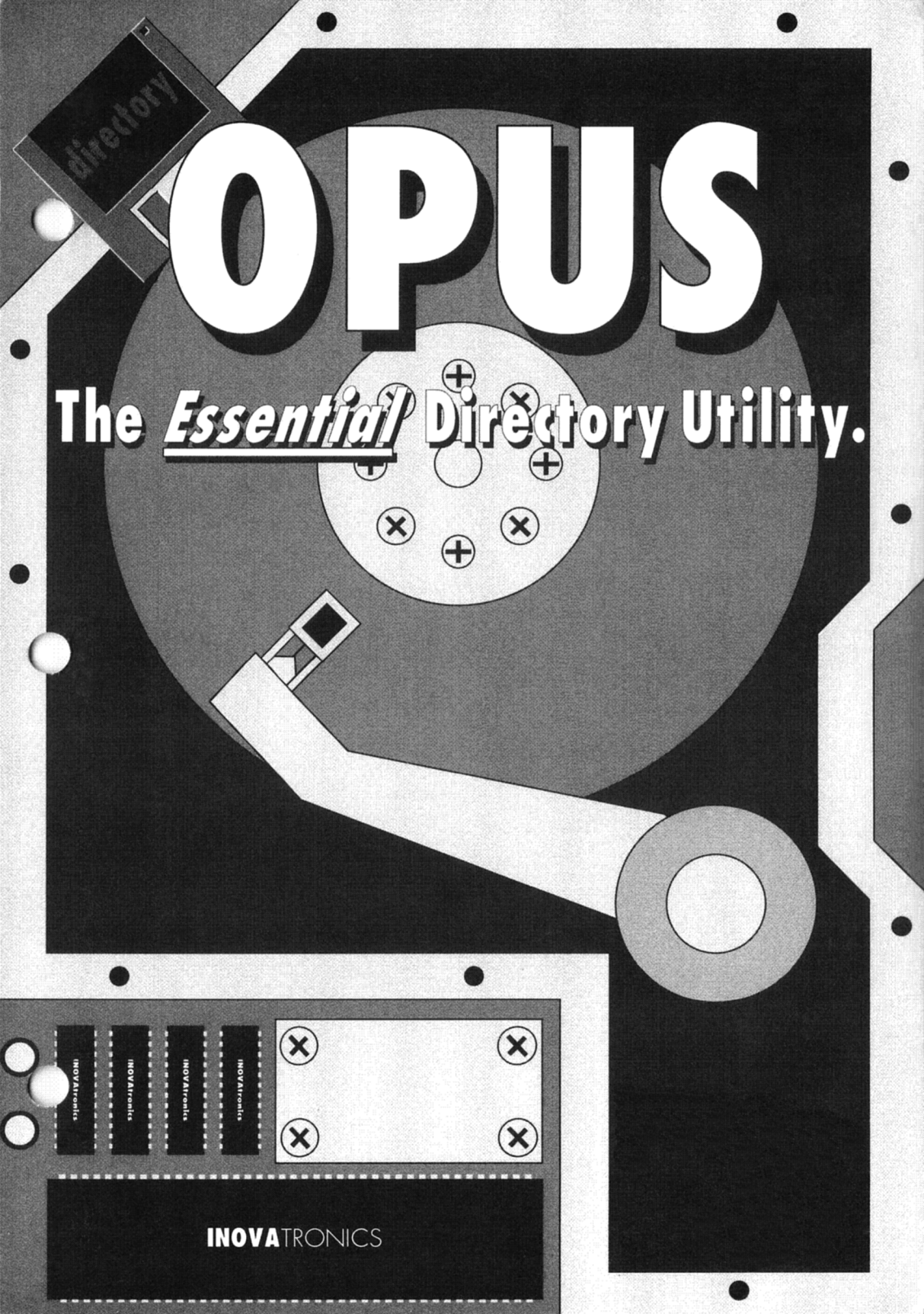
directory
OPUS
The Essential Directory Utility.
INOVATRONICS

Directory Opus Legalities

Directory Opus Manual Version 4.0 Contributers
Carol Bland
Eddie Churchill
Cash Foley
Joel Guerra
Jonathan Potter

Project Manager
Eddie Churchill

Manual Editor
Cash Foley

Cover Design
Scott Martindale

Directory Opus Manual version 4.0 - Produced with Radical Eye's AmigaTEX with special thanks to Tomas Rokicki.

For help, call (214)340-4992 between the hours of 10:00AM - 3:00PM CST.

Inovatronics, Inc.

Suite 209B	**In Europe, contact:**
8499 Greenville Avenue	Inovatronics GmbH
Dallas, Tx 75231-2499 USA	Im Heidkamp 11
Office Phone: 214/340-4991	W-5000 Cologne 91
Tech Support: 214/340-4992	Germany
FAX: 214/340-8514	Phone: +49-221-875126
BBS: 214/342-1930	FAX: +49-221-8704747

Contents

A Note from the Author

I have been working on Directory Opus for the past three years. What started as a simple directory utility (however revolutionary for its times) has grown into what I believe to be the most configurable, functional and easy-to-use directory utility available for the Amiga and possibly for any other computer.

Much of this would not have been possible without the help of many people. To list all the people who have contributed in any way would be impossible, but here are some of them:

> Mel Bice, John Cousins, Peter Jackson, Adrian Jones, Edward Lawford, Michael Lorant, Jay Miner, Ross Molden, Michael Portmann, Ian Rodbourn, Ian Steuart, Tim Strachan, Andrew Wilson

Nor could Directory Opus 4 have been possible without the valuable and much appreciated efforts of INOVAtronics, who are largely responsible for the user-interface improvements and hundreds, if not thousands, of other changes. To Eddie Churchill and everyone else at INOVAtronics, to almost all of the beta testers, and to everyone else who has contributed, thank you.

Enjoy the program!

Jonathan Potter

Chapter 1

A Directory Opus Walk-Through

1.1 What Is Directory Opus?

The theory behind Directory Opus is quite simple. On the screen there are two windows. These are called **Directory Windows**. You can read a directory into each of these from any device accessible by the Amiga. You can select files and/or directories in either of these windows, then manipulate them almost any way you like. Selected **entries** can be copied to the opposite window, deleted or renamed; text files can be read, picture files can be viewed, and sound files can be heard. Directory Opus offers much more than these "bare-bones" features, and you will learn more later in this manual.

1.2 Files and Directories

The Amiga's **DOS** (Disk Operating System) deals with two kinds of data-arrangement: **files** and **directories**.

Each file and directory must be given a unique name; within a directory you cannot have two files, two directories, or a file and a directory with the same name.

1.2.1 Files

Any data you record on a disk is stored in a file. Files contain information, which may be from a database, from a word processor, from a painting program, or the entire contents of a program.

The size of a file is expressed in **bytes**, each byte being equivalent to one character. Storing the string "Hello" in a file would use five bytes, as the word "Hello" is five characters long.

Whether a file can be displayed, executed, deleted, edited, or considered as a **script file** depends upon attributes that are explained on page 55.

All files have a **datestamp** which shows what the system date and time were the last time the file was written to.

Files may also have a comment of up to 79 characters attached to them.

1.2.2 Directories

To store information in a logical manner, disks are generally organized into **directories**, which are often referred to as **drawers**. If you picture a disk as a filing cabinet, with your programs, database files and pictures as the actual files, then directories are the drawers of the filing cabinet. Some of these drawers have further drawers inside them, called **sub-directories**, which themselves can contain drawers, and so on, indefinitely.

The directory or sub-directory containing any given sub-directory is known as its **parent directory**.

The highest level of organization is the **Root Directory**. If the directory is a filing cabinet, then the root directory is the room it stands in.

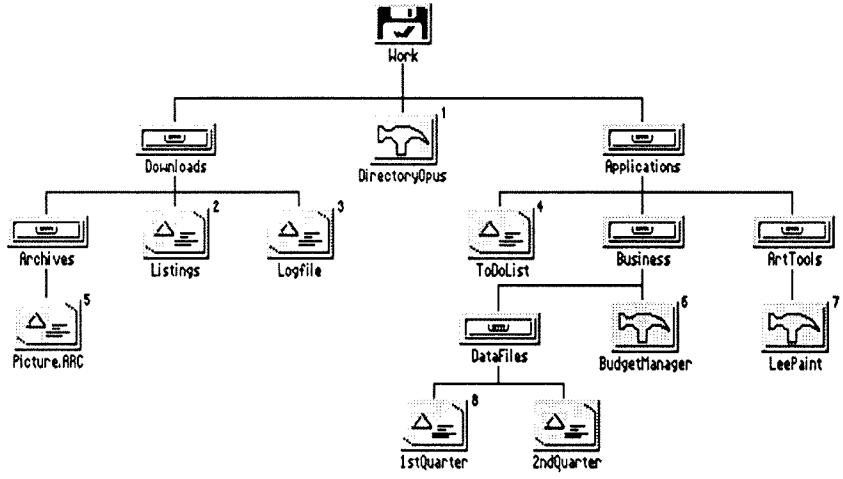

Figure 1.1: Directory Tree Example

The Directory Tree Example (Figure 1.1) demonstrates a sample disk structure.

The route you take along a **directory tree** to reach a file is called a **path**. As you proceed along the path, each branch of the tree is separated from the next branch by a / character. For example, the path of Archives is expressed as `Work:Downloads/Archives`.

To refer to Picture.ARC, you use what is called a **pathname**. This consists of a file's path, followed by the name of the file. Picture.ARC's pathname would be `Work:Downloads/Archives/Picture.ARC`.

The number of files and sub-directories any given directory can contain is limited only by the amount of space on the disk.

The list below shows the correct path to each of the files in the chart.

1. Work:DirectoryOpus
2. Work:Downloads/Listings
3. Work:Downloads/Logfile
4. Work:Applications/ToDoList
5. Work:Downloads/Archives/Picture.ARC
6. Work:Applications/Business/BudgetManager
7. Work:Applications/LeePaint
8. Work:Applications/DataFiles/1stQuarter

1.3 Installing Directory Opus

If your computer does not have a hard drive, you may want to run Directory Opus from the distribution disk; otherwise you need to use

the installation program, InstallOpus, which inserts Directory Opus into your hard disk's root directory. It will modify your **User Startup Sequence**. InstallOpus allows you to choose the approach best suited to your level of expertise.

For the novice, InstallOpus inserts Directory Opus into your hard disk's root directory and does everything for you, offering you few choices. You may choose to have the Directory Opus screen displayed when you first boot up your computer, otherwise you may run Directory Opus from the Workbench or from the CLI.

At a more ambitious level, InstallOpus allows you to select the directories into which Directory Opus is to be installed.

The expert user can take control: InstallOpus allows you to define every single choice available, and assumes that you know what you are doing. You are able to modify your startup sequence to your needs.

For more information, use InstallOpus' built-in help system.

1.4 Running Directory Opus

There are three common ways of running Directory Opus.

- From Workbench
- From a CLI
- From your startup-sequence

Each are described below.

1.4.1 Starting From Workbench

The easiest way to run Directory Opus is from the Workbench. Double-click on the Directory Opus icon on the Workbench screen and, several seconds later, the Directory Opus screen will appear.

1.4.2 Starting From a CLI

To run Directory Opus from the CLI enter:

```
DOpus:DirectoryOpus
```

This assumes one of two things. Either you are running Directory Opus from a floppy drive containing the Directory Opus diskette, or you have it installed on a hard drive with an assignment of DOpus: to its location. (The InstallOpus utility can create the assignment in your startup-sequence).

1.4.3 Automatic Startup in the startup-sequence

The startup-sequence is a script file that is automatically executed when you first turn on the computer. It can contain instructions to do many different things, including running Directory Opus.

Either the InstallOpus utility can modify your startup-sequence for you, or you can do it yourself. To modify your startup-sequence you need a text editor such as Commodore's Ed or MEmax. A highly recommended commercial text editor is CygnusEd Professional, by ASDG-Inc. This is the text editor with which Directory Opus was written, and is a very powerful tool.

Run your text editor, and load your startup-sequence into it. The

startup-sequence resides in the S: directory of your boot disk. Pick a convenient place to put the new instruction. A good place would be the line immediately before the `EndCLI` instruction (at the end), but you can really put it anywhere. Add the line:

```
Assign DOpus: DH0:Tools/DOpus
DOpus:DirectoryOpus -i
```

This assumes you have installed DirectoryOpus in a directory called `DH0:Tools/DOpus`. Of course you need to use the actual directory path.

The `-i` at the end of the instruction tells Directory Opus to start up iconified, which means that the program will open only a little window on the Workbench screen, instead of its own custom screen, and will sit unobtrusively in the background until needed. From the Workbench, you can add a ToolType of `ICONSTART=1` to the Directory Opus icon, for the same effect. If you do not want Directory Opus to start up iconified, simply remove the `-i`.

The more technically inclined may notice that there is no `RUN` on the instruction line. This is because Directory Opus detaches itself, freeing up the CLI and allowing it to continue with the startup-sequence (and ultimately closing).

Make sure you save the startup-sequence once you have modified it. If your changes make sense, Directory Opus will be run automatically when you boot up.

There is another flag you can use when you run Directory Opus. The `-c` flag tells Directory Opus where to load the configuration file from. For instance, `-cDH1:stuff/directoryopus.cfg` would load (and subsequently save) the configuration file from DH1:stuff/directoryopus.cfg. From Workbench, a ToolType of `CONFIGFILE=DH1:stuff/directoryopus.cfg` would have the same effect. By default, it will look in DOpus:Modules.

1.5 Introduction to Using Directory Opus

This section gives you a short tour of Directory Opus. It describes Directory Opus in its default configuration. The various configuration options will be described in a later chapter.

1.5.1 Aborting

Before using the program, it is important to know how to abort an action. Most functions can be aborted once they have begun, and this can be useful if you accidentally start deleting the contents of your hard drive.

To abort, press the left and the right mouse buttons simultaneously. This will only take effect if the Directory Opus screen is **active** (i.e., it is at the front of the display).

Do not be alarmed if the action does not abort immediately; some functions such as **copy**, will have to finish the file they are working upon before aborting.

1.5.2 Directory Opus Components

Let's start with a brief description of the main components of the Directory Opus Display.

Title Bar The Title Bar is at the very top of the Directory Opus display. It is used to display various status or error messages. If you are unsure what Directory Opus is doing, it is a good idea to look in the title bar for a hint. The title bar is also used to reposition the Directory Opus screen.

Clock/Memory Bar The Clock/Memory Bar is at the bottom of the Directory Opus display. It displays current memory and time

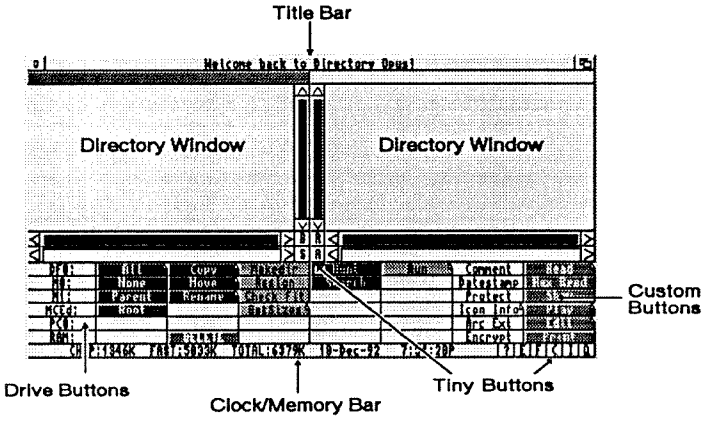

Figure 1.2:

information.

Directory Windows A directory window allows you to display the files and sub-directories contained in a directory. You can select (and deselect) files and sub-directories. Directory Opus has two directory windows. The Directory Windows are described in greater detail below.

Custom Buttons The Custom buttons perform actions which usually trigger an operation on the selected files in the active directory window.

Drive Buttons The Drive buttons read the contents of a specific directory and display them in the currently active directory window.

Custom Menus Custom menus, like Custom buttons, perform actions.

NOTE: Many parts of the DirectoryOpus Interface use the Right Mouse button for alternative operations. As with DPaint, the

mouse pointer must be positioned on the title bar when the Right Mouse button is clicked for the Menu Titles to appear.

Tiny Buttons Four Tiny buttons (a figure is shown later) are positioned between the two directory windows, and six Tiny buttons are placed on the Clock/Memory Bar.

1.5.3 Directory Windows:

There are two Directory windows. One window is active at a time. Because some actions, such as **copy**, work with selected files in the active Directory Window and use the inactive Directory Window as the destination, the active window is referred to as the Source Window and the inactive window is referred to as the Destination Window.

Selecting the Active Directory Window

Clicking anywhere in a window with either mouse button will activate that window. Unless you are extremely precise in your placement of the pointer, however, clicking with the left mouse button to activate a window may lead to your selecting an unwanted entry. This is when using the right mouse button can be useful, as selecting entries is a left mouse button function, but clicking with the right mouse button can never have that effect.

Pressing the space bar activates the inactive directory window. This allows you to toggle between the active and inactive windows.

Figure 1.3 shows the main components of the Directory Window.

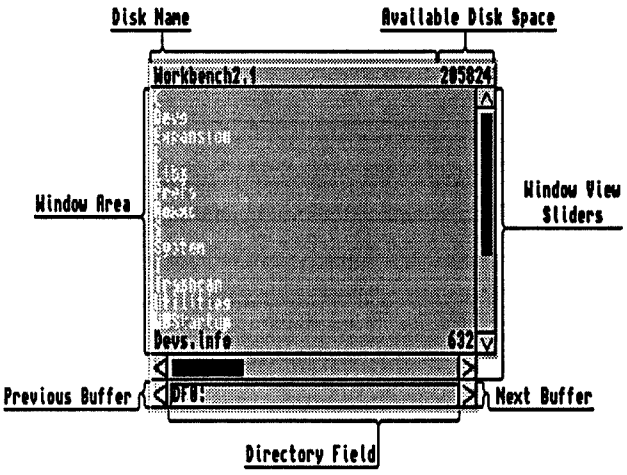

Figure 1.3: Directory Window

Directory Window components:

- Directory Field
- Disk Name
- Window Area
- Window View Sliders
- Next and Previous Buffer buttons

Directory Field

Many DirectoryOpus functions require that you read a directory into the active Directory Window. To read any directory, enter the name into the Directory Field.

When you click in a directory field it will be activated, and a cursor will appear. Pressing the Return key will activate the Directory Field

for the Active Directory Window. Once the Field is activated, enter the name of the directory you want.

There are several editing features you can use:

Backspace	Deletes the character to the left of the cursor
Delete	Deletes the character under the cursor
Cursor key	Moves the cursor one space in the direction of the arrow
Cursor key + Shift	Moves the cursor to the beginning or end of the string
Right Amiga + x key	This is known as **Amiga-X** Erases the entire string
Right Amiga + q key	This is known as **Amiga-X** Restores the contents of a field to the previous information.

Once you have entered the pathname you wish to read, press return. You will see the directory window activate (if it is not already active), and the directory will read.

Disk Name

The Disk Name area of the Directory Window shows the Disk or Volume Name for the currently displayed Directory. This area also displays the amount of free space on the directory's volume.

Window Area

The Window Area of the Directory Window displays the directory's files and sub-directories.

Selecting Files

To select an entry, file or sub-directory, click on it with the left mouse button. If you hold down the left mouse button and move the mouse up or down, all entries passed over will be highlighted. This is called **drag-selecting**. Figure 1.4 demonstrates selected files and directories.

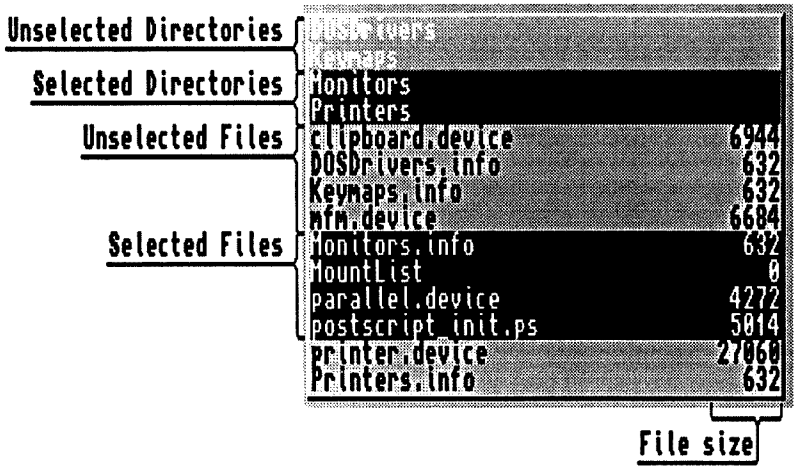

Figure 1.4: Directory Window with Selected Files and SubDirectories

Figure 1.5 demonstrates statistics which appear in the Title Bar when you are selecting files.

When you click on an entry which is already selected, it will deselect it. By drag-selecting when you click on a selected entry, already-selected entries that are passed over will be deselected.

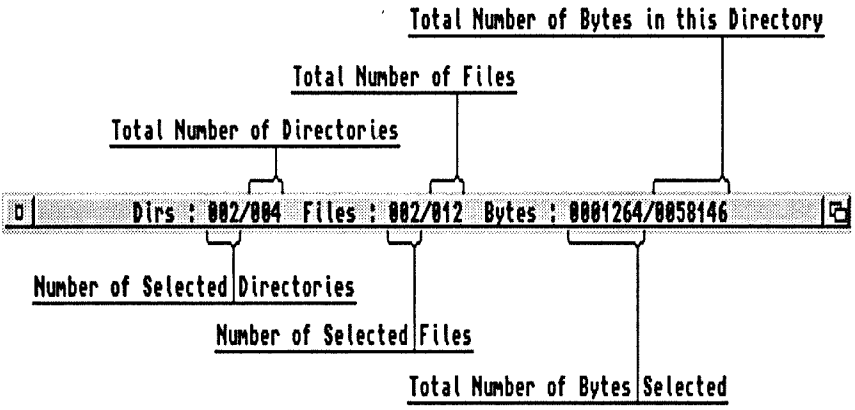

Figure 1.5: Status Bar when selecting Files

Entering Sub-Directories

Many directories that you read in will contain sub-directories. Instead of having to type in the name of a sub-directory, you need only to double-click on it. This will load the sub-directory into the **same** Directory Window.

Opening the sub-directory in the other directory window allows you to copy files between the sub-directory and its parent. To do this, you perform a Click Move Click (Click-M-Click) with the left mouse button. Click-M-Click means to click first on a sub-directory entry then to move the mouse pointer to the other (inactive) window and click again. This can be tricky because the two clicks must occur within the Amiga Configured Double-Click Time (see Amiga Input Preferences).

An alternative method to Click-M-Click (which requires less mouse pointer gymnastics), is to click and hold on an unselected sub-directory entry with the left mouse button. Then, without releasing the left mouse button, press and release the right mouse button. Without

releasing the left mouse button, move the mouse pointer to the inactive window. As you move the mouse pointer, a copy of the selected sub-directory will follow the mouse. When you position the mouse pointer over the inactive window, release the mouse button; the sub-directory will now be read into the inactive window, which now will become the active Directory Window.

