated by:           |=|| || Close   ||| <-- Closes the window
|  ||+-----+-----+-----+ +-----+||
|  |+-----+-----+-----+ +-----+|
|  +-----+-----+-----+-----+
|  |                               | / |
|  +-----+-----+-----+-----+
|
|
+---- Comment description
```

Menù items:

```
Diz Toolkit
  Close    <--- Closes the window (short key RAMIGA + C).
```

1.50 5.5.4.1 GuiX: New Diz

This window allows to insert all comment items:

```

+--+-----+--+
|*| New Diz                                |%|||
+--+-----+--+
|+-----+|
||                                     +-----+||
||           Name |                    ||| <- File name
||           +-----+||
||           +-----+||
||           Author |                    ||| <- Author name
||           +-----+||
||           +-----+||
|| Device to expand |                    ||^||| <- Device for unpacking
||           +-----+||
||           +-----+||
||           Date |                    ||| <- Creation date
||           +-----+||
||           +-----+||
||           Comment |                    ||| <- Comment
||           +-----+||
|+-----+|
|+-----+|
||+-----+ +-----+||           Ok      = Confirm the commento
|||           Ok      | |           Cancel    ||| <-
||+-----+ +-----+||           Cancel = Reject the input
|+-----+|
+--+-----+--+
|                                     |/|
+--+-----+--+

```

Menù items:

New Diz

Close <--- Closes the window (short key RAMIGA + C).

1.51 5.5.3 GuiX: File Splitter

Short key: RAMIGA + 3

The File Splitter is a GuiX section that is able to split a file-disk into more than one file with prefixed size. In this way, you can split very big files into more than one disks. File splitter is also able to execute the join operation to recreate the original file-disk, using a set of files with smaller size. You can also perform this operation using the AmigaDOS command JOIN. All files generated from the original file-disk, have a name composed of a prefix specified by user and of numerical suffix (extension), like: PREFIX_NAME.001 The maximum number of splitted files is 999.

Mode:

```

+--- Split = creates a set of files from a file-disk
|   Join  = joins all splitted files to create the original file-disk
|
| +-----+-----+-----+-----+
| |*| File Splitter                                     |%|||
| +-----+-----+-----+-----+
| |-----+-----+-----+-----+ File-Disk -----+|
| ||      +-----+-----+-----+-----+-----+ ||      File name to split
| || Name |                                     ||^| || <--- or name of any file
| ||      +-----+-----+-----+-----+-----+ ||      to join
| |-----+-----+-----+-----+-----+|
| |-----+-----+-----+-----+-----+|
| ||      +---+-----+-----+-----+-----+-----+ ||
+>|| Mode ||^| Split | Prefix |               ||^| || <--- File prefix
    ||      +---+-----+-----+-----+-----+-----+ ||
    ||      +-----+-----+-----+-----+-----+ ||
+>|| Size |               | ||^| Custom          | || <--- Presetted sizes
| ||      +-----+-----+-----+-----+-----+ ||
| |-----+-----+-----+-----+-----+|
| |-----+-----+-----+-----+-----+|
| || +-----+-----+-----+-----+-----+ ||
| || |      Execute      |      |      Cancel      | || <--- Closes the window
| || +-----+-----+-----+-----+-----+ ||
| |-----+-----+-----+-----+-----+|
| +-----+-----+-----+-----+-----+|
| |                                     | / |
| +-----+-----+-----+-----+-----+|
|
|           ^
|           |
|   Activates the process
|
|                                     Size (bytes) of files
+-----+-----+-----+-----+-----+ to create

```

Menu items:

Splitter

Execute <--- Activates the Join/Split process (short key RAMIGA + E)

Chiudi <--- Closes the window (short key RAMIGA + C)

Notes:

The presetted file sizes are:

| Botton | Bytes | Floppy |
|---------|---------|--------------------------------|
| 360 Kb | 364032 | 5.25" DD, 3.5" SD Ms-Dos |
| 720 Kb | 730112 | 3.5" DD Ms-Dos |
| 880 Kb | 899072 | 3.5" DD Amiga FFS |
| 1.44 Mb | 1457664 | 3.5" HD Ms-Dos |
| 1.76 Mb | 1800192 | 3.5" HD Amiga FFS |
| 2.88 Mb | 2913280 | 3.5" ED Ms-Dos |
| 3.5 Mb | 3602432 | 3.5" ED Amiga (Super XL drive) |

```
=====
```

1.52 5.5.4 GuiX: UnPacker

Short key: RAMIGA + 4

UnPacker is a tool that is able to depack all file-disks created with xFX. It use the XPK library set and the unpacked file is overwritten on the original. The supported formats are: XPK, PowerPacker and GZip.

```
+--+-----+--+
|*| File-Disk UnPacker      |%||
+--+-----+--+
|+----- File-Disk -----+|
||      +-----+--+||
|| Nome |                ||^|| <--- File-disk name to depack
||      +-----+--+||
|+-----+|
|+----- Progress -----+|
||+-----+|
||          0 %          || <--- Progress bar
||+-----+-----+-----+|
||0%          50%          100%|
|+-----+|
|+-----+|
||+-----+ +-----+||
+> |||   UnPack   | |   Cancel   ||| <--- Closes the window
|  ||+-----+ +-----+|
|  |+-----+|
|  +-----+|
|  |                | / |
|  +-----+|
|
|
+----- Activates the unpacking procedure
```

Menu items:

UnPacker

Close <--- Closes the window (short key RAMIGA + C)