A final alternative (which requires less mouse button gymnastics), is to click and hold on an unselected sub-directory entry with the left mouse button. Then, without releasing the left mouse button, move the mouse pointer to the unselected Directory Window. The trick to this method is to move the mouse pointer horizontally without moving up or down (this is for the sure handed mouse gymnast). Should you inadvertently move the mouse pointer over an entry above or below the selected sub-directory, it will not follow the mouse pointer to the other window. When you do it correctly, a copy of the selected sub-directory will follow the mouse pointer over to the inactive window and will be read when you release the mouse pointer.

Moving to a Parent Directory

If the directory displayed in a Directory Window is a sub-directory, you may wish to return to its Parent Directory. You could return to the Parent Directory by editing the Directory Field text directly, but the procedure is cumbersome and error-prone; there are better ways.

- The default configuration includes a Custom Button which returns you to the Parent Directory.
- Each Directory Window contains a hidden Parent button on the outer edge of the window area which will move you to the Parent Directory. On the left Directory Window, it is the white border on the far left. On the right Directory window, it is the black border on the far right. When you click the left mouse button on

a hidden Parent button, Directory Opus will display the directory's parent, if it has one. When you click the right mouse button on the hidden Parent button, Directory Opus will display the Root directory.

Window View Sliders

Once the directory has been read in, it is displayed in the directory window. There is limited space on the screen, however, so the whole directory may not be visible at once. A pair of Sliders extends your view of the window. In the center of the two directory windows are two vertical Window View Sliders. You can slide these up and down, to scroll through the contents of the directory.

- To slide the vertical window view sliders, press the left mouse button on the black bar and, while holding down the mouse button, move the mouse up and down.
- Click above or below the solid bar for the slider to move one page in that direction.
- At the bottom of the two vertical Window View Sliders are arrow buttons. These will move the slider one line in either direction when clicked upon.
- Click and hold the left mouse button on one of the arrow buttons for the directory to scroll through the window.
- Press either the up or down cursor keys to move one line in that direction. These too will scroll when held down.
- Press the up or down cursor keys in conjunction with either SHIFT key to move a page in that direction.

- Press a cursor key in conjunction with the CTRL key to display the top or bottom of the directory.
- To scroll quickly to the first entry starting with a particular letter, press that letter; the directory will scroll as close as it can get to the first entry - whether file or directory - starting with that letter.
- Press a letter in conjunction with either SHIFT key to scroll only to the first file beginning with that letter.

Whether or not the entire directory fits in vertically, it will definitely not fit in horizontally. There is more information available for entries than just the **filename**. Also displayed is the size in bytes, the protection bits assigned to that entry, the date that entry was last written to the disk, and any comment that may be assigned to that entry.

Horizontal Window View Sliders can be seen at the bottom of the directory windows, with arrows at the inner ends. These work like the Vertical Window View Sliders, including the keyboard equivalents. You can scroll up and down, and left and right, by holding the right mouse button and moving the mouse.

The Directory Windows are usually the same size. If you want to see more information than is displayed in the windows you can stretch the divider between them left or right to resize them.

There is a hidden size bar between the two vertical Window View Sliders. When you click and hold this button, the size bar will appear. While holding the mouse button, move the mouse left or right. When you have the window as you want it, release the mouse button.

There are three shortcut keys for resizing the Directory Windows:

- - Slides the Window View Sliders to the far left
- \ Slides them to the far right.

- = Re-centers the Window View Sliders

Next and Previous Buffer buttons

Directory Opus maintains up to 100 directory **buffers** for each directory window.

- To move back and forth through the buffers, click on either the Previous Buffer button or the Next Buffer button. The Buffer will be displayed without re-reading the drive. The keyboard equivalent alternative is to press the ALT key in conjunction with the left or right cursor key.
- To activate the directory field, press the Return key.
- To re-read the directory, press the Return key twice.

1.5.4 Custom Buttons

There are default Custom buttons, all of which have left mouse button functions. Some of the buttons are also configured to perform Right Mouse button functions.

The Custom buttons with right mouse button functions are indicated by a "dog-ear" in the upper right corner of the button. The All Custom button (Figure 1.6) demonstrates the dog-ear.

IMPORTANT:

Many of the Custom buttons perform actions on the selected files in the current directory window. Before clicking on a Custom button, verify that the intended directory window is active. Novice Directory

Opus users can be confused when they have clicked on a Custom button in an inactive window, and it does nothing. Of course you should verify that the selected files and directories within the directory window are the ones you wish acted on.

Figure 1.6: Default Custom Buttons

All

The **All** button selects all entries in the currently active directory window.

Toggle

Selecting **All** with the right mouse button will **Toggle** or reverse the state of all entries. This causes selected entries to be deselected and deselected entries to be selected.

None

The **None** button deselects all entries in the currently active directory window.

Parent

The **Parent** button reads the parent directory of the directory open in the active directory window. If the parent directory is contained in the window's **Buffer List**, it will be displayed without re-reading it.

When the current directory is an **assignment**, for example C:, the assignment will be expanded, and then the parent function will move to the parent directory of the assigned directory.

Root

The **Root** button reads the root directory of the active directory window. As with the Parent function, the buffer list will be searched before the parent directory is re-read.

When the current directory is a sub-directory to an assigned directory, the Root function will display the assigned directory. When the current directory is an assigned directory, the Root function will display the root drive of the assigned directory.

Copy

The **Copy** button copies the selected file to the destination (inactive) directory. If any directories are selected to copy, the sub-directory Filter will be used to determine which files will be copied (the sub-directory Filter is described on page 43).

CopyAs

The **CopyAs** function is similar to **Copy** except that the entries are copied to the destination directory using new names that you specify. **Wildcards** can be used here; see the Wildcard discussion in the Appendices.

Move

This function will move all selected entries from the source directory to the destination directory. The entry will no longer exist in its original place. If any directories are selected to move, the sub-directory Filter will be used to determine which files will be moved (the sub-directory Filter is described on page 43).

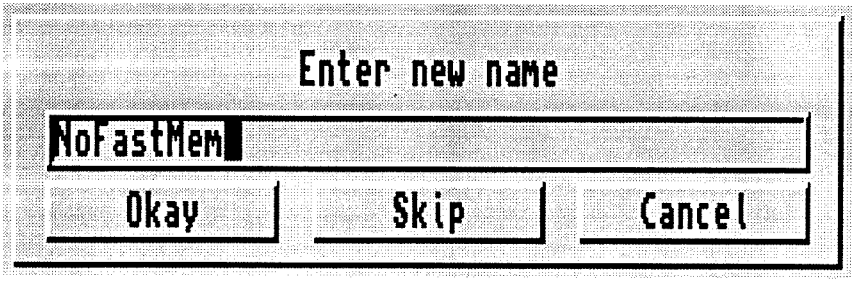

Figure 1.7: CopyAs Requester

Be careful with this function. Directory Opus will delete the file if you are moving it to a different *device*.

MoveAs

MoveAs allows you to give each entry a new name before it is moved.

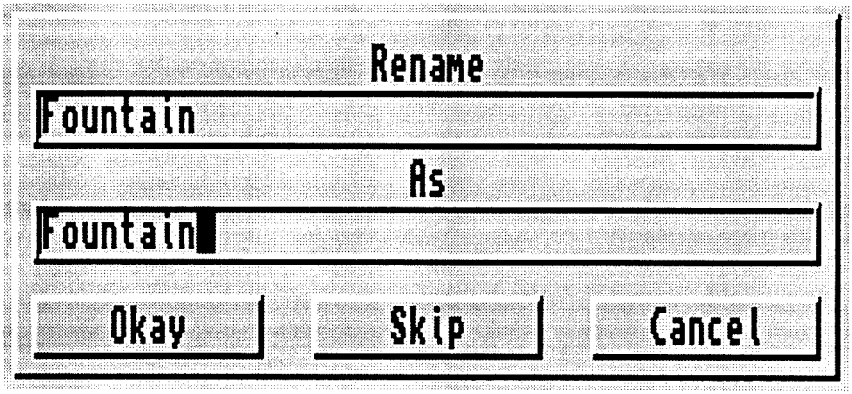

Figure 1.8: Initial Rename Requester

Figure 1.9: Subsequent Rename Requester

Rename

Rename allows you to give new names to all selected entries in the active directory window. A requester will appear for each entry in turn, asking for the new name. The initial rename requester has two string fields instead of one. You will usually just edit the name in the lower of the two to the new name.

A limited type of **wildcard** rename is possible. Entering a * in the bottom field allows you to add prefixes or suffixes. For instance, entering *.pic will add a .pic suffix to all selected entries. Entering A* will add an A prefix. Only one * may be used in this process.

You cannot give a file a name that contains a *.

If you enter a * in the top as well as the bottom field, you can replace sections of the name. For example, entering *.pic in the top Field and *.iff in the bottom Field will replace the .pic suffix in any entry that has one with a .iff suffix. If an entry does not have a .pic suffix, it will be left untouched. The * may also be embedded. For instance, renaming FOO*BAZ as GEE*WIZ would rename FOOBARBAZ as GEEBARWIZ. Again, only one * may be used in each of the string Fields.

Clone

Clone allows you to make a copy of selected entries in the same

directory, but with different names. A requester will appear for each entry, asking for the new name.

Delete

Delete will delete all selected entries in the active directory window.

Be careful with this, as it is easy to wipe out valuable data if you are careless. Directory Opus's default configuration brings up a requester before deleting. Nevertheless, you should always double check the selected files before clicking this button.

Makedir

Makedir allows you to create a new sub-directory in the active directory window; its name is limited to 30 characters. A configuration option, Create Icons With Directories, can be used to create an icon at the same time. The name of the directory is limited to 25 characters if Create Icons With Directories is enabled.

Assign

Assign allows you to create an AmigaDOS assignment to the directory in the active window. The Assign Command gives more examples. (see page 182)

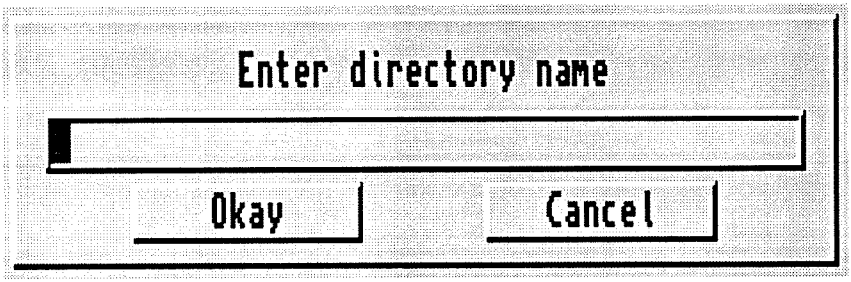

Figure 1.10: MakeDir Requester

Check Fit

The **Check Fit** button tests whether the selected file will fit on the destination drive. The Screen title bar will display the number of bytes needed, the available space, and the percentage of the file which will fit on the destination drive.

GetSizes

The **GetSizes** button causes any selected sub-directories to be scanned. The scanning process calculates the total size, in bytes, of all files contained in the sub-directory. Once scanned, the sub-directory will be displayed.

If a sub-directory has been previously scanned, it will not be re-scanned when you select the GetSizes button. See **Clear Sizes** below.

The GetSizes button also displays in the status bar the number of files, directories and bytes that have been selected out of the total number of files, directories and bytes. If there is enough space

for the total selected bytes on the disk in the destination directory window, all selected files would fit on the destination disk if they were copied and a Y will be displayed after the count. If they won't fit, a N will be displayed.

If you select an operation which causes a sub-directory to be scanned (e.g., COPY, PROTECT, HUNT, etc.), the size will be displayed as though you had performed a **GetSizes**.

ClearSizes

The **ClearSizes** button will clear the sizes for selected directories, but not for selected files.

If a sub-directory has been scanned already, pressing the Get Sizes button will not work. It is possible that a previously scanned directory's size is no longer accurate; any application, including Directory Opus, might have changed the contents of the sub-directory since it was last scanned. To re-scan a sub-directory, the size must be cleared with the ClearSizes button.

Hunt

The **Hunt** button allows you to search all selected sub-directories for a specified file or files. A requester appears asking for the **pattern** to hunt for. You can use full **pattern matching** for this search.

If a file matching the pattern is found, you are asked if you wish to enter the directory containing it, or to continue the search. If you elect to enter the directory, the directory will be read and then all matching entries will be highlighted.

You can also select entire devices to hunt, using the DEVICE LIST feature that is described later.

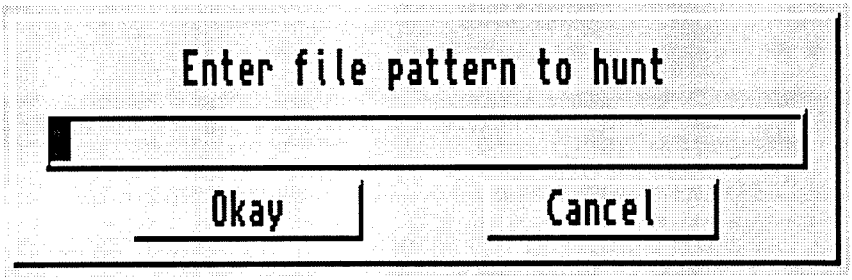

Figure 1.11: Find File Requester

Search

The **Search** button allows you to search the contents of all selected files, and the files within selected directories, for a specified string. A requester appears asking for the string pattern to hunt for. Again, full pattern matching is supported.

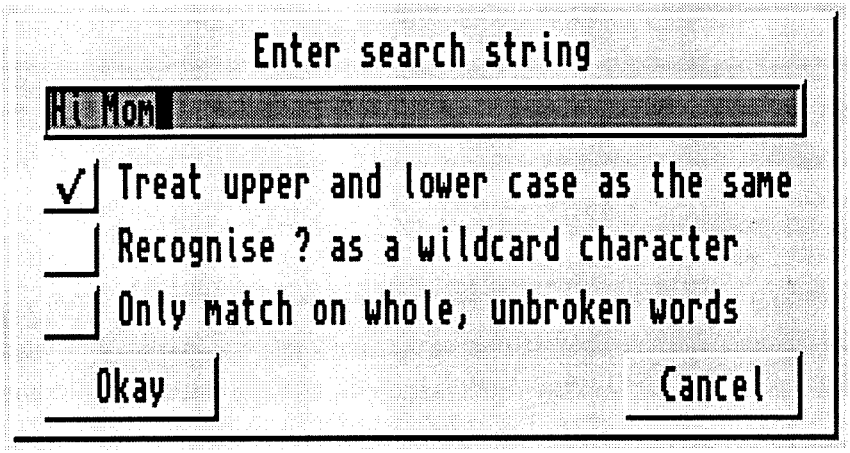

Figure 1.12: Search Requester

If a file containing the string is found, you are asked if you wish

to read that file, or continue the search. If you elect to read the file, it will be loaded into the text viewer, and a search will automatically be initiated for the string you want.

You can also select entire devices to search, using the DEVICE LIST feature that is described later.

If any directories are selected to search, the sub-directory Filter will be used to determine which files will be searched (the sub-directory Filter is described on page 43).

Run

The **Run** button allows you to run each selected file in turn, providing that file is executable. It is simmilar to double-clicking on the file's icon, or running it from the CLI a requester will appear, asking for any arguments (should you require any).

Execute

If you select this button with the right mouse button, the files will be "executed" as **batch files**. This has the same effect as running the batch file via IconX (refer to AmigaDOS manual), or executing it from the CLI.

Comment

The **Comment** button allows you to add **comments** to all selected entries, or to edit existing comments. The maximum length of a comment is 79 characters. If any directories are selected to comment, the sub-directory Filter will be used to determine which files will be commented (the sub-directory Filter is described on page 43).

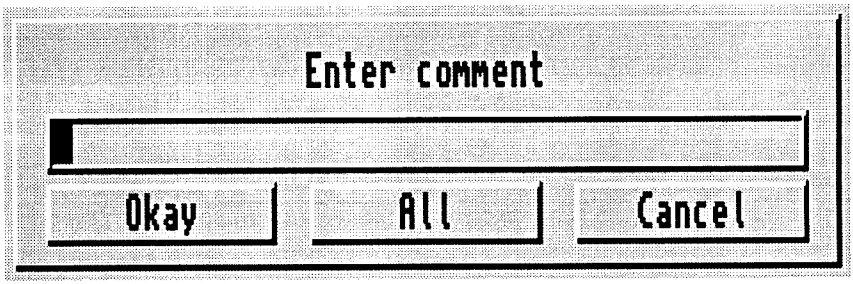

Figure 1.13: Comment Requester

You will be prompted for a comment for each selected entry. If you wish to attach the same comment to all entries, you should select the All button.

Datestamp

The **Datestamp** button allows you to change the datestamp of the selected files and directories in the active directory window. When you select directories, you are asked whether you wish the files within them to have their datestamps modified also.

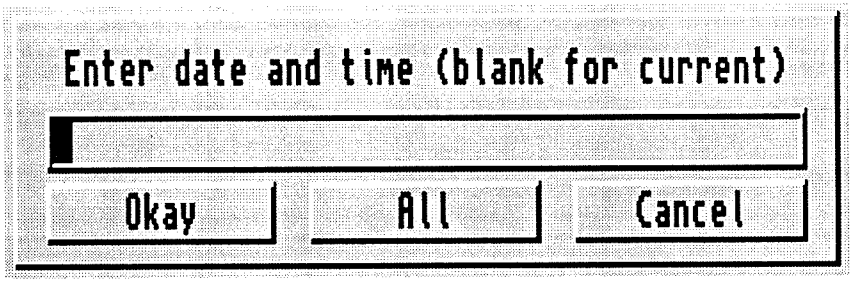

Figure 1.14: DateStamp Requester

If any directories are selected to datestamp, the sub-directory Filter will be used to determine which files will be datestamped (the sub-directory Filter is described on page 43).

For each entry, you are presented with a requester. If you wish the file to have its datestamp set to the current date and time, simply press return. Otherwise, enter the date and time you want.

To set the datestamp of all selected entries you should select the All button from the datestamp requester.

Choose Okay or press return to set the datestamps one file at a time.

Protect

The **Protect** button allows you to modify the **protection bits** of the selected files and directories in the active directory window. When you select directories, you are asked whether you also wish the files within them to be protected. For each entry, you are presented with a requester displaying the protection bits currently set for that entry. This is described in greater detail in the next chapter (see page 56).

If any directories are selected to protect, the sub-directory Filter will be used to determine which files will be protected (the sub-directory Filter is described on page 43).

Icon Info

The **Icon Info** button brings up the Directory Opus Info Requester for each of the selected files.

Add Icon

AddIcon allows you to add icons to all selected entries in the active directory window. Directory Opus will automatically

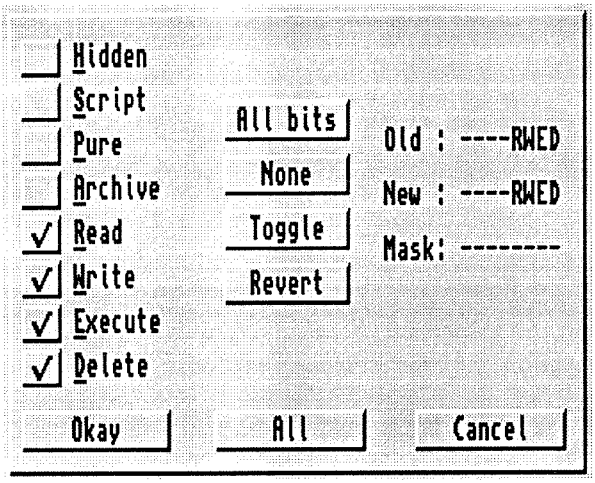

Figure 1.15: Protect Requester

sense what type of file it is and add the appropriate icon (drawer, tool or project). Default icons are used unless you have specified the name of your own icons within the configuration (described later).

Arc Ext

This button allows you to extract files from an *Archive File*. Directory Opus comes configured for ARC, ZOO, LHARC files. When you click on this button, the files are extracted from the selected files and copied into the destination window.

NOTE: Directory Opus' distribution disks do not include the actual archive programs ARC, ZOO, and LHA. Because they are Shareware, we are not allowed to distribute them. They are available free on most bulletin boards.

Encrypt

Have you ever had files that you wanted to encrypt so that only people who knew the password could understand them? This function allows you to do just that. It will encrypt all selected files, using the password that you enter, with a complex algorithm that most people will find impossible to work out. The resulting files are not written over the originals, but are instead written to the destination directory. They will be the same size as the original files, so you can ensure you have enough room in the destination directory.

To decrypt a previously encrypted file, you should enter the same password preceded by a minus sign. For example, to decrypt files you encrypted with the password FOO, select the files, choose the encrypt function and enter -FOO as the password.

Read

The **Read** button allows you to read selected files. The name of the file is displayed at the bottom of the screen. You can use the slider and cursor keys to move through the file. Buttons along the bottom move up and down by a page, move to the top and bottom, allow you to search for text, and print the file. Pressing the right mouse button or the Q key quits the the text viewer. For more information, see the Text Viewer on page 72.

ANSI Read

The **ANSI Read** button brings up the same Reader as **Read** except that it handles the special ANSI control sequences.

Hex Read

The **Hex Read** button will read the selected files in the same way as READ, except in hexadecimal format. This allows you to view binary files and other files containing non-text characters.

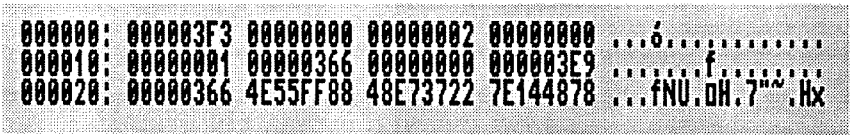

Figure 1.16: Example Hex Reader Format

Figure 1.16 shows an example of the Hex Viewers output. The first value is the offset, displayed in hex. This is the number of bytes you are into the file. The next four values are each a four-byte longword, with the actual ASCII representation at the end. Any non-text characters are shown as a . character.

Show

The **Show** button is very versatile indeed. It will display IFF ILBM pictures and brushes, animations, Workbench icons and fonts.

If the file is an IFF ILBM, it will be displayed using information from the file. Directory Opus will show most pictures and brushes, including overscan, extra halfbrite (EHB), HAM (4096 color) pictures, Dynamic HiRes pictures created by MacroPaint (from Lake Forest Logic), and AGA 8 bit pictures[1].

[1]Only when using OS3.0 and an AGA compatible Amiga

The following keys can be used when viewing a picture or animation.

Tab	- Cycle Colors
Cursor Keys	- Scroll
Mouse Pointer	- Scroll
Shift Cursor Keys	- Scroll by Page
Ctrl Cursor Keys	- Scroll to End
.	- Mouse Pointer On/Off
[	- display in 4 bit format
]	- display in 8 bit format
Esc or Right Mouse Button	- Abort
Q or Left Mouse Button	- Next
Space, Help or P	- Help and Print Requester

These keys can be used when viewing an Animation.

S	- Starts and Stops
N	- Next Frame
-	- Slow Down
=	- Speed Up
\	- Original Speed
F1 - F10	- Various Speeds (F1=Fastest)

The show routine will also display icons. If the file has a .info suffix it is assumed to be an icon. If the icon has an alternative image, press return to toggle between the two.

To display a font, you need to enter the actual font drawer, and select the Size file (that is, the file 8, 12, 19, etc.), NOT the .font file.

Play

The **Play** button plays the selected files as sound files. If the file is an 8SVX format IFF sampled sound (of the type generated by AudioMaster, for instance), the speed and other information from within the file will be used. This button will play any file, though, and if the file is not 8SVX, it will assume a speed of 10000 samples/second.

Using the inovamusic.library, the Play button also plays Star/Sound/Noise/ProTracker, Med, OctaMed, Octalizer, and Med with MIDI modules.

Each file is played once only. To abort a sound while it is playing, press the left and right mouse buttons.

Loop Play

If you select this button with the right mouse button, the sounds are played continuously (looped). To advance to the next sound, you must press the left mouse button.

Edit

The **Edit** button allows you to edit any selected files, using the text editor (or word processor) specified in the configuration.

The Custom button configuration is explained later. By default, it is configured to use Commodore's `c:ED` program. You will probably want to alter this configuration to use an editor of your own choosing.

New File

The **New File** button prompts you for a file name.

If you select this button with the right mouse button, you will be asked for a filename to create a new file.

Print

The **Print** button brings up a requester which allows you full control of Print formatting. For more information, see the Print File Requester on page 67.

1.6 The Tiny Buttons

The Directory Opus screen has several buttons that are not configurable; they activate the most necessary and useful functions, and are included so that these functions are always available, whatever your configurations may be. If you had mistakenly configured Directory Opus so that no button or menu called the configuration program, you would never be able to run the configuration program to change this!

These buttons are called the **Tiny Buttons**, as they are much smaller than the configurable buttons on the screen. Four of them are arranged in a square approximately at the center of the screen, six others are found in the bottom right-hand corner of the screen. They all have one-character titles.

B - Buffer list

This activates the bufferlist function. This function displays a list of all the directories contained in Directory Opus' internal buffers. You may then double-click on one of the displayed buffers so as to jump to that buffer immediately, rather than clicking the arrows to cycle through the buffers one by one.

Pressing the right mouse button over the Buffer List button will activate the **Clear Buffers** command which clears from memory

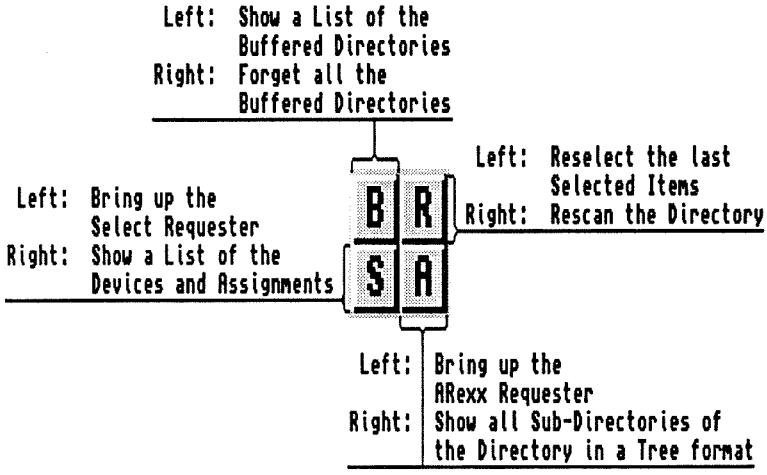

Figure 1.17: The Tiny Center Buttons

the contents of all buffers except the two that are currently displayed.

R - Reselect

This activates the reselect function. This function will reselect all entries that were selected before the last operation was initiated. The entries are reselected only if they still exist, and regardless of whether the buffer containing them is currently displayed.

Pressing the right mouse button over the **R - Reselect** button performs the Rescan function on the active directory window.

S - Select

This button activates the select function. This allows you to select files and directories in the active directory window using **wildcards**. All standard wildcards are supported (*,,? etc..) as well as the AmigaDOS wildcards #? and ?.

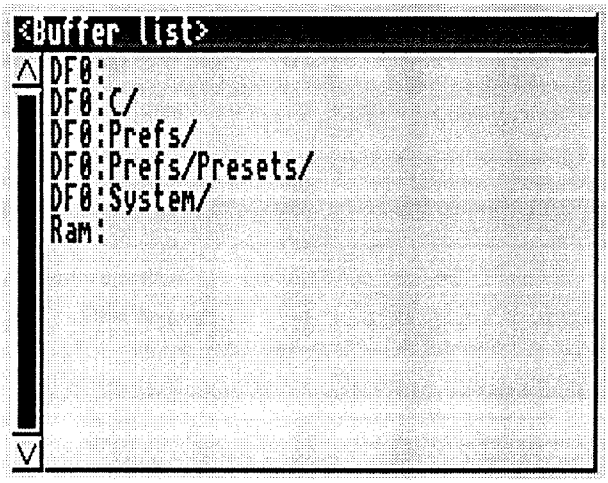

Figure 1.18: Buffer List

Pressing the right mouse button over the File Select Requester Button will activate the Get Devices function (the DEVICE LIST function). This function produces a list of all devices, volumes and logically assigned directories that exist in the system.

A - ARexx

This activates the ARexx function, which allows you to launch

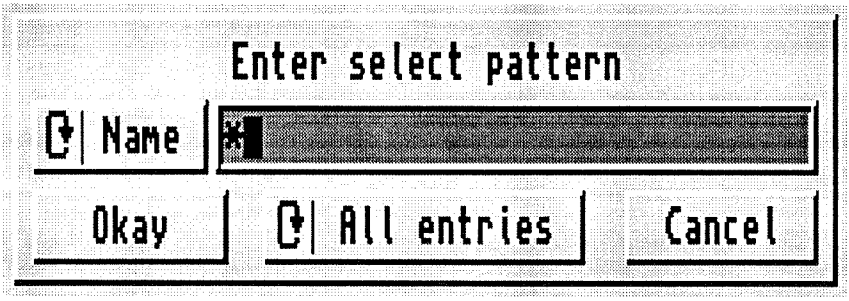

Figure 1.19: File Select Requester

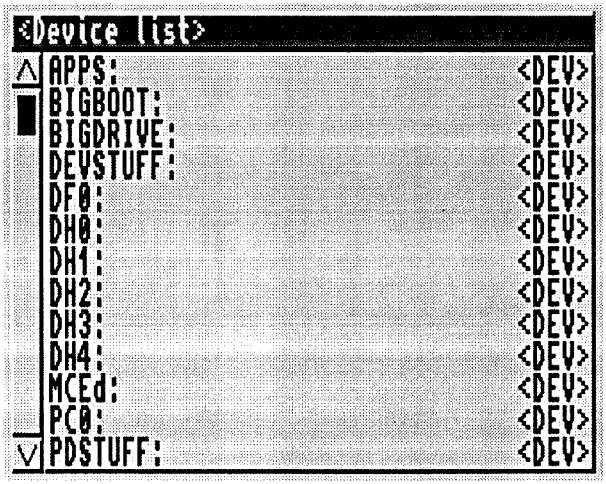

Figure 1.20: Device List

an ARexx script or execute an ARexx command from within Directory Opus.

Pressing the right mouse button over this button will activate the dirtree function (the Directory Tree function). This function scans all sub-directories in the currently displayed directory, and produces a tree structure.

1.7 The Tiny Bottom Right Buttons

These buttons are optional; if you don't want them you can make them disappear by de-selecting the Bottom-Right Tiny buttons flag on the Config/Screen/General configuration screen.

? - Help

This button activates the help function. Provided a help file has been loaded, selecting any button or menu after enabling the help

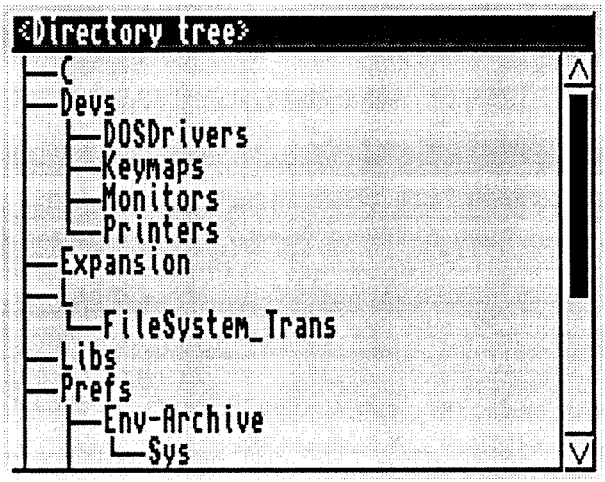

Figure 1.21: Directory Tree

mode should result in the appearance of some helpful text.

Note: The highlighted question mark indicates the help mode is active.

E - Error help

This activates the errorhelp function, which gives you more

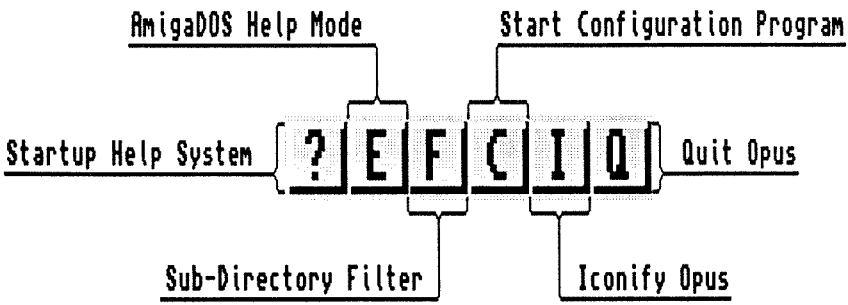

Figure 1.22: The Tiny Bottom Right Buttons

Figure 1.23: Help Example

information about DOS error codes than is generally available.

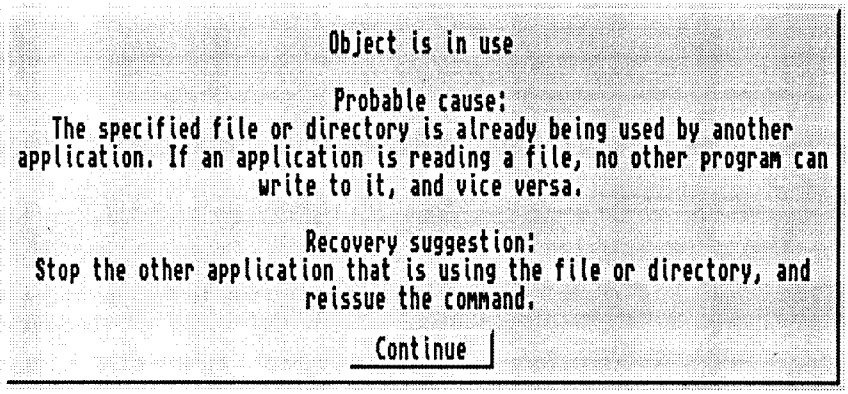

Figure 1.24: Error Help Example

F - Sub-Directory Filter

This Button activates the sub-directory Filter. When you click on this button, a requester will ask you to enter a file filter. The filter can contain a file specification containing wildcard characters. When you click Okay or press return, the requester will disappear and the F button will be highlighted. This indicates the filter is activated.

When the filter is activated, any command which processes the files in selected sub-directories will only act on files which match the filter. These commands are:

- **Copy**
- **CopyAs**
- **Move**
- **MoveAs**
- **Hunt**
- **Search**
- **Protect**
- **DateStamp**
- **Comment**

If the F button is highlighted, you can turn off the filter by clicking on it again. It will become un-highlighted.

C - Configure

This activates the configure function, and will either load into memory or invoke the configuration program.

I - Iconify

This button activates the iconify function. It closes the Directory Opus screen, de-allocates as much memory as possible, and then opens a small window on the Workbench screen.

This allows you to have Directory Opus constantly available, while using the minimum amount of memory possible.

Q - Quit

This activates the quit function, which allows you to exit Directory Opus (providing there is not a print operation under way). If the configuration has been modified and not saved, you are asked if you wish to save it first.

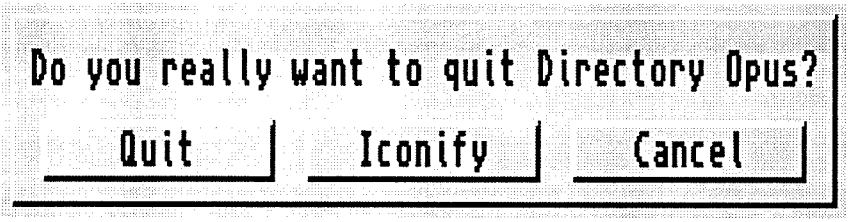

Figure 1.25: Quit Requester

1.8 Default Menus

Directory Opus has 100 user-definable menus. The **Functions** chapter explains more about the menus, and the **Function Editor** chapter explains how to configure them. This chapter simply gives a brief description of the **Default Menus**. All the menus can be changed, but they are configured initially for certain functions.

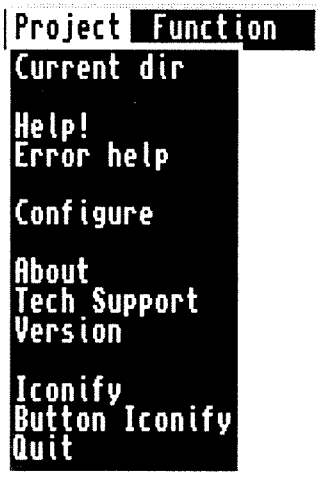

Figure 1.26: Main Screen Project Menu

1.8.1 Project Menus

Project/Set date

This function allows you to set the **system time** and **date**, especially if you do not have a battery-backed clock. After you have set the time, the Amiga clock keeps running to provide an accurate timekeeper. All files or directories created will have a datestamp of the current system time. The **DateStamp** command of Directory Opus uses the system time.

Two requesters will appear; the first containing the current date, and the second containing the current time. To change either the date or the time, delete and enter a new one. The date is expected to be entered in the form DD-MMM-YY. For example, 22-Sep-89. Time is expected in the form HH:MM:SS, as in 18:30:43. You must include the dashes and colons as demonstrated above.

Project/Current dir

This function allows you to set the current working directory for Directory Opus. The **current directory** is the directory loaded when you activate either directory field, delete the existing pathname (if there is one), and press return on a blank string.

A requester will appear, containing the pathname of the active directory window. If this is acceptable as the new current directory, press return. Otherwise, edit it to read as you wish.

Project/Help!

Provided a help file has been loaded in, selecting any button/menu after enabling help mode with this menu should result in the

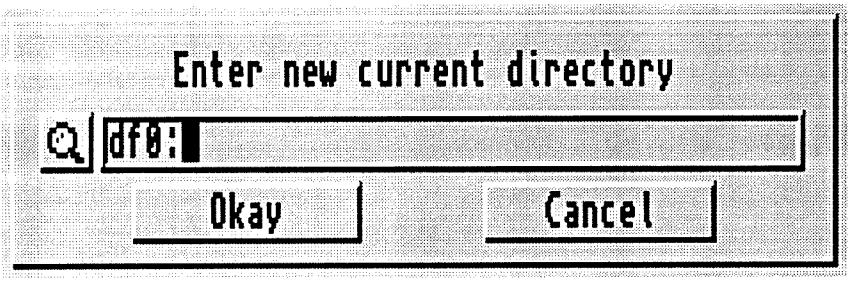

Figure 1.27: Change Current Dir Requester

appearance of some helpful text.

Help can be read from any file, although the default is S:DirectoryOpus.HLP. The file name can be changed with the Helpfile configuration option, described later.

The Help File Chapter describes the help file in detail, and explains how you can edit it to add help for your own buttons and menus.

Project/Error help

Error help gives you information about DOS error codes. When you select this option, you are asked for which DOS error code number you want help. Examples of DOS error codes are 123, 205 and 225. You are then presented with a description of that error, and information pertaining to the possible cause and cure of the error.

Error help is also stored in the help file (see above); if the help file has not been loaded, error help will not be available. See the help file chapter for more information.

Project/Configure

Directory Opus uses a separate program to modify the

configuration. This saves about 127K of memory, as the configuration program need only be in memory when it is being used.

If you do have plenty of memory, Directory Opus can be configured to load the configuration program on start-up. In this case, it is run from memory each time the configure function is invoked, rather than loaded from disk. This can be much faster, especially if you have a floppy-based system.

See the configuration program chapter for more detailed information.

Project/About

This function displays some information about the program.

Project/Tech Support

This menu displays information about getting Technical Support. Please refer to the requester before contacting INOVAtronics as there are specific times and methods for getting Technical Support.

Project/Version

This menu displays the current version of Kickstart, Workbench and Directory Opus. This information will be necessary when getting Technical Support.

Project/Iconify

This function closes the Directory Opus window and screen, de-allocates as much memory as possible, and then opens a small window on the Workbench screen. This is known as iconifying, and

allows you to have Directory Opus constantly available, while using the least amount of memory.

The iconfied window may contain a clock, or may not be visible at all, depending on how you have set the Iconify type configuration option (described later).

To determine the initial position of the iconified window (especially if you plan to start up Directory Opus in the iconified state), you should iconify Directory Opus, and position the window as you would like it to appear. That done, re-enter Directory Opus, and save the configuration (described later).

When iconified, you can drop Workbench Icons on the Directory Opus window. This will have the same effect as if you displayed the file in a Directory Opus window and double-click on it. For Example, if you drop a picture on the window, it will show it.

To re-enter Directory Opus, click the left mouse button in the iconified window, then press the right mouse button. If you wish to quit Directory Opus without going back into it, click the close button at the far left of the iconified window.

Note that if you have the configuration option Iconify Type set to Hidden, the only way to re-enter Directory Opus is with the hotkeys. These default to CTRL-SHIFT-ALT (the CTRL key, the left SHIFT key and the left ALT key simultaneously), although this may be changed in the configuration (explained later).

Project/Button Iconify

This causes a special variation of Directory Opus iconification. Normally iconification is set up by the ConfigOpus/System/Clocks screen, but with this command you can iconify Directory Opus into a bank of buttons on Workbench. In this mode Directory Opus allows

you to do almost all, if not all it normally does. For instance, if you select the Copy button a requester will open asking you to select something to copy followed by another asking where to copy it. You can also use the bank of buttons as an AppWindow when running in OS2.0. To show a picture, you would simply drag the icon of the picture onto the Show button.

The slider to the right of the button bank allows you choose which button bank is shown.

To uniconify back to Directory Opus, press the Uniconify button above the slider to the right of the bank, or select the Close button of the window. The close button will ask you if you wish to quit Directory Opus or uniconify it.

Project/Quit

This function will exit Directory Opus (providing there is not a print operation happening at the time). If the configuration has been modified and not saved, you are asked if you want to save it first.

1.8.2 Function Menu

The Function Menu contains miscellaneous disk operations, and Archive adding.

Function/Disk copy

This function brings up a requester which allows you to make an exact copy of one disk on another. For more information, see the DiskCopy Requester on page 59.

Figure 1.28: Main Screen Function Menu

Function/Format

This allows you to format a new disk. All new disks need to be formatted before the computer can write files to them.

For more information, see the Format Requester on page 61.

When this is selected, a requester with several buttons appears. Drives DF0: through DF3: are available. Separate names may be specified for each disk.

Verify allows you to turn Verify on or off. As with **Function/Disk Copy**, the process is faster with Verify turned off, but you won't be made aware of any errors, so it's better to leave Verify turned on if you don't completely trust your disks.

If Quick is selected, the disk will just be initialized (wiped). This provides an extremely fast way to erase an old disk. This will not work on new disks, however; only on disks that have previously been formatted.

If No Icons is selected, the Disk.info, Trashcan.info and Trashcan directory will not be created on the disk.

Function/Install

This function brings up a requester that allows you to make a disk bootable. To be bootable, disks must have special information written on their first sector. For more information, see the Install Requester on page 63.

Function/Relabel

This function allows you to change the name (label) of the disk that is open in the active directory window.

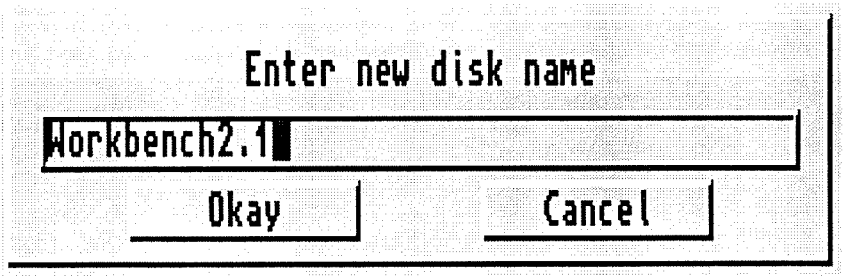

Figure 1.29: Relabel Requester

Function/Print dir

This function allows you to print the directory in the active directory window. You are able to choose the information you want to print (file sizes, protection bits, dates and comments). This function is not started as a separate process, unlike the print command. For more information, see the Print Dir Requester on page 65.

Function/Disk info

This function displays some information about the disk the active directory resides on, including space used, space free, datestamp and number of errors on the disk.

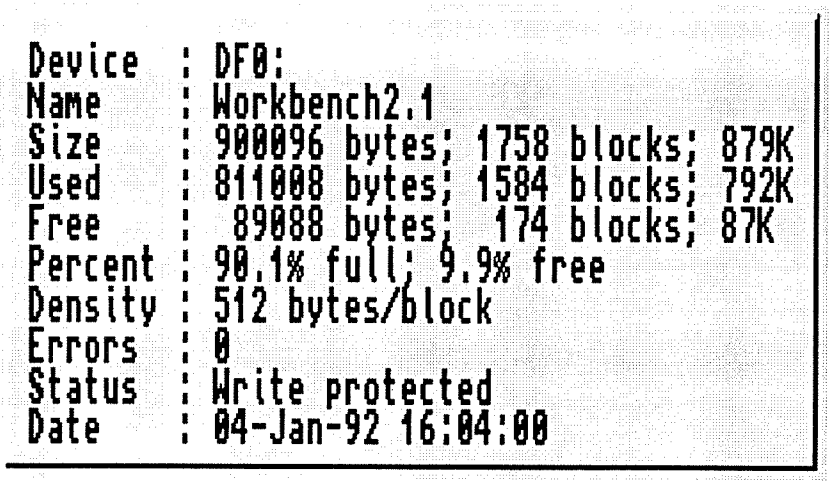

Figure 1.30: Disk Info Requester

Function/LHA Add, Arc Add, Zoo Add

These functions allows you to create and add files to archives, using the popular archivers ARC, ZOO and LHA or LHARC (and up to three others; see the Functions section of chapter 7 for more information).

To add files to an archive, select the files you wish to add, and select this function. You will be asked for the name of the archive to create or add to. You should have the destination directory open in the inactive window.

Chapter 2

Requesters & Viewers

This chapter presents the more complicated requesters and viewers. Each of these is associated with Directory Opus Commands which are activated by Custom Buttons and Menus.

2.1 Protection Requester

The Protection Requester (see Figure 2.1) allows you to change a file's or a sub-directory's protection bits.

The protection bits are a group of flags that are stored with the file, that determine the characteristics of the file. These flags are given one character names. The protection bits currently in use are `HSPARWED`.

H — *Hidden.* If this flag is set, the file is not normally displayed. This allows you to mark certain files as "invisible", to avoid cluttering your directories. The file can still be accessed normally, and not all programs implement this flag.

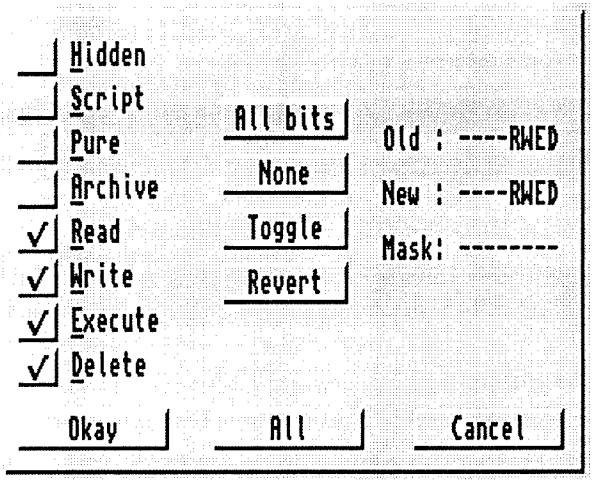

Figure 2.1: Protection Requester

S — *Script.* A script file is a file containing a list of AmigaDOS commands to execute; it is like a simple computer program. This flag indicates that the file in question is a script file. A script file is sometimes called a batch file.

P — *Pure.* If a program file is flagged as pure, it can be made to remain in memory, even when not in use. This can save a great deal of time, especially if the program is used a lot, as it does not have to be loaded from disk each time.

A — *Archive.* This flag indicates that the file has not been changed. If this file is ever written to, the A flag will be turned off. This can be used in a hard disk backup program, to record which files have been backed-up, and need not be backed-up again.

R — *Readable.* If this flag is set, the file can be accessed.

W — *Writeable.* If this flag is set, the file can be written to (ie, more information can be stored in it than is already there).

E — *Executeable.* If a program file does not have this flag set, it can not be run.

D — *Deleteable.* If this flag is not set, the file can not be deleted.

While AmigaDOS and other programs do not fully support all of these bits, DirectoryOpus gives you access to all documented protection bits. As AmigaDOS is enhanced, some bits, such as Writable, will become more useful. If you have additional questions about the usage of the bits, refer to your Amiga Documention.

There is a button for each protection bit along the left edge of the requester. Clicking on each of these will toggle a checkmark on and off within the button. When the checkmark is on, it means that the bit will be set when you click the **Okay** button.

In the middle of the requester are four buttons which manipulate multiple protection bits at a time. These buttons do not cause any action to take place within the selected files; they simply change the state of the protection bit buttons.

All bits

Sets all protection bit buttons.

None

Causes all protection bit buttons to be cleared.

Toggle

Sets all clear protection bit buttons, and clears all set protection bit buttons.

Revert

Restores the protection bit buttons to the settings they had when you entered this requester.

On the right side of the requester are three groups of protection bits: Old, New, and Mask. Each group contains a protection bit letter or a '-' (dash) for each bit. The letter indicates that the associated protection bit is set, a dash indicates that it is clear.

The Old group indicates the original bits when you entered the requester. The New indicates the current status of the protection bit buttons. Whenever you click the protection bit button, it will be reflected in this group.

Unlike the previously described Old and New groups, the Mask group bits can be changed by clicking on the associated letter or dash. The Mask group allows you to specify protection bits which should not be modified, regardless of the protection bit buttons. When the Letter is shown, the bit will not be modified. The dash means that you need to set the bit according to the setting of the protection bit button. This is most useful when changing files within a sub-directory or using the **All** button (described below). For example: you want to set the Archive bit in all files contained in a sub-directory, but you don't want to affect any of the other protection bits. Set the Archive Protection button on the left side of the requester and set all the Archive bits in the Mask group except A. This will preserve all the file's bits with the exception of the Archive bit, which will be set.

Along the bottom of the requester are three buttons:

Okay

Causes the current file (shown in the screen title bar) to be set as indicated in the protection bit buttons (and possibly filtered according to the Mask Group).

All

Causes **all** selected files to be set, without additional prompting, as indicated in the protection bit buttons (and possibly filtered according to the Mask Group).

Cancel

Aborts the Protection function.

2.2 Disk copy Requester

This function allows you to make an exact copy of one disk on another. When this function is invoked, a requester with several buttons appears.

From...

This list contains the possible disk drives that may be used as the source. When you click on one, it becomes the selected drive.

To...

This list contains the possible destination drives which are compatible with the selected source drive. The source disk drive is always available as a destination to allow you to make single drive copies. This only makes sense with removable media such as floppy drives as it would accomplish nothing to copy a hard drive to itself.

Verify

This button allows you to turn off the integrity verification when writing data to the destination drive. Although it is faster, you probably won't want to do this.

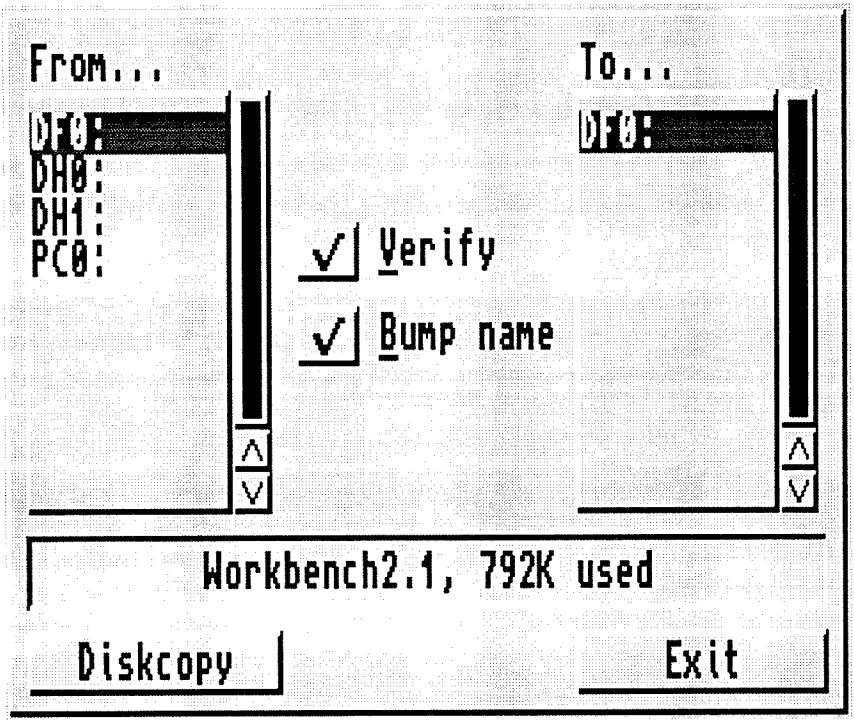

Figure 2.2: DiskCopy Requester

☑ Bump Name

This button allows you to change the volume name using the same naming convention as WorkBench's DiskCopy. (See AmigaDOS documentation for details.)

This function will not copy any protected software, or non-AmigaDOS format disks.

Selecting the **DiskCopy** button will start the copy. The **Exit** button will abort without attempting a DiskCopy.

2.3 Format Requester

This allows you to format a new disk. All new disks need to be formatted before the computer can write to them.

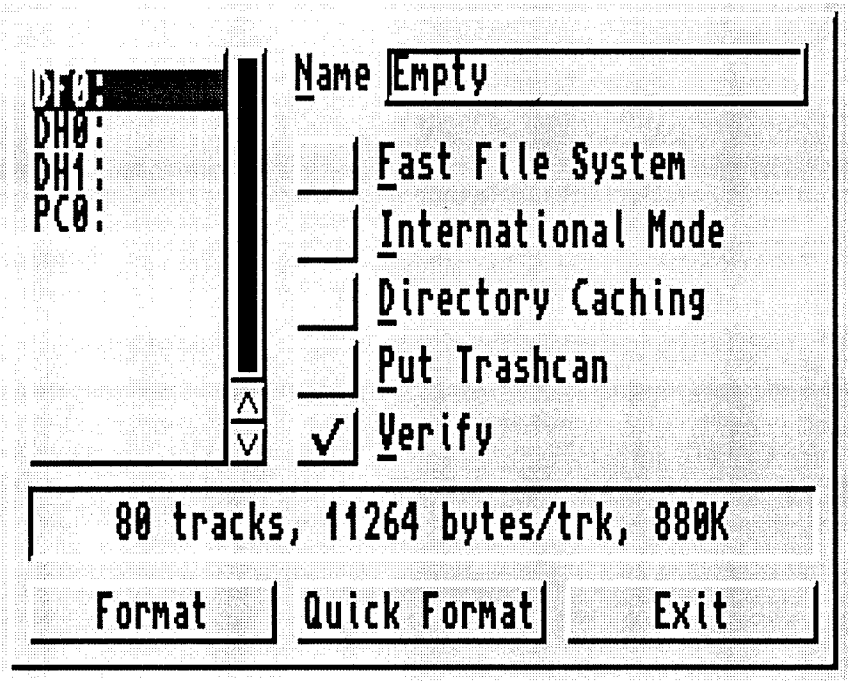

Figure 2.3: Format Requester

When this is selected, a requester with several buttons appears. On the left side is a list containing the devices which can be formatted using this operation.

Warning! This option will destroy existing data on a disk. Be sure you want to erase the data before you click **Format** or **Quick Format** buttons.

The list on the left side of the requester contains the devices which you can format. The selected device is highlighted. **Be sure the device you intend to format is the one that is highlighted!**

Name

This field allows you to give the drive to be formatted a volume name.

Fast File System

This allows you to format a device using the Fast File System option of AmigaDOS. You should consult AmigaDOS documentation for more detail. This option does not work on floppies prior to OS2.0.

International Mode[1]

This allows file and directory names to include accented characters.

Directory Caching[2]

Directory caching mode will decrease the capacity of your disk but the directory reading speed will be much greater.

Put Trashcan

This button allows you to put a trashcan in the root directory of the newly formatted device.

---

[1] available in OS2.1 and later
[2] available in OS3.0

☑ **Verify** This button allows you to disable the format verification. As with **Function/Disk Copy**, the process is faster with Verify turned off, but you won't be made aware of any errors, so it's better to leave Verify turned on unless you completely trust your disks (you really shouldn't).

Format

This button begins the formating process. Be very careful you have selected the correct device. Once a Format begins, it can be aborted, but data will be lost!

Quick Format

When this button is selected, the disk will just be initialized (wiped). This provides an extremely fast way to erase an old disk. This will not work on new disks, however; only on disks that have previously been formatted.

Exit

This button will abort without attempting a Format.

2.4 Install Requester

This function allows you to make a disk bootable. To be bootable, disks must have special information written on their first sector.

The list on the left contains the possible devices on which you can install a boot sector. Select the device you want to install on.

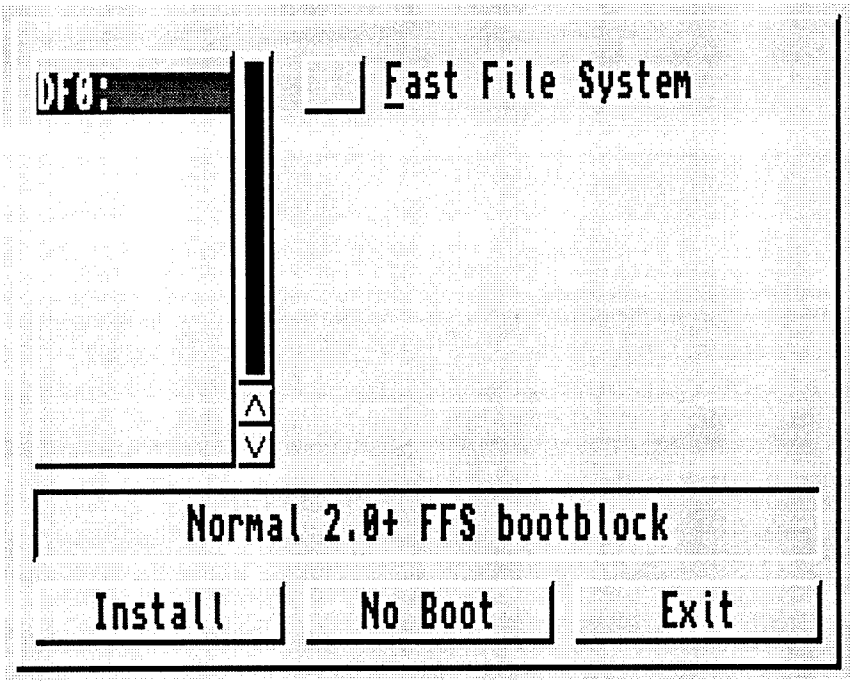

Figure 2.4: Install Requester

When you select a device (or insert a new disk in the device), Directory Opus will examine the boot sector and inform you of the disk's current status. It does this automatically when you first bring up the Install Requester. It will tell you one of the following:

No Disk Present This simply means that you haven't inserted a disk in the drive yet. When you do so, it will update this status message.

Normal 1.3 OFS bootblock This means the disk was intalled with the old file system used before version 2.0 of AmigaDOS.

Normal 2.0+ FFS bootblock This means that the disk was installed with the new file system used for 2.0 and later versions of AmigaDOS.

Non-Standard This means that the disk is corrupt. It may contain a virus and it would be a good idea to run a virus checker program to make sure.

☑ Fast File System

This button allows you to install a Fast File System bootblock.

Install

This button installs the boot sector on the Disk.

No Boot

This button removes the boot sector from the Disk. This will keep the disk from being bootable. However, you can always re-install the boot sector by clicking the **Install** button.

Exit

This button will abort without attempting an Install.

2.5 Print dir Requester

This function allows you to print the directory in the active directory window. If no files are selected, all files will be printed. However, if only some files are selected, only those files will be printed.

You can select the information you want to print: file sizes, protection bits, dates, comments, and file types.[3] Directory Opus can

[3]File types are discussed in the Configuration Chapter on Page 102

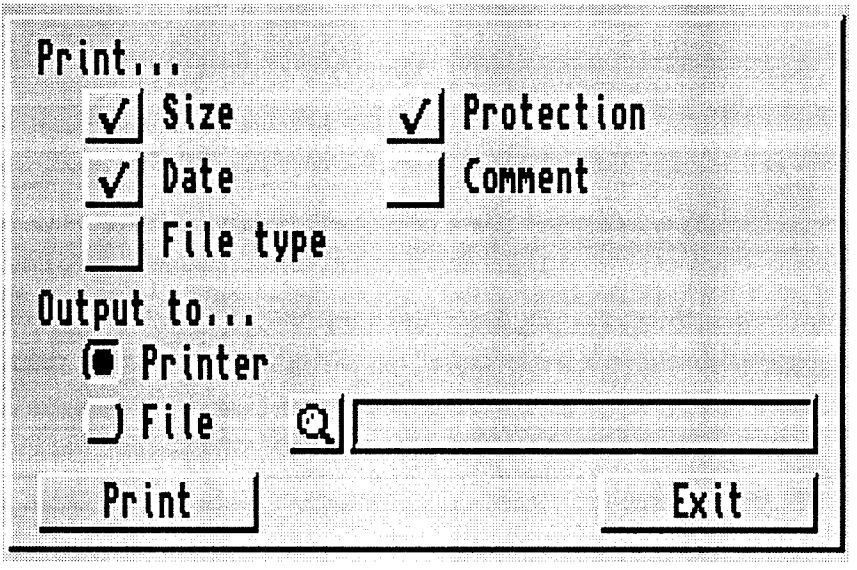

Figure 2.5: Print Dir Requester

only print the selected information when it is being displayed in the Directory Window.[4])

Output to...

By default, the output will be sent to the printer. However, you can redirect the output to a file of your choosing.

Printer This option directs output to the printer.

File When this option is enabled, output is directed to the file indicated in the accompanying field,

Click on the magnifying glass next to the File field to use a file requester to find the file name; otherwise type the name directly into the File field.

[4]Selecting Directory Window information is discussed in the Configuration Chapter on Page 138.

Print

When you click this button, Directory Opus will begin printing the information. It is not started as a separate process, so you might have to wait while Directory Opus completes printing the information.

Exit

This button will abort without attempting to print the directory.

2.6 Print Requester

This Requester gives you full print formatting control for text files.

NOTE: All printers are not created equal. Some printers will ignore some of these configuration options.

NOTE: These configuration options work in accordance with the Amiga Printer Preferences. The Amiga Preferences may override these preferences or simply make the output look silly. For example, you cannot use these options to display more lines on a page than is specified in Amiga Preferences.

There are several configuration items in the print requester:

Top margin

This field contains the Line Number where text should start. A Top Margin of 1 starts on the first line and does not have any blank lines.

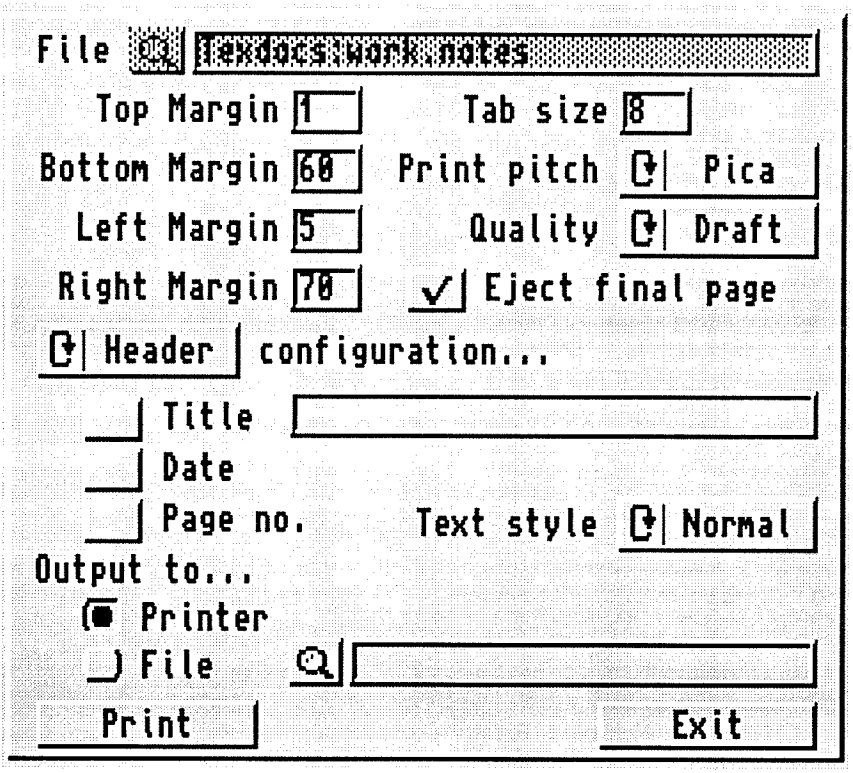

Figure 2.6: Print File Requester

Bottom Margin

This field contains the number of printed lines on each page. The Top Margin Lines are not included in this value. For Example, a Top Margin of 5, and a Bottom Margin of 60 the last line will be printed on Line 65.

Left margin

This field contains the number of characters to skip before printing each line.

Right margin

This field contains the number of printed characters allowed on each line. The Left Margin characters are not included in this value. For example, a Left Margin of 5 and a Right Margin of 70 will result in the last printable character in column 75.

Tab Size

This field contains the Number of spaces to which a tab character is equivalent. Directory Opus converts tabs to spaces and will insert the appropriate number of spaces to create columns based on Tab Size. For example, a Tab Size of 8 specifies Tab positons of 8, 16, 24, 32, 40, 48, 56, and 64.

Print pitch

This button cycles between Pica, Elite, and Fine. These values specify the size of letters to print. Your printer will determine the exact dimensions of these values.

Quality

This button cycles between Letter, and Draft. Some printers can be toggled between letter and Draft quality printing.

Eject Final Page

A check in this button enables a final Form Feed character after printing the last page.

configuration. . .

This button cycles between Header and Footer. The Title, Date and Page no. buttons (described below) can be used with creating a Header or Footer line for each page in the print out. When the

configuration button is Header, these buttons affect the Header line; otherwise, they affect the Footer line. By default, Directory Opus will not create a Header or a Footer line.

☑ Title

When the checkmark next to **Title** is on, a title will be generated. By default, the filename will be the title. However, you can override this by putting text in the Title field. You can have different titles in the header and footer lines.

☑ Date

When the checkmark next to **Date** is on, the current date will printed. Usually this is enabled for either the header line or the footer line, but not for both.

☑ Page no.

When the checkmark next to **Page no** is on, the page number will printed. Usually this is enabled for either the header line or the footer line, but not for both.

[ ] Text style

This button allows you to modify the appearence of all the printed text except the headers and footers. (Some printers do not support all of these styles.) Clicking on the Text Style Cycle button allows you to choose from the following options: Normal, Bold, Italics, UnLined (Under Lined), DblStrik (Double Strike), and Shadow.

◉ Output to...

By default, the output will be sent to the printer. However, you can redirect the output to a file of your choosing.

Printer This option directs output to the printer.

File When this option is enabled, output is directed to the file indicated in the accompanying field.

Click on the magnifying glass next to the File field to use a file requester to find the file name; otherwise type the name directly into the File field. (If you leave the field empty, the output will go to the printer.)

Print

When you click this button, Directory Opus will begin printing the information.

This function will print all selected files, one at a time. If you select only one file to print, the print routine will be started up as a separate process, allowing you to continue working with Directory Opus. To cancel this type of print, simply select the print function again. A requester will appear asking if you want to continue with the print or halt it. This requester will also appear if you attempt to quit Directory Opus while a print operation is in place, as you cannot quit until the print has finished.

Even if you abort a print, the printer may not actually stop for some time. This is because most printers have buffers, some quite large ones, which store data for printing and will need to empty themselves before the printout will stop.

Exit

This button will abort without attempting to print.

2.7 Text Viewer

The name of the file is displayed at the bottom of the screen, giving the current position within the file both in lines and as a percentage, and also the total number of lines in the file.

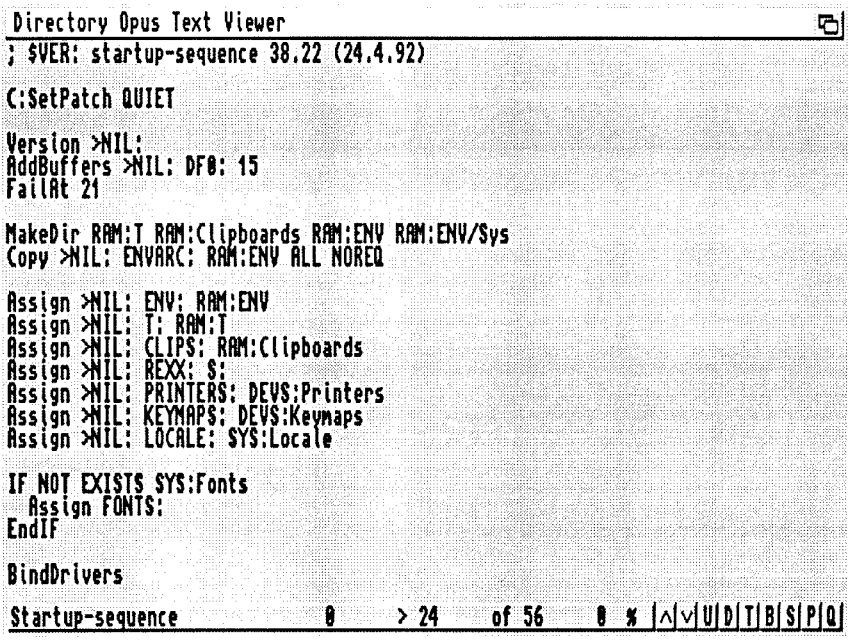

Figure 2.7: Text File Viewer

Press the left mouse button to scroll through the file. Press the left mouse button to turn scrolling on or off. The direction and speed of the scroll are governed by the position of the mouse pointer in the window. The mouse pointer will become invisible while the text is scrolling. No scrolling occurs if you have the mouse in the center of the window.

To scroll forward, move the mouse down until the text starts to scroll (providing there is actually more than one page of text). The

further down you move the mouse, the faster the text will scroll. This procedure is reversed for backwards scrolling.

The buttons in the bottom-right of the screen also allow you to move around the file. Most buttons will repeat if held down for any length of time. (Each button uses the corresponding letter as a shortcut key.)

up/down arrows	Move up and down a line at a time
U	Moves up a page at a time
D	Moves down a page at a time
T	Moves to the top of the file
B	Moves to the bottom of the file
Cursor keys	Move up and down
Cursor key + Shift	Moves one page at a time
Cursor key + Ctrl	Moves to the top or bottom

The current line and percentage values (in the middle of the bottom bar) act as hidden buttons: click on them and a requester will allow you to specify a new value. It will jump to the new location.

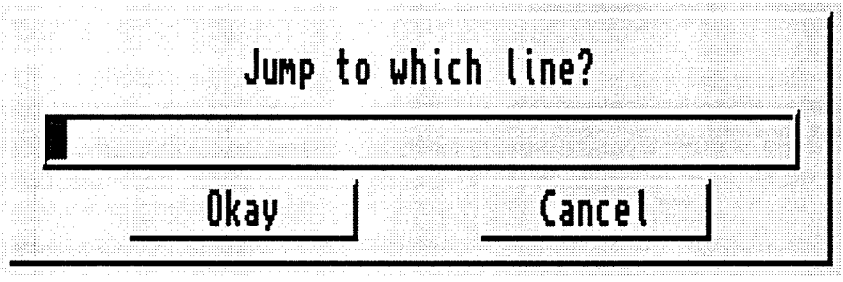

Figure 2.8: Jump to Line in File

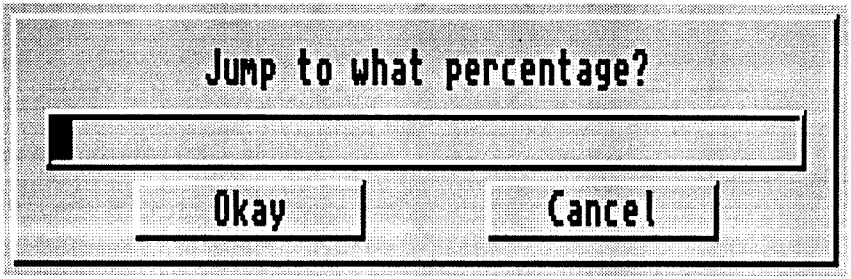

Figure 2.9: Jump to Percent of File

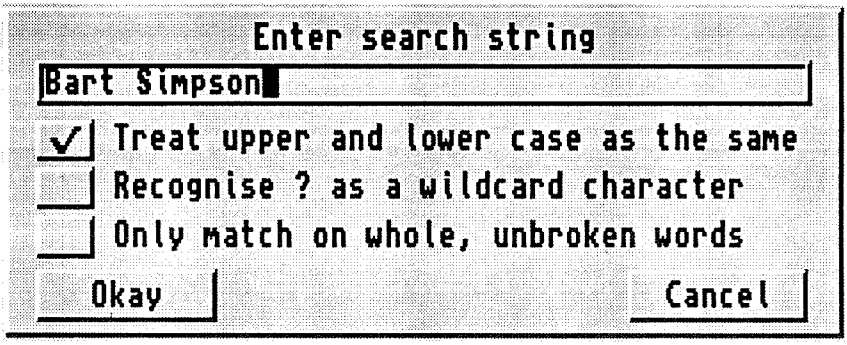

Figure 2.10: Search in Text File Viewer

S Searches for a string. You can use full pattern matching in this search.

P Prints the current file.

C key Prints only the current page. This has no button equivalent.

X or Q or Esc Leaves the text viewer.

You can also use the right mouse button to exit.

If you have selected more than one file, the next one will be read

when you exit. To exit without reading the next file, press the right mouse button.

2.8 Hex Viewer

The Hex Viewer will read files in the same way as READ, except in hexadecimal format. This allows you to view binary files and other files containing non-text characters. The file is displayed in the following manner :

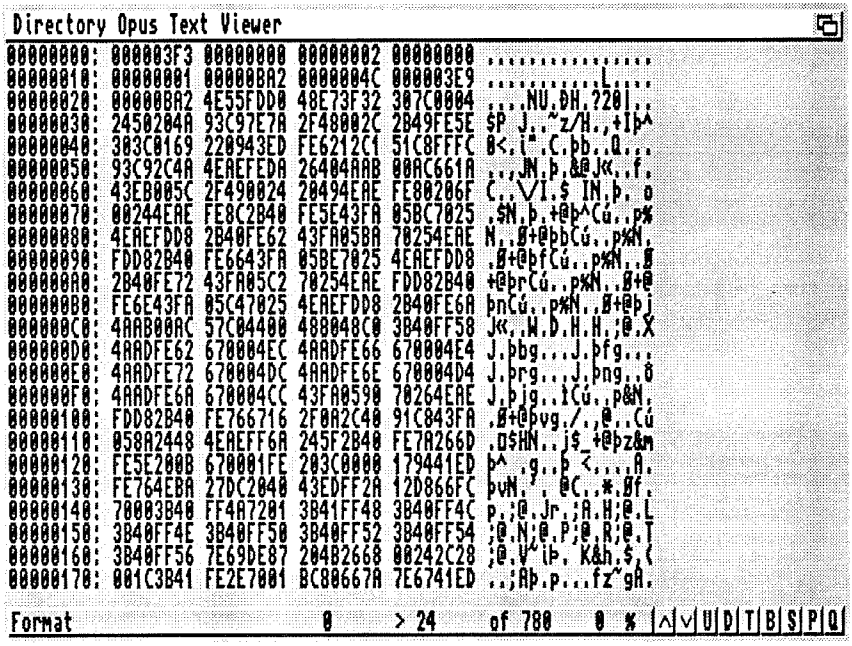

Figure 2.11: Hex File Viewer

The first value is the offset, displayed in hex. This is the number of bytes you are into the file. The next four values are each a four-byte

longword, with the actual ASCII representation at the end. Any non-text characters are shown as a . character.

Chapter 3

Opus Configuration (ConfigOpus)

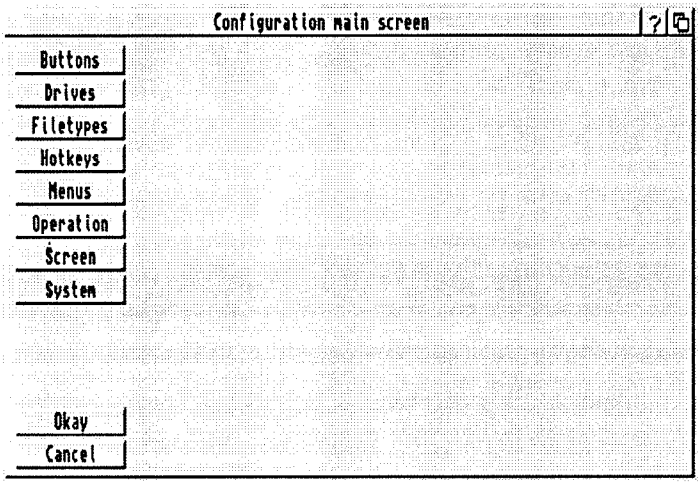

Figure 3.1: ConfigOpus main screen

3.1 Introduction

The Opus Configuration utility is a powerful and easy-to-use instrument for customizing Directory Opus precisely to your tastes. Do you want to use an interlace screen? Eight or sixteen colors? Lots of buttons? Fancy buttons? Directories sorted by date, in reverse order? Instant access to various assignments? With **ConfigOpus**, these are but a few of the myriad choices you can make.

Sophisticated though it is, the ConfigOpus utility is quick and easy to use. This chapter is intended as a complete set of instructions for users new to Directory Opus, and as a convenient reference for experienced users.

Directory Opus uses a separate program to modify the configuration. This saves approximately 130K of memory, as the configuration program need only be in memory when it is being used. Not only that, but the configuration program can be run independently of Directory Opus, if you are really short on memory.

If you do have plenty of memory, however, Directory Opus can be configured to load the configuration program on start-up. In this case, it is run from memory each time the Configure program is invoked, rather than loaded from disk. That can be much faster, especially if you have a floppy-based system.

From within Directory Opus, you can start up ConfigOpus by clicking on the small **C** button on the lower right or selecting the "Configure" item from the Project Menu.

From outside Directory Opus, you can start ConfigOpus by itself either by double-clicking on the ConfigOpus icon or starting it from a CLI.

3.2 ConfigOpus Main Screen

On the ConfigOpus screen (Figure 3.1 on page 77) there are several items of interest. At the top of the screen you will see Screen Depth and Help buttons and the Screen Title Bar.

Screen Depth button Allows the user to push the screen behind the other screens opened.

Help button Brings up help information. When you have questions about something on the screen, try here first. If the Help button does not answer your question, then you can drag out the manual.

Screen Title Bar Describes the current screen. Doubles as the Screen Drag Bar, which allows you to drag the screen down and see the screen(s) behind the ConfigOpus screen.

The next set of buttons to look at are the subject buttons along the left side of the screen. When you select one, the screen will change to show that part of the ConfigOpus.

Buttons

Configure the custom buttons, allowing you to launch internal commands and external programs.

Drives

Configure the custom drive buttons.

Filetypes

Configure the file classes and actions necessary for file recognition.

Hotkeys

Configure the global Hotkeys, allowing you to launch programs from anywhere in the system while Directory Opus is active or iconified.

Menus

Configure the custom menus, allowing you to launch internal commands and external programs.

Operation

Flags and settings that affect the way Directory Opus behaves.

Screen

Modifies the characteristics and appearance of the Directory Opus screen and *graphic user interface*.

System

Additional items that affect the way Directory Opus interacts with the operating system.

The last buttons on this screen (Figure 3.1 on page 77) are the Okay and Cancel buttons. These allow you to say either "OK, let's use this setting and go back to Directory Opus", or "Oops! Forget all that, and let's get out of here!"

3.3 ConfigOpus Configure Menus

There are four menu items available on most screens in ConfigOpus that allow the user to control the configuration data. You can revert to

the default configuration, load a configuration file or save the current configuration. On the ConfigOpus screen these menus will allow you to affect **all** the configuration data. When you select the Open item, you load all the configuration data from that file, replacing any changes that you have made with this new data.

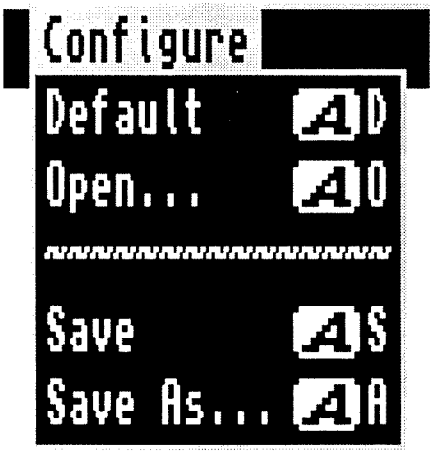

Figure 3.2: Configure Menus

On the other screens in ConfigOpus, the Open menu will only override the information **specific** to that screen. For instance, on the Custom Button screen only the Custom Button banks will be loaded from a configuration file. This allows you to load banks of buttons that you have defined in advance, so as to control the NewTek Video Toaster,[1] for example, or some other application.

Default Reset to the factory default configuration settings.

Open... Load configuration settings from a disk file. These banks will be inserted at the end of the banks that you have already defined. If you are on Custom Buttons, Drives, Filetypes, Hotkeys or Menus this will only load that part of the

[1] NewTek and Video Toaster are Trademarks of NewTek Inc.

configuration file. As an example, by using this, you can import Custom Button banks from someone else's configuration file.

Save Save the current configuration settings as the default.

Save As... Same as Save, but this allows you to specify a filename.

3.4 BUTTONS/MANAGER

There are 42 custom buttons, located at the bottom of the Directory Opus screen. Each button can have a left and/or right mouse button events, bringing the total number of custom buttons to 84.

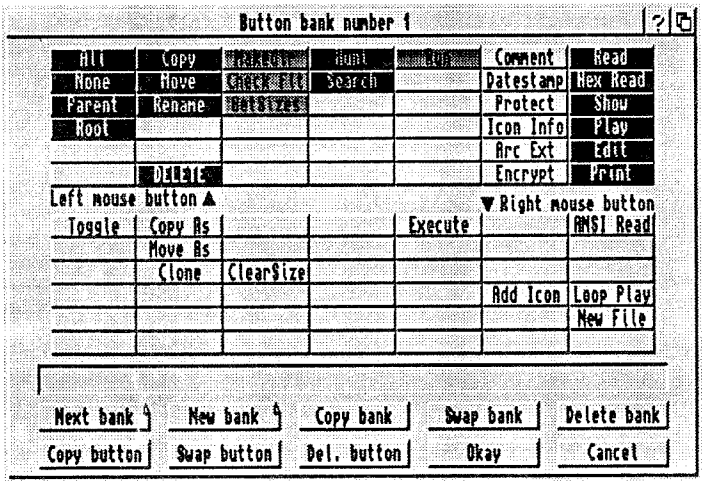

Figure 3.3: Buttons/Manager

The custom buttons are arranged in banks. There is only one bank of these buttons by default, but the configuration program allows you to add or delete as many banks as you want, which means that the number

of custom buttons is limited only by available memory and your imagination.

The button configuration screen (Figure 3.3) displays the current button bank, with the left mouse button events at the top and the right mouse button events at the bottom of the screen.

There are ten buttons across the bottom of the Buttons/Manager screen. The Copy, Swap and Delete buttons stay selected if they have been selected with the left mouse button. For instance, Copy button will stay selected to remind you that you are now in copy button mode. When you want to leave this mode, select Copy button again. The buttons below that work in this fashion are: Copy, Swap and Delete.

Next bank

With this button you can cycle through the button banks. The current bank number is displayed in the title bar of the screen. Select this button with the left mouse button to move to the next bank.

Select this button with the right mouse button to move you back to the previous bank.

New bank

This button allows you to create a new bank of buttons. If you select this button with the left mouse button, it will create an empty bank at the end of the bank list. This bank will then be shown.

If you press this button with the right mouse button, the new bank will be inserted before the current bank, instead of at the end of the bank list.

Copy bank

Use this to copy the currently displayed bank. A requester will

appear asking you to enter the bank number to copy the current bank onto. If the bank number you enter does not exist then a new bank will be created, at the end of the bank list. The current bank will be copied there.

Swap bank

Swap bank allows you to swap the contents of the currently displayed bank with those of another. A requester will appear asking you to enter the bank number to copy to. If the bank number you enter does not exist, the swap operation will be aborted.

Delete bank

After asking for confirmation, this button will delete the current bank. If the current bank is also the only bank, you will be warned.

Copy button

This button allows you to copy one custom button onto another. Select the custom button to be copied, then select a custom button to copy it onto. To abort the copy, turn off Copy mode, select Copy button again, or select the custom button you are copying again.

Swap button

This permits you to swap two custom buttons. It works the same way as the Copy custom button except that it simply exchanges the two custom buttons.

Delete button

After asking for confirmation, this will reset all the data associated with a custom button.

When in Copy or Swap button mode, the Next bank button is still usable and allows you to copy and swap custom buttons between banks.

3.4.1 Button Manager Menus

The Default or Open... menu items in the Configure menu allow you to import custom button banks from either the default configuration or a configuration file from disk. These banks will be inserted at the end of banks you have already defined.

There is an additional menu strip available on this screen. The Button Rows menu allows you to configure the actual number of rows of custom buttons that will be displayed at one time on the main Directory Opus screen. You can choose from zero (no buttons displayed), one, two, three or six rows. This will also affect the drive bank height.

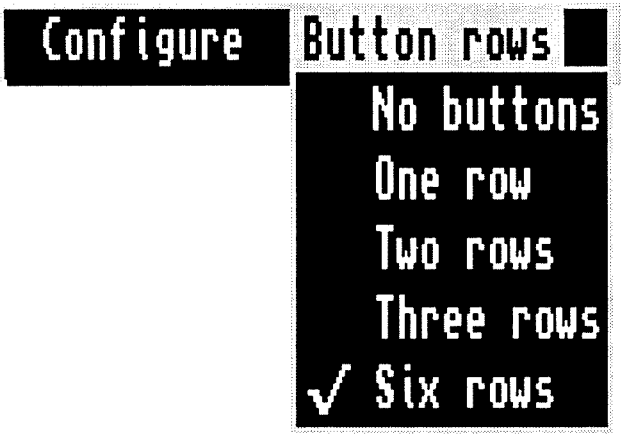

Figure 3.4: Button Rows Menu

To edit a custom button, make sure you are not in Copy, Swap or Delete mode. Select the custom button, and you will be shown the

custom button editor screen.

The last two buttons on this screen (lower right corner) are the Okay and Cancel buttons. These allow you to say, "OK, let's use all these custom button changes and go back to ConfigOpus' main screen", or "Oops! Forget all this, and let's just leave."

3.5 BUTTONS/EDITOR

At the top of the Custom Button Edit screen you can see what the current custom button looks like. The palettes on either side of the example allow you to select the button's foreground and background colors. If you click on the example button itself, it will highlight to show you what the real button will look like when selected.

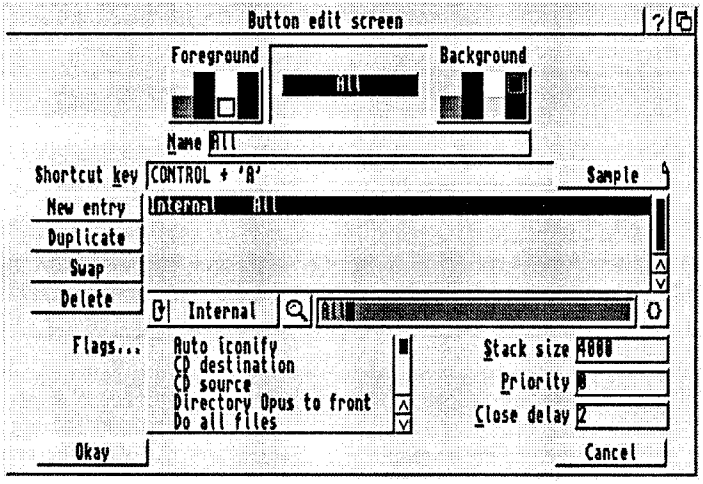

Figure 3.5: Buttons/Editor

Name

The Name field allows you to modify the name that appears on

the button, and is limited to 63 characters.

The **Shortcut key** displays the keyboard equivalent for this custom button.

Sample

To modify the Shortcut key, select the Sample button at the right of the Shortcut key field, then any keys you press, along with any qualifier keys, will be inserted in the Shortcut key field. To turn off Sample mode, select the Sample button again.

To remove the shortcut key, select the Sample button with the right mouse button.

3.6 Command Editor

A Command tells Directory Opus what to do when something happens, and can be attached to Custom Buttons, Filetype Actions, Hotkeys and Menus. Each of these has its own configuration screen with unique aspects (e.g., the title and colors used on a Custom Button).

The unique aspects of the various editors are described in their own sections, but they all have a common portion, referred to as the Command Editor, which allows you to edit the Command.

At the center of the Command editor is the Command List which contains the commands associated with this function. To the left of this list are four buttons that allow you to modify the order and effectiveness of these commands.

New entry Adds a new Command to the list.

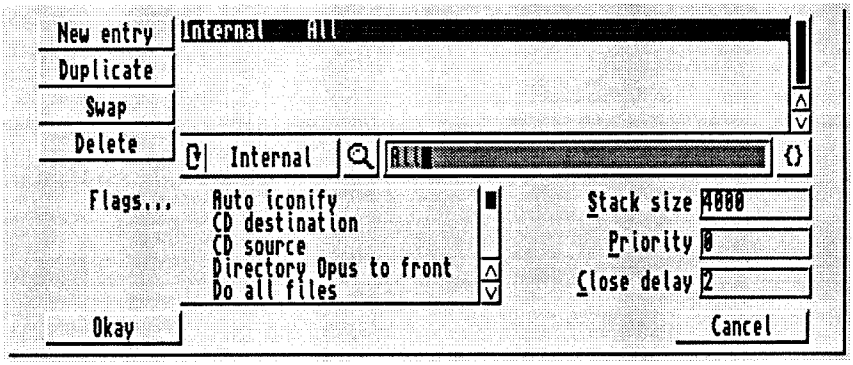

Figure 3.6: Command Editor

Duplicate To duplicate an existing Command, select the Command and select the Duplicate button. The duplicated Command will be added to the end of the Command list.

Swap To exchange the position of two Commands, select the first, then select the Swap button. With the swap button active, select the second Command to swap. To cancel the swap operation, select the Swap button again.

Delete To delete a Command from the list, select the Command and select the Delete button.

To edit a Command, simply click on it. The Command string will be copied to the field below the list for you to edit it.

3.6.1 Edit Fields

Below the Command list (Fig.1.5) is a group of editing tools. These tools allow you to edit active Command entries. When you click on an entry in the Command list it becomes active, or an empty one is created when you select **New entry**.

Command Type

The cycle button on the left below the Command List allows you to specify which kind of Command is used. When you click on this button, it will cycle through the following Command types:

- Command
- Executable
- Workbench
- Batch
- ARexx

Each of these Command types is described below.

When you click on the magnifying-glass button just to the right of the Command Type button, a requester appears allowing you to pick an appropriate entry for the selected Command type. Each of the following descriptions indicates the kind of requester which will appear.

Command These are internal Commands, built into Directory Opus. Many of these Commands can take *parameters* from buttons and menus, as well as from ARexx. Internal Commands are documented in the Commands Chapter.

The magnifying-glass button brings up a list of internal commands.

Executable Executables are launched as if you were running them manually from the CLI. Thus, with an output window enabled, they can receive keyboard input from the user and display output on the screen.

The magnifying-glass button brings up a file requester.

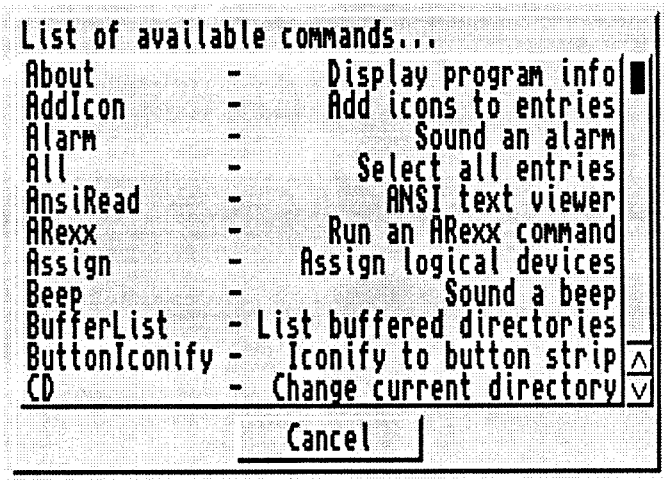

Figure 3.7: Internal Command Requester

Workbench Workbench programs are also executable programs. They are launched, however, as if you were double-clicking on their icons from Workbench. This can be an advantage, as many programs do not take arguments, or do not work at all if run from the CLI.

If the Workbench program given is a tool (i.e., an executable program), Directory Opus will look for its icon file to determine the necessary stack size to give to the program. If the icon cannot be located, the stack size defined in the button, menu or filetype will be used.

If the Workbench program given is a project (a non-executable file created by another program), Directory Opus will look for its icon file to find its Default Tool, the actual program needed to load the file. If the icon cannot be found, or a Default Tool can not be loaded successfully, Directory Opus will not launch the file. The project's icon is also used to determine stack size.

Workbench programs can also take arguments from Directory Opus using the {f} and similar sequences. (Figure 3.9 on page 93) This can be very useful indeed. DeluxePaint, for instance, does not accept arguments if run from the CLI; you would be unable, therefore, to select a picture file for DeluxePaint to load from Directory Opus if you were running DPaint as an Executable.

If, however, you have the command defined as:

DPaint {f}

and have the Command type set to Workbench, DeluxePaint will be run as a Workbench program. From the Workbench, DeluxePaint will accept arguments, so the first file you selected would be loaded into DPaint automatically.

 The magnifying-glass button brings up a file requester.

Whenever Directory Opus opens a Command Editor, it creates an AppIcon on Workbench (Figure 3.8). The Directory Opus AppIcon looks like a normal Icon, but it does not do anything when you double-click on it. However, if you drop an Icon on it, Directory Opus will add a WorkBench Command entry.

Figure 3.8: Workbench AppIcon

Batch Batch files, also called script files, are files that you might run with the **Execute** Command, or with the execute command from the CLI. Selecting a Command type as Batch will cause the file to be executed as a script file.

The magnifying-glass button brings up a file requester. This file requester is initially set to S: because this is where batch files come from by default.

ARexx This type indicates that the Command is an ARexx script. The script will only be launched if ARexx is active in the system. Note that the address of the ARexx port will be set automatically; you do not need to use the ARexx ADDRESS command to address the command to Directory Opus.

The magnifying-glass button brings up a file requester. This file requester is initially set to Rexx: because this is the place ARexx files come from by default.

{}

This button is located next to the Command Edit Field (Figure 3.6 on page 88). This button brings up a list of the argument functions (Figure 3.9). Refer to the Functions section (page 261) of the manual for more information about this powerful feature.

Function strings can contain many different command sequences to do different things with files and directories. For documentation on these, and more information on functions in general, consult the Functions section of the manual.

Flags...

Below the Command list is the Flags list. This is a list of all the flags available for custom Commands. These flags apply to all Commands in the Command list. The flags are:

Auto iconify Causes Directory Opus to enter iconified mode when the Commands are launched. This could be useful in low-memory situations. It only works with Run Asynchronously enabled.

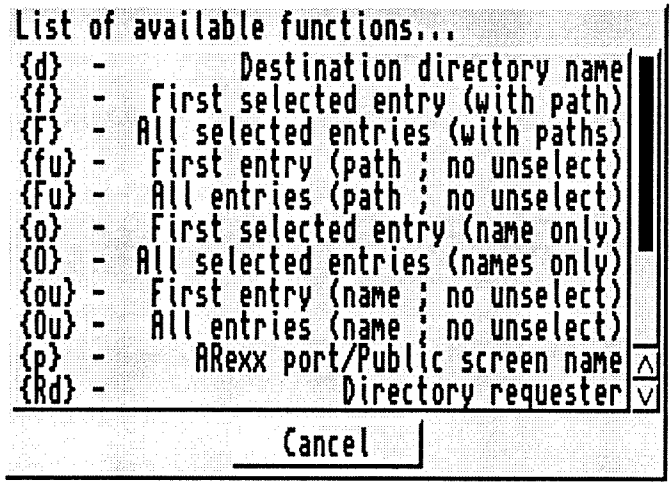

Figure 3.9: Function Argument Requester

CD source If this is turned on, the current directory of the custom Command will be set to the current source directory (active directory window).

CD destination This has a similar effect to CD source, except that the current directory of the custom Command will be set to the current destination directory (inactive directory window).

Directory Opus to front Causes the Directory Opus screen to be brought to the front when the Commands have terminated. It is used in conjunction with Workbench To Front. If enabled in conjunction with Run Asynchronously and Output Window, this flag will have a slightly different effect. In that case, the output window will be opened on the Workbench screen as normal, but the Workbench screen will not be brought to the front.

Do all files This causes the Commands to act on each selected entry in turn, instead of just the first entry. This is used for commands that do not support multiple filenames on the

command line, where {F} to send all selected entries, would not work.

Include DOpus-Startup Has Directory Opus execute a file called S:DOpus-Startup before any external Commands are actually executed. This file is used in the same way as the Shell-Startup file above, but it is distinct from the Shell-Startup file, and therefore allows you to put into it any Directory Opus-specific commands.

Include Shell-Startup If this option is enabled, then Directory Opus will execute the standard S:Shell-Startup file before any external Commands are actually executed. This allows you to have all your path lists, resident commands, etc. available to custom Commands.

No filename quote This option enables Directory Opus to operate correctly with some older or poorly written software. Normally, whenever Directory Opus sends a filename to a custom Command with the {f}, {o}, etc. (Figure 3.9 on page 93) flags, the filename is enclosed in quotation marks. This allows you to use filenames containing space with external programs. However, some software does not interpret the quotation marks correctly. If you find this is the case with any program, simply select the No filename quote flag.

Output window Opens a window for output from these Commands. The window will open on the Directory Opus screen, unless the Workbench to front option or the Run asynchronously option is enabled. In this case, it will open on the Workbench screen, and the Workbench screen will be brought to the front automatically. The window is opened using the handle specified in the System/AmigaDOS section of the configuration.

Output to file Redirects all output from the Commands to a temporary file in the T: directory, which is then read via the

`text viewer`. This allows you to read the output of a program thoroughly, and even to print it. Note that if you are sending output to a file, the Command cannot receive input from the keyboard.

Recursive directories Allows the Command access to files within sub-directories. Normally, whenever a {f} or {F} or similar sequence would result in the name of a directory being included in the program's parameters, Directory Opus would treat the directory in the same way as a file. In other words, the Command would not act recursively on all files within the directory.

If this option is enabled, the names of all files within that directory, and within sub-directories within the directory, and so on, are included in the program's parameters. This allows the Command to act on all files in the directory and not just on the directory itself.

Reload each file Causes Directory Opus to rescan a file after it has been acted upon by a Command, and updates the size, datestamp, comment and protection bits of the file. You can therefore reflect changes in size, for instance, made by a text editor to a file.

Rescan destination This flag makes Directory Opus reload the destination directory (inactive directory window) when the Command terminates. This, and the above option, allows Directory Opus to display correctly any changes made to either directory window by external programs, such as archivers.

Rescan source Makes Directory Opus reload the source directory (active directory window) when the Command terminates.

Run asynchronously Indicates that the Commands are to be launched as a new process, and Directory Opus is not to

wait for it to return. If this is the case, and an output window is specified, the output window opens on the Workbench screen.

Workbench to front Causes the Workbench screen to be brought to the front.

Three fields at the lower right allow you to specify stack size, priority and output window close delay.

Stack size (Figure 3.6 on page 88)

The stack size is the size (in bytes) that external programs will use as stack space. The default and minimum is 4000 bytes. The documentation of the program you are running should advise you of the best stack size to use.

Priority (Figure 3.6 on page 88)

The priority field allows you to determine the priority at which the programs will run. The default is 0. You should not normally use anything lower than -10 or higher than 10, except for certain programs (the documentation of the program may advise you of appropriate priorities).

Close delay (Figure 3.6 on page 88)

This field is the number of seconds after the Commands have finished before the output window closes (if one is specified). The default is 2, which means that the output window will close two seconds after the Commands have terminated. If you set the close delay to 0, the output window will close immediately. If you set it to -1, it will wait for you to press the left mouse button before closing.

3.6.2 Edit Menu

If you move the mouse to the top of the screen and press the right mouse button, you will see that the main Configure menu has been replaced by the Edit menu. The options in this menu operate on the currently displayed Command List. This includes the Flags and fields.

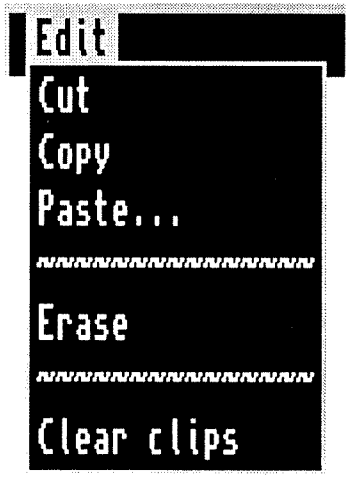

Figure 3.10: Command Edit Menus

All screens with a Command Editor contain the Edit Menu with Cut, Copy, and Paste functions. These allow you to swap function definitions between custom buttons, menus and Filetypes.

Cut Copies the settings of this Command List to the clipboard then erases them, resetting all fields to their default settings.

Copy Copies the settings of this Comand List to the clipboard, but will leave the values untouched.

Paste... Displays a list of all Command Clips in the clipboard (if there are any present). If you select an entry from this list, it will

be pasted into the current Command, replacing any information that was there already.

Erase Resets all fields to their default settings, erasing the Command List. The Command will *not* be copied to the clipboard.

Clear Clips When you use the Cut and Copy Edit Menus, the Commands are stored in the file `t:configopus.clip`. Clear Clips deletes this file, causing all previously clipped Commands to be cleared.

3.7 DRIVES/MANAGER

The Drives button configuration screen allows you to configure the drive buttons. These buttons give you quick and easy access to a device or directory. There are 30 drive buttons, arranged in five banks with six buttons in each. The Drives configuration screen displays all 30 drive buttons at the top.

There are ten buttons across the bottom of the screen. Some of these buttons stay selected when selected with the left mouse button. For instance, Copy bank will stay selected, reminding you that you are now in copy bank mode. To leave this mode, select Copy bank again. The buttons below that work in this fashion are: Copy bank, Swap bank, Delete bank, Sort bank, Copy drive, Swap drive and Delete drive.

Get drives

Causes the drive buttons to be assigned to the names of all the devices and assigned directories (the first 30) present in your system. You would normally only do this when performing a major reconfiguration.

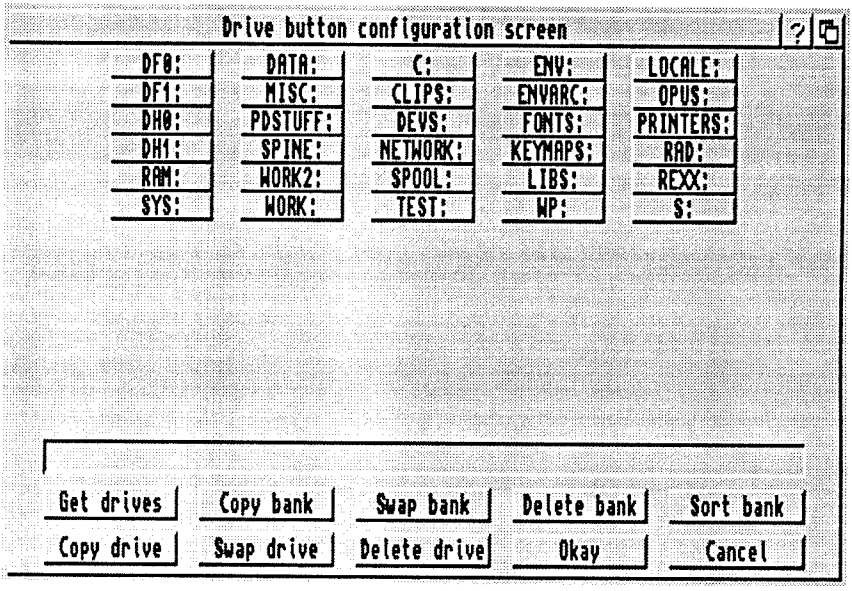

Figure 3.11: Drives/Manager

Copy bank

Copies one drive bank onto another. Select any drive button in the drive bank to copy, and then any drive button in the destination bank.

Swap bank

Allows you to exchange the contents of two drive banks. This works the same way as the Copy bank button.

Delete bank

Erases the contents of a drive bank. If a drive bank has no buttons defined in it, it will not be displayed on the main Directory Opus screen (ie, Directory Opus skips over unused drive banks). In fact, if you delete all the banks, the drive banks

will not even be shown on the main Directory Opus screen. This allows the main custom buttons to be wider.

Sort bank

Alphabetically sorts the contents of a drive bank. Select any drive button in a bank and that bank will be sorted.

Copy drive

Copies one drive button onto another. Select the drive button to be copied, then select a drive button to copy it onto. You can abort the copy by turning off copy mode, selecting Copy drive again, or selecting the copied drive button a second time.

Swap drive

Swaps two drive buttons. This works the same way as the Copy drive button except that it exchanges the two drive buttons.

Delete drive

After asking for confirmation, resets all the data associated with a drive button.

The last two buttons on this screen (lower right corner) are the Okay and Cancel buttons. These allow you to say "OK, let's use all these drive button changes and go back to ConfigOpus' main screen", or "Oops! Forget all this, and let's just leave".

3.8 DRIVES/EDITOR

When you select a drive button from one of the banks, the Drives

Editor will appear or be updated with that drive button's information. The drives editor will be displayed directly below the drive banks.

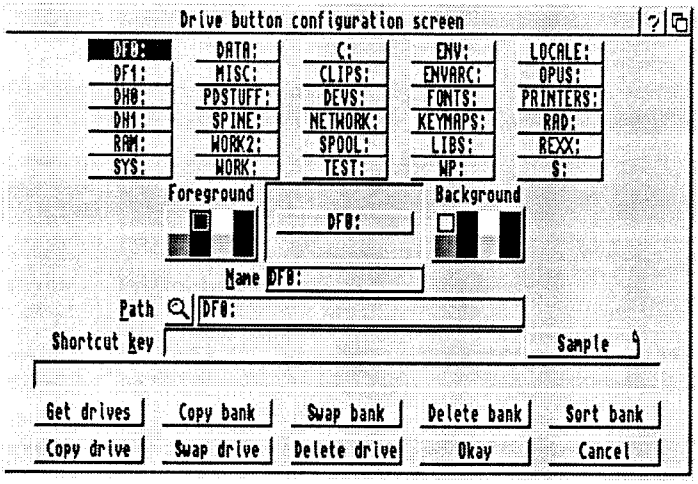

Figure 3.12: Drives/Editor

In the middle of the screen is an example of what the current drive button will look like. The palettes on either side of the example allow you to select the foreground and background colors of the button.

Name

This field allows you to modify the name that appears on the drive button. The Name field is limited to fifteen characters.

Path

This is the actual AmigaDOS path that is read in whenever this drive button is selected. The Path field is limited to to 256 characters.

Shortcut Key is a display area for the keyboard equivalent for this drive button.

Sample

To modify the Shortcut key, select the Sample button. While the Sample button is selected, any keys, along with any qualifiers, you press will be inserted in the Shortcut key field. To turn off Sample mode simply select the Sample button again.

To remove the shortcut key simply select the Sample button with the right mouse button.

3.9 FILETYPES/MANAGER

A file is simply stored data. Files can contain executable programs, IFF pictures, Icons for Workbench, or a multitude of other kinds of data. Most (but not all) files have an identifiable structure. Directory Opus's **FileTypes** system is designed to examine a file's structure and identify the kind (or Type) of data it contains. You can configure Directory Opus to understand and unlimited number of Filetypes.

Filetypes are a versatile feature of Directory Opus. Using Filetypes, you can configure Directory Opus to play animations when they are double-clicked, to load a database program when you attempt to "Read" a database file, or to uncompress an Archived file when you click-m-click on it.

This is the essence of the Filetypes: when you do something to a file, Directory Opus can figure out what kind of file it is, and take the appropriate action for that type of data.

There are two sections to the Filetypes configuration:

Filetypes There is a list of Filetypes which is searched when an action occurs. If the file being processed matches a FileType, you can configure Directory Opus to perform a specific action.

File Classes There is a DataBase of File Class definitions: each describes the process of matching a particular kind of File. Directory Opus comes with more File Classes than you will probably use. You can select from this list when you are creating a new Filetype entry. You don't have to be an expert or know anything about the internal structure of the file to do this.

The File Class Manager, described in the next section, allows you to edit existing File Classes, or define new ones. (This can be somewhat more complicated.)

3.9.1 Filetypes

The Filetype configuration screen displays the list of filetypes that Directory Opus recognizes. By default, it recognizes LHA archive, ARC archive, ZOO archive, and Workbench icons.

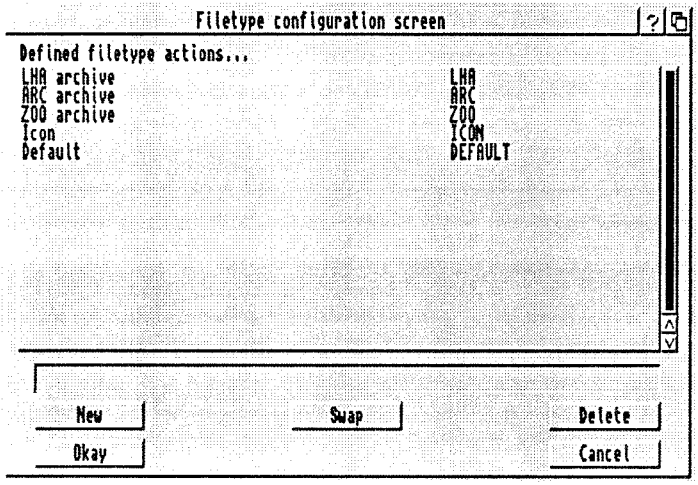

Figure 3.13: Filetypes/Manager

There are five buttons across the bottom of the screen. Some of

these buttons when selected with the left mouse button stay selected. For instance **Delete** will stay selected indicating that you are now in delete mode. To leave this mode simply select **Delete** again. Both the **Delete** and **Swap** buttons work in this fashion.

New

This allows you to create a new filetype entry based on a predefined File Class. When you select this button a requester appears containing currently defined classes. To create a new entry for one of these File Classes simply select it and you will be brought to the filetype action editor (described below).

Swap

This button allows you to exchange the positions of two Filetypes in the list. This can be very useful or even neccessary. In the case of having two filetypes defined in which one is a sub-set of the other, e.g., 24 bit ILBM pictures vs. regular IFF ILBM pictures, you would want the 24 bit ILBM pictures to come first because they are a special case of the regular IFF ILBM picture filetype. Otherwise, pictures (in this case) will be matched with the regular IFF ILBM picture filetype and will never have a chance to match with the sub-set 24 bit ILBM picture filetype.

To actually perform a swap, click on the **Swap** button to enter that mode. Click on the first filetype, then on the second. Those two filetypes will exchange places in the list. To exit swap mode simply press **Swap** a second time.

Delete

In this mode you can select filetypes to remove from the list. Click on the **Delete** button, then click on the entries you wish to delete. If you accidentally delete an entry, click on the **Cancel** button.

Okay and Cancel

These buttons allow you to say "OK, let's use all these Filetypes changes and go back to ConfigOpus' main screen", or "Oops! Forget all this and let's just leave".

To edit a filetype, make sure you are not in Swap or Delete mode. Then simply select it. You will then be taken to the Filetype Editor screen (Figure 3.14).

3.9.2 Menus

There is an additional menu on this screen which controls File Classes. Please refer to the section on File Class Manager for information about this menu.

3.9.3 Editing an existing entry

To edit an entry in the list, simple click on it and you will be taken to the filetype editor (described below).

3.10 FILETYPES/EDITOR

This editor allows you to define the actions taken on a matching Filetype. In a preceding section, you learned to configure an action when a custom button is selected. Essentially, you say "When this button is clicked - Do this.". In the Filetypes editor, you say: "When I do this to this particular Filetype - Do that." The primary difference is that a custom button can only be activated by being clicked on; the

functions available for editing Filetypes can be activated in several ways.

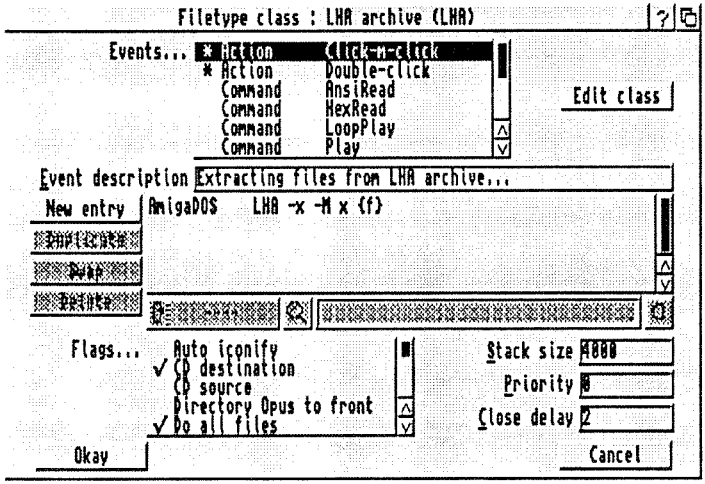

Figure 3.14: Filetypes/Editor

Events...

At the top of the Filetypes Editor is a list of Event types, which are instructions you may choose to give Directory Opus regarding Filetypes. Each of these Events is associated to either a mouse action or a Directory Opus command, so each entry on the list is tagged either *Action* or *Command*. An asterisk * indicates that that entry already has an Event Command List defined for it. Clicking on an entry in the list highlights it to show that it is the Command List currently being edited. The current Command List is displayed in the editing list in the middle of the screen. The small field immediately above it describes the Event.

While each of these Event types can be defined to do something, usually only a few are actually defined. It certainly is not necessary to define all Events for a Filetype.

When one of these Events is activated, Directory Opus does the following:

- It first searches the Filetypes list, starting with the first entry, and checks whether it matches the entry's File Class.
- If it matches, it looks to see if the corresponding Event is defined from this Editor. If it is defined, it performs the Command List.
- If it does not match the File Class, or if it matches and does not have a Command List defined, it continues to search the Filetypes list. It is possible for it to match a subsequent entry that does have a Command List defined.

3.10.1 Mouse Action Events

Mouse Events occur when you either Double-click on a file or perform a Click-m-click.

Double-click This occurs when you double click on a file. A popular use of this is to examine a file (e.g., show it if it is a picture or play it if it is a sound). The actual double click speed is defined by the operating system's preference program.

Click-M-Click This Event occurs when a file is clicked on then the mouse is moved to the other window and clicked. One popular use of this Event is for archive extracting. Another way this Event can happen is by using click-drag-release. With this you literally drag a file into the window and drop it there.

3.10.2 Command Events

Command Events are called when a file is acted upon by a Directory Opus command. For example, when a **Read** Command is performed

on a file, Directory Opus will search for a FileType which matches the File and has a Read Event defined. You could, however, have **Read** do something special for an archive (i.e., list the contents of the archive). Likewise if you have a music score you could define the **Play** (or **LoopPlay**) to call up your player for that music score.

Each of the following Command Events is associated with a corresponding Directory Opus command. This allows you to redefine the specific action taken for a particular filetype. By convention you should use a command Event for something similar to its normal use, but there is nothing to stop you from giving a command Event a label that is totally irrelevant to its use. It's up to you to keep the context of the command (i.e., a Read would be confusing if it did a Disk Format).

The following list describes the normal usages. You can decide what you want it to mean for any given filetype.

AnsiRead Fancy text viewer

HexRead Binary or Hexadecimal file viewer

LoopPlay Normally associated with playing sounds in loops

Play Playing sounds

Read Simple text viewer or viewing contents of archives

Show Picture viewer

User Events The terms User1, User2, User3, and User4 may seem cryptic, but they are here to give you flexibility. Each of the preceding commands has an implied usage, but you may have an application which doesn't really mean any of these. In that case, you can decide that one of these User Events means "Perform this special operation".

Directory Opus' default configuration employs User1 for extracting from archives.

Edit Class

The button to the far right of the list allows you to modify this filetype's class definition. We will talk about editing File Classes in the File Class section (see page 109).

Below the list of Events associated with this filetype is a field describing this particular Event. For instance you could have the Read Event say something like "Reading file...". This message will appear in Directory Opus' Title Bar when the Event is being performed.

The remaining part of this requester is described in the Command Editor section (see page 88).

3.11 FILE CLASS/MANAGER

File Class management is handled by a menu accessible from the Filetype manager screen (Figure 3.15 on page 110) that offers management tools and storage options:

New Selecting this menu option allows you to create a new File Class definition.

Edit This will open a requester showing you all the currently defined File Classes. To pick one to edit, simply select it; you will then be brought to the File Class editor. To abort press cancel.

Once you have made a filetype entry based on a File Class definition then you must use the **Edit Class** button on the filetype editor screen.(Figure 3.14 on page 106)

When you make a Filetype, it uses a copy of the File Class in the actual filetype entry. This has several important results. Once a Filetype is made, you can edit its File Class definition without changing the original definition. The **Edit Class** button on the

Figure 3.15: File Class Menus

filetype editor allows you to do this. Selecting Edit from the Classes Menu allows you to edit the original File Class Definition. This will not automatically update any Filetypes which used the original definition. You can even delete the original File Class without affecting a filetype which used it.

Duplicate This allows you to create a new File Class based on an already defined File Class.

Delete This will bring up a requester allowing you to select a File Class to clear from memory. Simply select the File Class to delete and respond Yes to the confirming requester. Press Cancel to abort deleting.

Open With this you can import File Class definitions from a file.

Save Save the current File Classes to their default place on disk.

Save As... Save the current File Classes to a specific file on disk.

Clear Clear all File Class definitions from memory.

3.12 FILE CLASS/EDITOR

Selecting New, Edit, or Duplicate from the Filetype manager screen (Figure 3.15 on page 110) or clicking **Edit class** button on the Filetypes Editor (Figure 3.14 on page 106) brings up the File Class Editor.

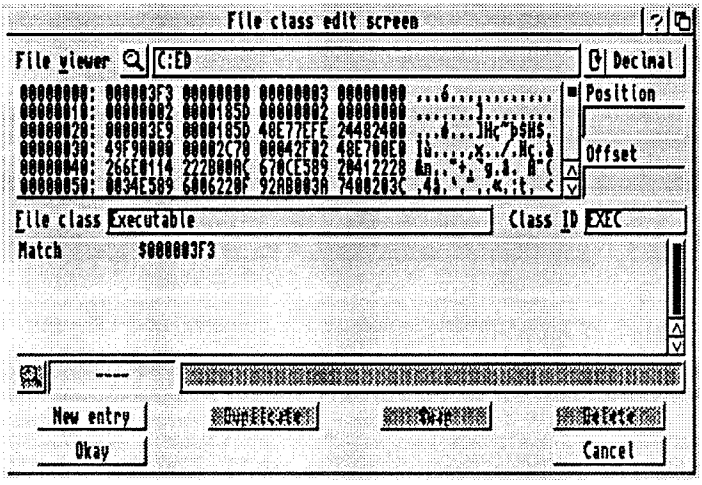

Figure 3.16: File Class Editor

This screen is the most complex section of ConfigOpus. It even looks scary, but don't worry: it is rather simple once you get to know it. The best way to describe this screen is to start at the top and go down from there.

File viewer

The field at the top of the screen is simply a file viewer. To view a file, type its name into the field or click on the magnifying-glass to bring up a file requester. The file selected will appear below the field. It will be displayed in Hex/ASCII format similar to the Hex Read command in Directory Opus itself.

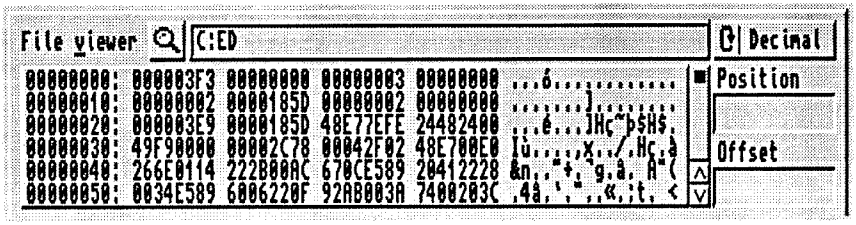

Figure 3.17: File Class - File Viewer

You may wonder why we have a file viewer on this screen. We have found that when creating a new File Class definition it is useful to have a way to look at files, since you are trying to describe to Directory Opus a way of identifying files.

C| To the right of the file viewer is a button that cycles from **Decimal** to **Hex**. This button tells ConfigOpus in what format to display the Position and Offset information in the two read-only fields below the cycle button.

The simplest way to show how this all works is with an example. Select the file "C:ED" (by either typing into the field or using the magnifying-glass file requester).

The format of the data in the fileviewer is as follows: The first column is the address into the file we are looking at. The next four columns are 16 bytes of data from the file in hexadecimal format. The last column is simply those 16 bytes in ASCII format. Where there is a byte that is '00' or null we display it in ASCII as a period.

Position your mouse pointer over the 'F3' characters and press the left mouse button. A cursor will appear and the position read-only field will be filled in with '000000003'. This means that the cursor is positioned on the 4th byte in the file (remember byte locations are zero relative, that is, first byte is at location zero). If you click the cycle

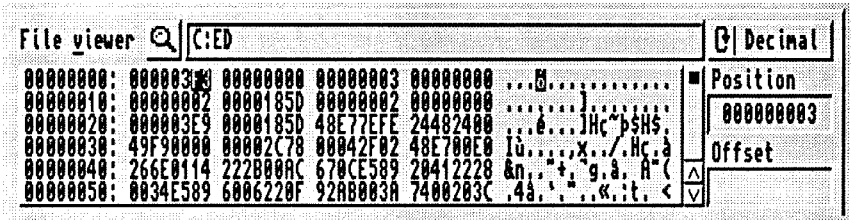

Figure 3.18: File Class - File Viewer (showing position)

button above the position read-only field and change it to Hex then the position will be filled in with '$00000003' which is hexadecimal for three. You can move the cursor around in the file with either the mouse pointer (simply select another place in the file) or the cursor keys.

Figure 3.19: File Class - File Viewer (showing offset)

Now double click on those 'F3' characters. This time the characters change colors. They are now the position from which offsets are computed. Move the cursor by pressing the down key a few times and you will notice the offset read-only field contains the offset from the double click position.

3.12.1 File Class Editor Components

That's about it for the file viewer part of the File Class editor screen. Now we go off to the real power house of Directory Opus.

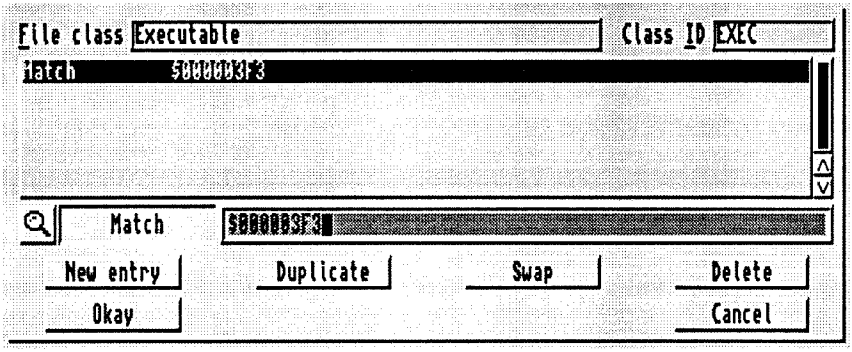

Figure 3.20: File Class Editor Components

File Class

The field below the file viewer is for the name of the File Class. (Figure 3.16 on page 111). This is for the name of the File Class.

Class ID

The Class ID will appear beside the Filetype in the Filetype manager screen and is a shorthand way for Directory Opus to write the name of the file class.

Underneath the fields is the File Class identification definition. This is a series of actions that Directory Opus will perform in order to identify a file class. The action may be as simple as matching a filename to a pattern or as complex as scanning an IFF form looking for data in a specific IFF chunk.

Below the list is a magnifying-glass, a read-only field and an argument field. These are used for editing the File Class identification definition (see page 115).

New entry

Creates a new entry in the File Class definition script.

Duplicate

Duplicates the current File Class definition script line. The copied line is inserted after the current line.

Swap

Lets you swap the positions of two File Class definition script lines.

Delete

Will let you delete the current line.

3.12.2 Editing

To edit a line simply click on it, and the read-only and argument fields will be filled in. To change the command in the read-only field, click on the magnifying-glass button and a list of other commands will be displayed (Figure 3.21 on page 116). Select the one you want or press Cancel to abort.

The general structure of the File Class definition script is that there is a clause or a sequence of clauses that describe what should be considered a matching file for a given filetype. There are only two directives that delimit clauses: And and Or. They are used to tell the system what to do if a clause fails or succeeds.

When all the clauses are finished and the result is true then the file is of the right type.

3.12.3 File Class Edit Commands

And if the preceding clause succeeds, then also do this clause; otherwise skip to the next clause. If the preceding clause

executed did not succeed, then execution stops and the file does not match.

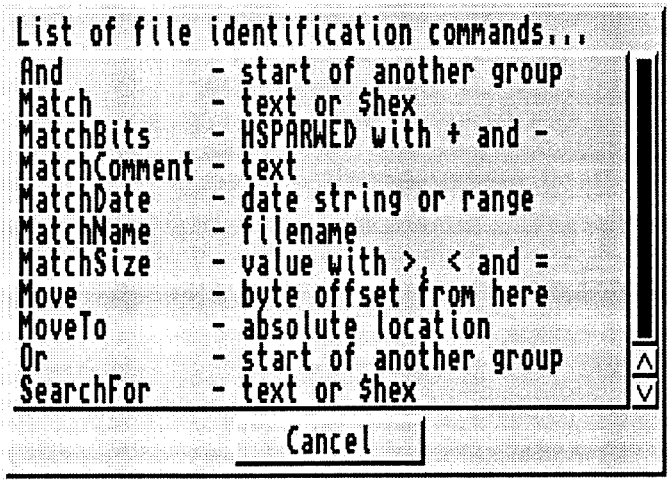

Figure 3.21: Filetype Edit Commands Requester

Or if the preceding clause fails then do this clause; otherwise skip to the next clause.

3.12.4 Testing directives

Match Match text or $hex

This states that a sequence of bytes starting at the current file offset must match the given pattern. To match a single unknown character when text is given, use the **?** character. To match a single unknown byte when $hex is given use two of them (**??**) You can also use \000 syntax in text to specify ASCII characters by their decimal number. A \009 would be a tab character, a \114 would be the lowercase **r**.

Example:

```
Match $000003F3 (executables start with these bytes)
Match FORM????ILBM (the way a IFF ILBM picture starts)
Match Hey\009Overthere ("Hey" then a tab then "Overthere")
```

MatchBits MatchBits HSPARWED

This tests the file's protection bits. To see if a bit is set, put a **+** before the character. To see if the bit is unset use the **-**.

Example:

```
MatchBits +RW - read write must be on, others don't matter
MatchBits -E - executable must be off, others don't matter
MatchBits +RW -E - read, write must be on and no executable
```

MatchComment MatchComment text

Simply compare the text given against the comment of the file. Any valid AmigaDOS wildcard pattern is usable here.

Example:

```
MatchComment Silly_Picture (file with silly_picture as a comment)
MatchComment #?eddie#? (any file with eddie in its comment)
```

MatchDate MatchDate dates

Test the date of the file against a date. For information about datestrings or ranges see the Select command (see page 246).

Example:

```
MatchDate 08-Sept-92
MatchDate < 10-Jan-92
```

MatchName MatchName filename

The filename must match the character pattern given. Any valid AmigaDOS wildcard pattern is usable here.

Example:

```
MatchName #?.ilbm
MatchName *.lzh
```

MatchSize MatchSize > or < or = integer

Test the size of the file against a value.

Example:

```
MatchSize > 1000
```

3.12.5 Movement directives

MoveTo MoveTo ByteLocation

Move to a specific byte offset from the beginning of the file. Initially you are always at the beginning of the file, but you may have been moved in a previous clause, so you might want to put a MoveTo at the beginning of a clause in order to know exactly where you are.

Example:

MoveTo 0 (back to beginning of the file)
MoveTo 100 (move to the 101st byte of the file)

Move Move ByteOffset

Move to a byte relative to the current file offset.

Example:

Move 16 (move sixteen bytes forward into the file)
Move -4 (move back four bytes from where we are)

SearchFor SearchFor text or $hex

Search (starting at the current file offset) for a certain byte pattern that matches the given pattern. See the Match command for valid options for this directive. If the match occurs, then the current file position will be the first character matched.

Example:

SearchFor CMAP (look for the 'CMAP' , position on the 'C')
SearchFor M.K. (search for 'M.K.', position on the 'M')

A failure of any Movement directives causes the clause to fail.

Example

One example of usage is the file class 24bit picture.

Example:

Match FORM????ILBM (file must start with these characters)
And (if the previous cause is true then do the following)
SearchFor BMHD (then search for the BMDH chunk ID)
Move 16 (move sixteen bytes into the file)
Match $18 (this must be 24 (or $18 in hex) to be a 24bit picture.)

We would suggest looking at the predefined File Classes to get an idea of the type of thing you can do with this system.

3.13 HOTKEYS/MANAGER

Directory Opus can have an unlimited number of Hotkeys defined. Hotkeys are systemic global key definitions. These keys are available from every application you run, as long as Directory Opus is running or iconified. If Directory Opus is busy doing something else (i.e., copying files, etc.) then the key's operation will start up as soon as Directory Opus has finished. Under 2.0, Hotkeys are implemented via Commodities and are controllable via the Commodities Exchange program (see Amiga 2.0 manual).

There are six buttons across the bottom of the screen. Some of these buttons stay selected when selected with the left mouse button. For instance, Delete will stay selected to remind you that you are now in delete mode. To leave this mode, select Delete again. The buttons below that work in this fashion are: Duplicate, Swap and Delete.

New Hotkey

Takes you to the Hotkey editor so you can define a new Hotkey.

Duplicate

Duplicates a Hotkey. You can abort the copy by simply turning off duplication mode, selecting Duplicate again.

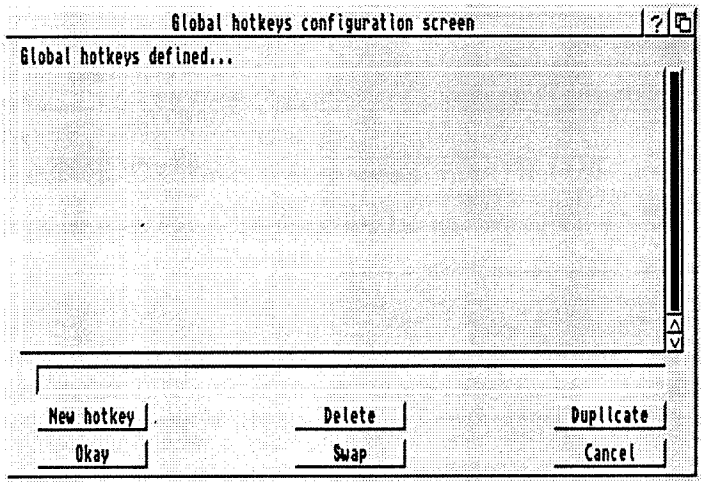

Figure 3.22: Hotkeys/Manager

Swap

Allows you to swap the order of two Hotkeys. Remember that the first Hotkey is seen first, so if you have two Hotkey entries defined to the same key sequence, only the first one will be seen.

Delete

After asking for confirmation, this deletes Hotkeys.

3.13.1 Menus

The Default or Open... menu items in the Configure menu allow you to import Hotkeys from either the default configuration or a configuration file from disk.

To edit a Hotkey, make sure that you are not in Duplicate, Swap or Delete mode, then simply select it. You will then be taken to the Hotkey editor screen.

The last two buttons on this screen (lower right corner) are the Okay and Cancel buttons. These allow you to say, "OK, let's use all these custom button changes and go back to ConfigOpus' main screen", or "Oops! Forget all this, and let's just leave".

3.14 HOTKEYS/EDITOR

Name

This field is simply a name for the Hotkey and can contain anything. Directory Opus only uses this as a label so that you will know what the Hotkey does.

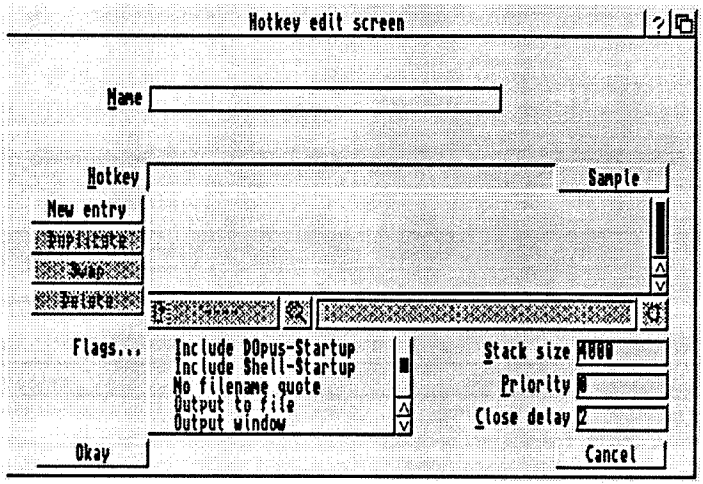

Figure 3.23: Hotkeys/Editor

Hotkey is a display area for the Hotkey sequence. It shows the defined Hotkey or is empty if one has not been defined.

Sample

To modify the Hotkey, select this button. While the Sample

button is selected, any keys, along with any qualifiers you press will be inserted in the Hotkey field. To turn off Sample mode, select the Sample button again.

To clear the Hotkey sequence simply select the Sample button with the right mouse button.

The remaining part of this requester is described in the Command Editor section (see page 88).

3.15 MENUS/MANAGER

This section allows you to configure the menus. There are 100 custom menu items, arranged as five menus of twenty items each. The Menu Configuration screen displays the five menus, with the menu titles at the top.

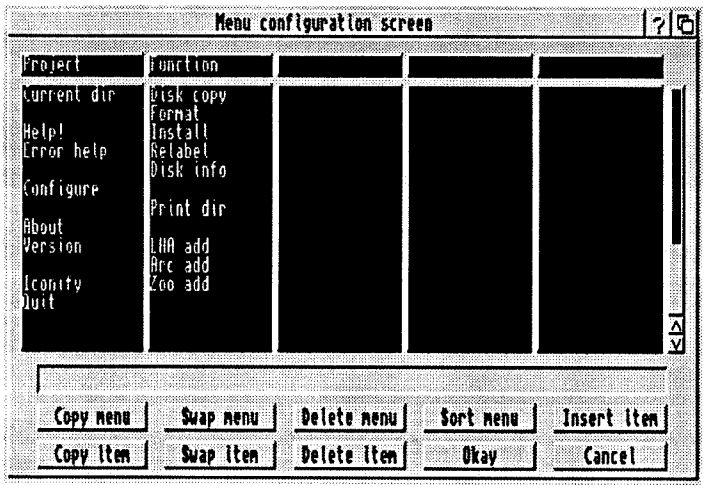

Figure 3.24: Menus/Manager

All the menu items in each menu strip are not displayed at once; because space is limited on the screen, only a certain number are visible at a time. To move through the menu items, use the slider at the right of the screen.

There are ten buttons across the bottom of the screen. Some of these buttons stay selected when selected with the left mouse button. For instance, Delete Item will stay selected to remind you that you are now in Delete Item mode. To leave this mode, select Delete Item again. The buttons below that work in this fashion are: Copy menu, Swap menu, Delete menu, Sort menu, Insert item, Copy item, Swap item and Delete item.

Copy menu

Copies the contents of one menu to another. When you have selected this option, select a menu item in the actual menu you wish to copy, and then a menu item in the menu you want to copy the first menu to.

Swap menu

Swaps the contents of two menus. It works in the same way as the Copy menu function, except that the contents of the second menu are exchanged and not replaced.

Delete menu

Removes all functions from one of the five menus. Select a menu item in the appropriate menu; a requester will appear to verify the operation. If you select Okay, the contents of the menu will be deleted.

Sort menu

Sorts the contents of each selected menu alphabetically by name.

Insert item

Allows you to insert a new item between two existing items. Select the item in front of which you want the new item to be inserted.

Copy item

Allows you to copy one menu item to another. When you have selected this option, select the menu item to copy, and then the menu item to copy it to.

Swap item

Allows you to swap two menu items. It works in the same way as the Copy item function, except that the second item is exchanged and not replaced.

Delete item

Deletes menu items. Select the menu item you wish to delete. All menu items below it will be moved up to fill the empty space.

The Default or Open... menu items in the Configure menu allow you to import menus from either the default configuration or a configuration file from disk.

To edit a menu item, make sure you are not in Copy menu, Swap menu, Delete menu, Sort menu, Insert item, Copy item, Swap item or Delete item mode. Then simply select it, and you will be taken to the menu item editor screen.

The last two buttons on this screen (lower right corner) are the Okay and Cancel buttons. These allow you to say 'OK, let's use all these custom button changes and go back to ConfigOpus' main screen,' or 'Oops! Forget all this and let's just leave.'

3.16 MENUS/EDITOR

At the top of the menu item edit screen is an example of what the current menu item looks like. The palettes on either side of the example allow you to select the Foreground and Background colors of the menu item.

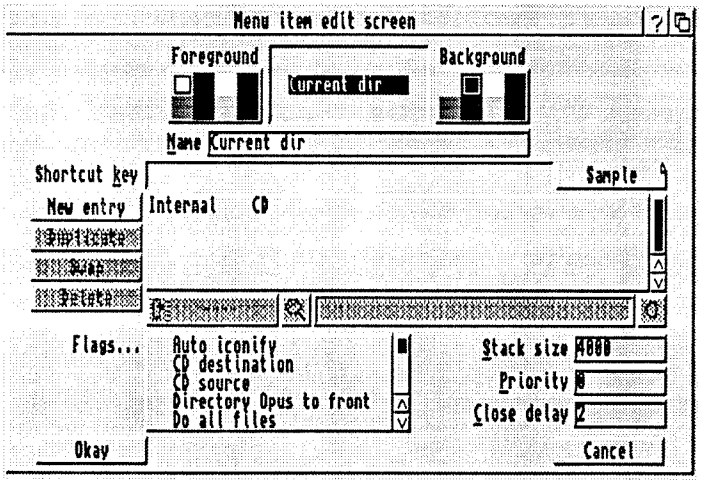

Figure 3.25: Menus/Editor

Name

This field allows you to modify the name that appears on the menu item, and is limited to 63 characters.

Shortcut key is a display area for the Shortcut key sequence. The Shortcut key gives the keyboard equivalent for this menu item.

Sample

To modify the Shortcut key, select this button: any keys, along

with any qualifiers you press will be inserted in the Shortcut key field. To turn off Sample mode, select the Sample button again.


To clear the Shortcut key sequence simply select the Sample button with the right mouse button.

The remaining part of this requester is described in the Command Editor section (see page 88).

3.17 OPERATIONS

The Operation Configuration Screen gives you complete control over the way each of the listed operations is performed. You can cover any contingency by specifying precisely how you want each operation to be carried out in any situation that might arise. When you select one of the buttons to the left of the screen, requesters will appear, asking you how to proceed.

Copy

This button allows you to check the amount of space available for copying files and directories, and to set the archive bits once the copying is finished. You are able to choose which parts of the file information need to be copied. You are also given options to consider if the file being copied already exists in the destination.

Date format

With this button you may choose between the Amiga system's default or three other formats for displaying the date.

Delete

Without the reminders that this button summons, you might lose some very valuable information.

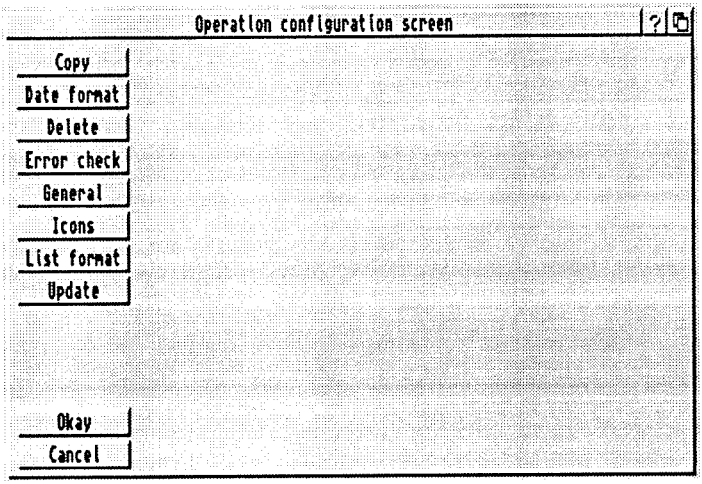

Figure 3.26: Operations

Error check

You may choose either DOS requesters or the more comprehensive Directory Opus error requesters when you select this button.

General

This button displays miscellaneous flags that are detailed in the Operations/General subsection.

Icons

With this button you may decide whether to create a drawer icon, whether each file's icon undergoes every change that befalls its file, and whether an icon is selected automatically with its file.

List Format

The screen summoned by this button gives you control of the format of the lists in both screens.

Update

This button offers various features that are detailed in the Operations/Update subsection.

3.17.1 OPERATIONS/COPY

When copying files and directories...

Check destination's free space before starting

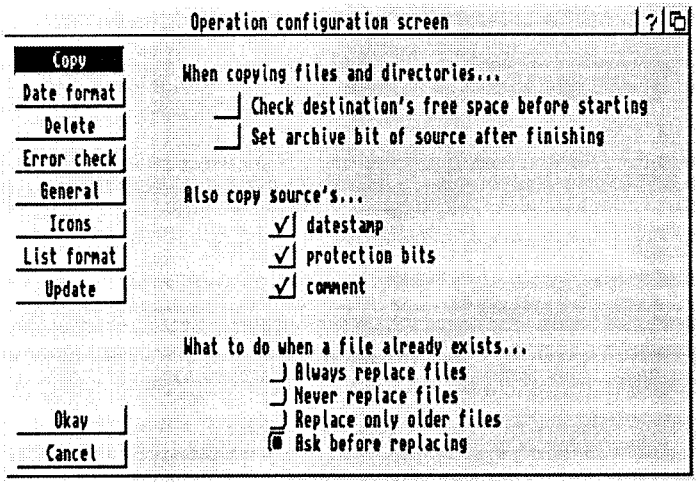

Figure 3.27: Operations/Copy

This causes Directory Opus to check that the destination directory has enough free space in it before attempting to copy selected files and/or directories. If Directory Opus does think that all selected entries will fit in the destination directory, it will alert you with a requester, and give you the option of abandoning the copy process. If any selected directories do not have sizes

displayed for them, Directory Opus will ask you if you wish it to perform a byte count on these directories before seeing whether or not they will fit.

☑ **Set archive bit of source after finishing** After Directory Opus copies a file, the original file will get its archive bit set to true. This means that the file has been archived.

Also copy source's. . .

These flags tell Directory Opus that when it copies a file, it must also copy these parts of the file information as well. You can have all, none, or any combination of these selected.

☑ **Datestamp**

☑ **Protection bits**

☑ **Comment**

What to do when a file already exists. . .

◉ These flags tell Directory Opus what to do if a file that is being copied already exists in the destination. Only one of these may be selected. You have four options:

Always replace files Replaces the original file regardless.

Never replace files Will never replace the original file.

Replace only older files Replaces the original file only if the file to be copied has a more recent last modification date.

Ask before replacing Asks the user what to do if the file already exists.

3.17.2 OPERATIONS/DATE FORMAT

Date Format...

This tells Directory Opus how to format dates. The Amiga's operating system uses the first (DD-MMM-YY) by default, but you may choose whichever one you prefer.

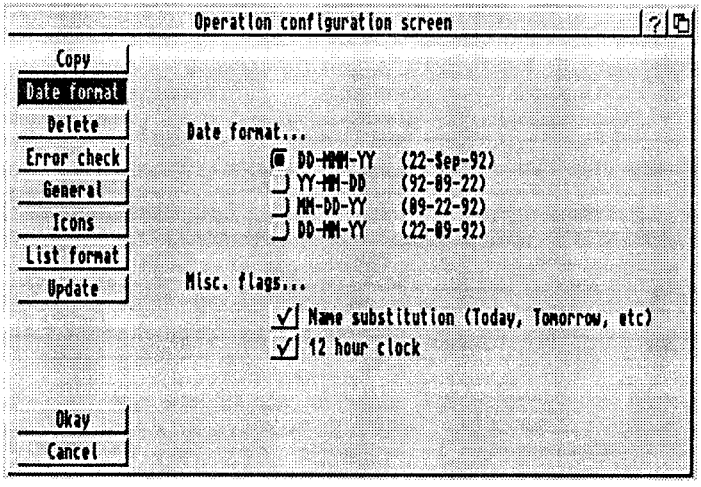

Figure 3.28: Operations/Date Format

DD-MMM-YY (22-Sep-92)

YY-MM-DD (92-09-22)

MM-DD-YY (09-22-92)

DD-MM-YY (22-09-92)

Misc. flags...

☑ **Name substitution (Today, Tomorrow, etc)**

Causes words like Today, Tomorrow or Tuesday to be substituted for a date, if appropriate.

☑ **12 hour clock**

This allows you to choose whether the clock format used by Directory Opus be displayed in 12 hour or 24 hour format.

3.17.3 OPERATIONS/DELETE

Ask before...

☑ **Commencing delete**

Tells Directory Opus to check with you before actually starting to delete. This is a rather useful everyday option because it's always better to be safe than sorry.

☑ **Deleting files**

Tells Directory Opus to ask for confirmation before deleting every file.

☑ **Deleting non-empty directories**

Tells Directory Opus to ask for confirmation before deleting directories that have items in them.

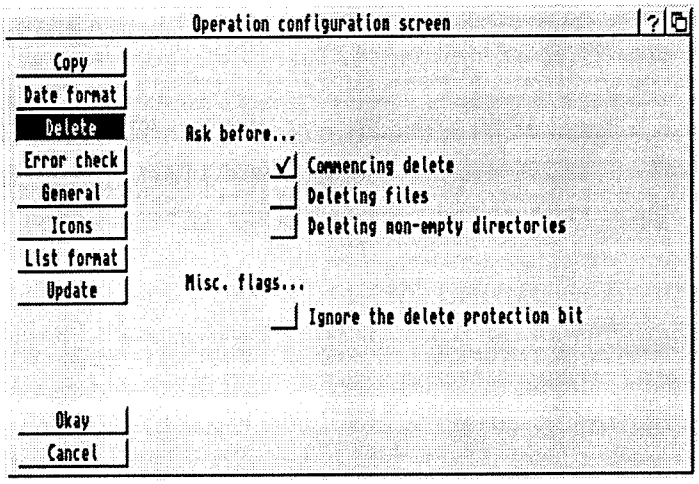

Figure 3.29: Operations/Delete

Misc. flags...

☑ Ignore the delete protection bit

When deleting, this sets the Delete protection bit on all files that do not have it set. This means that the files protected against deletion will be deleted anyway.

3.17.4 OPERATIONS/ERROR CHECK

Enable...

☑ DOS requesters

Disables AmigaDOS requesters that would appear on the Directory Opus screen (disk is write protected, etc.). Instead, you

will only see an error message appear in Directory Opus' status bar.

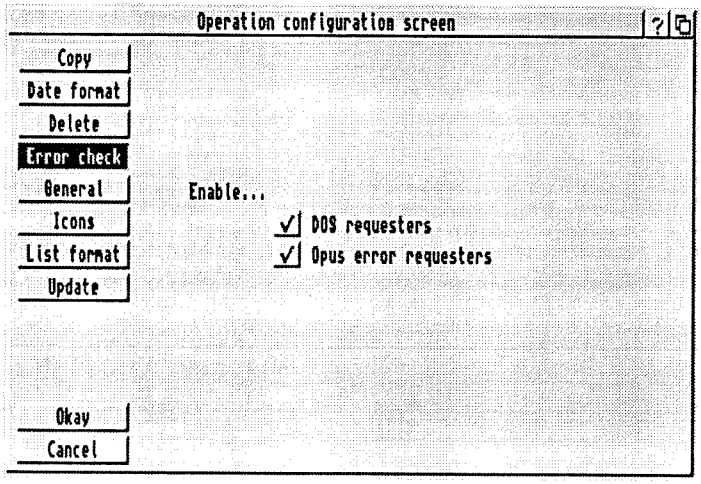

Figure 3.30: Operations/Error Check

☑ Opus error requesters

Causes Directory Opus to open its own error requester when an error is encountered. This requester contains more information than the standard AmigaDOS.

3.17.5 OPERATIONS/GENERAL

Misc. flags...

☑ Click-M-Click drag

Drags files or directories. To drag a file you must (holding down the left mouse button) move the mouse to the left or right of the

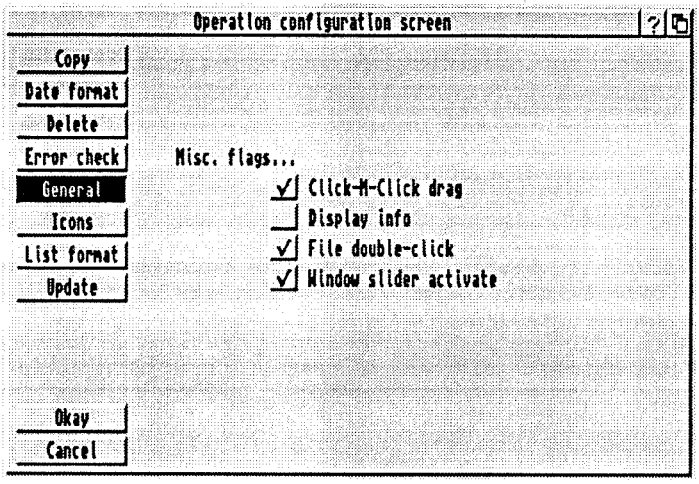

Figure 3.31: Operations/General

file. Do NOT move up or down until the selected file has "lifted" as this would initiate drag-selecting. You can also press the right mouse button while holding down the left to initiate dragging. If you drop the file in the other directory window it will be the same as click-m-clicking. If you drop the file on a custom button at the bottom of the screen that button's action will be performed.

☑ Display info

Makes Directory Opus behave as it did in version 1, by displaying a file's protection bits, datestamp and comment in the status bar. Directory Opus now displays these in the directory windows, and keeps a count of selected files, etc. in the status bar instead. If you prefer the old-style display, turn Display info on. All file information will be displayed in the status bar as you select entries, as well as in the directory windows. With this turned on, you can use the **Byte** Command to view the count of selected entries.

☑ File double-click

Directory Opus will interrogate a file that is double-clicked on with the left mouse button, and will perform the default action for that Filetype.

You may define your own Filetypes, and specify what action is to be performed on them. However, there are also several hard-coded Filetypes. These are IFF ILBM pictures and brushes, icons, fonts, IFF 8SVX sounds, and executable programs.

If a file does not match any of the user-defined Filetypes, it is checked to see if it is one of the hard-coded types. In that case, the default action for these Filetypes is performed. If a file does not match any Filetypes, user-defined or otherwise, it is treated as a text file, and loaded into the text viewer. If any ANSI control sequences are detected, these will be displayed correctly. If the file contains any non-text characters, it will be read in hexadecimal format.

☑ Window slider activate

Causes a directory window to become active whenever its sliders or arrow buttons are selected. If this option is not enabled, selecting the sliders or arrow buttons of a directory will not cause it to become active.

3.17.6 OPERATIONS/ICONS

Misc. flags...

☑ Create icons with directories

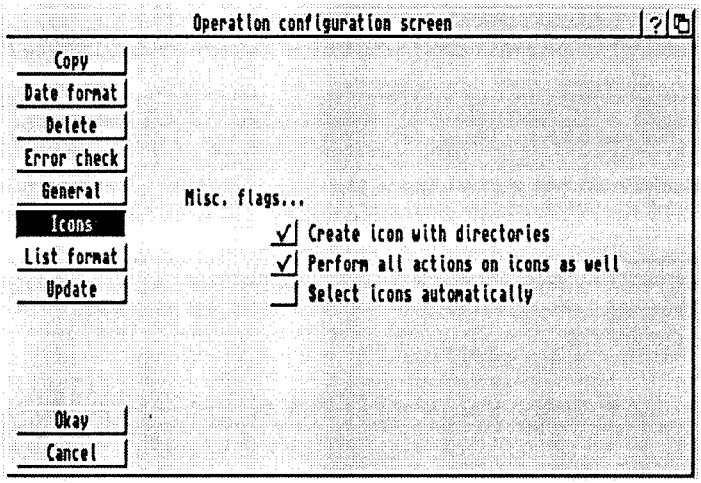

Figure 3.32: Operations/Icons

Creates a drawer icon whenever you create a directory with the **MakeDir** Command.

☑ Perform all actions on icons as well

Whatever happens to a file will happen to that file's icon as well. For instance, if you delete the file DATA, DATA.info (if it exists) will be deleted too. If you rename the file FROG to BUFFALO, then FROG.info (if it exists) will be renamed as BUFFALO.info automatically.

☑ Select icons automatically

Has similar results to Perform all actions on icons as well, except that whenever you select a file, its associated .info file (if it exists) is also selected.

3.17.7 OPERATIONS/LIST FORMAT

Format of the list in the

This cycle button specifies which directory window you are modifying. This allows you to have different list formats in each window, but it does require you to specify the format for each window. If you change the format (using the buttons described below), and you want it to be the same for both windows, don't forget to click on this button and make the corresponding changes to the other window.

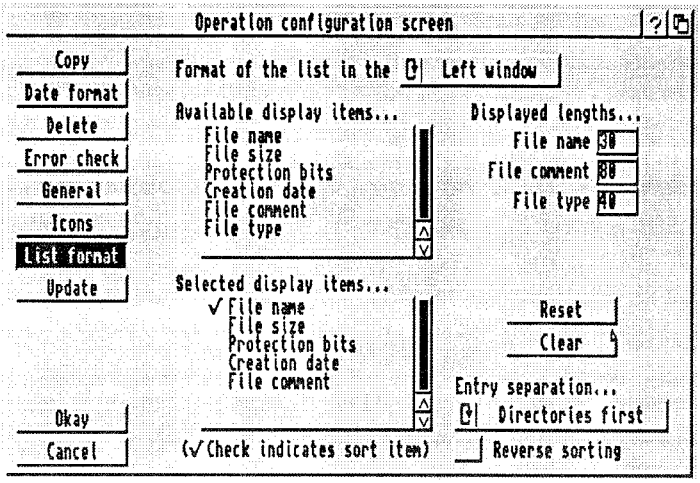

Figure 3.33: Operations/List Format

- **Left window**
- **Right window**

Available display items...

This list allows you to add display items to the **Selected display items...** list (Figure 3.33 on page 138) by clicking on them. An

item can only be added once, so if you click on an item that is already in the list, nothing will happen. The available display items are:

- File name
- File size
- Protection bits
- Creation date
- File comment
- File type

Displayed lengths

The fields allow you to adjust the width of some of these items. The adjustable lengths are:

File name

File comment

File type

Selected display items...

When you select an item from the list of **Available display items...** (Figure 3.33 on page 138) it appears here. This allows you to specify exactly what to display. When you select an item in this list, it places a check mark next to it. This indicates which item to sort the window list on. Clicking on the checked item unchecks it. If no items are checked then the directory window will not be sorted.

Reset

This button causes the list to reset to the factory default settings.

Clear

The left mouse button clears all entries, allowing you to start all over again.

The right mouse button clears the last entry in the list, allowing you to correct a mistake.

Entry Separation...

This cycle button allows you to specify how directories and files are sorted.

- Directories first
- Files first
- Mix files/directories

Reverse Sorting

This specifies that the list should sort in reverse order. For instance instead of **A** coming before **B**, **B** would come before **A**.

3.17.8 OPERATIONS/UPDATE

When processing files...

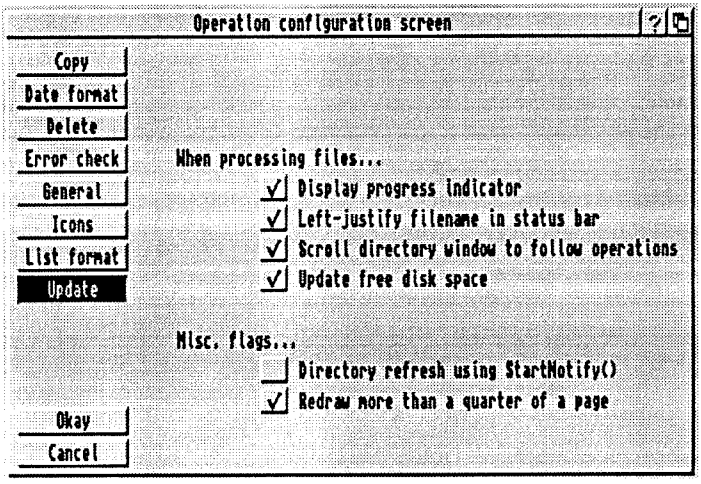

Figure 3.34: Operations/Update

☑ Display progress indicator

Makes Directory Opus open a progress indicator window when it is processing files.

☑ Left-justify filename in status bar

Left-justifies filenames in the status bar rather than centering them as usual.

☑ Scroll directory window to follow operation

Causes the directory windows to scroll, ensuring the visibility of the current entry being acted upon in an operation. The display is returned to its original position when the operation is complete.

☑ Update free disk space

Causes the free space display in the disk name display at the top of the active directory window to be updated whenever a file is

copied to or deleted from the directory. Otherwise, the display is only updated at the end of a complete operation.

Misc. flags. . .

☑ Directory refresh using StartNotify() [2]

Sets up automatic notification of the currently visible directories. If the directories are modified outside Directory Opus (i.e., by another program), Directory Opus will automatically re-read the directory to reflect the changes.

WARNING: While this feature works in many circumstances, the current versions of 2.xx and 3.xx of the operating system have a bug preventing this from being completely useful. Currently the operating system does not notify us to all events, but we have left this option in place so than when the bug is fixed Directory Opus will be ready.

☑ Redraw more than a quarter of a page

Causes Directory Opus to redraw the window if you move more than a quarter of a page in either direction. This flag can greatly speed up movement through directories. Ordinarily, if you use the sliders to move a directory less than a complete page in either direction, the directory window will scroll one file at a time. It is only if you move a complete page or more at once that the window will actually be redrawn. Although not as pleasing aesthetically, redrawing is considerably faster, especially in high-resolution modes.

[2] only under OS2.0/OS3.0

3.18 SCREEN

The Screen configuration screen allows you to modify the appearance of the screen to suit your needs and preferences. Requesters concerning the screen's features will appear when you click on one of the buttons at the left side of the screen.

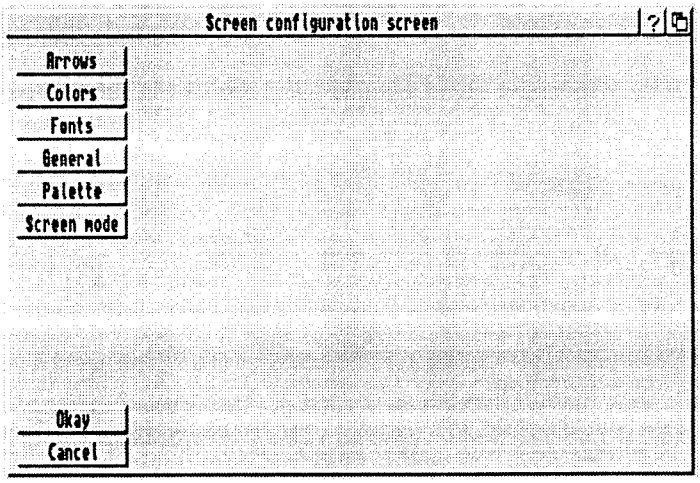

Figure 3.35: Screen

Arrows

Allows you to specify where the arrows on the sliders used by Directory Opus will appear.

Colors

Allows you to modify the pen colors of almost all the components of the Directory Opus screen.

Fonts

Allows you to specify the fonts used in different parts of

Directory Opus.

General

Allows you to specify various options for the screen's appearance.

Palette

Allows you to set the colors of the Directory Opus screen.

Screen mode

Allows you to specify the mode, size and depth of the Directory Opus screen.

Okay and Cancel

These allow you to say either "OK, let's use the Screen settings and go back to Directory Opus", or "Oops! Forget all that, and let's get out of here!"

3.18.1 SCREEN/ARROWS

This screen allows you to specify where the arrows on the sliders used by Directory Opus will appear. There are three choices for each of the three slider types. You can also adjust the size of the arrows.

The easiest way to learn how to use this is just to play with it. Select the slider type by clicking on it, or cycle through the types by clicking on the cycle button. The selected slider type will be highlighted.

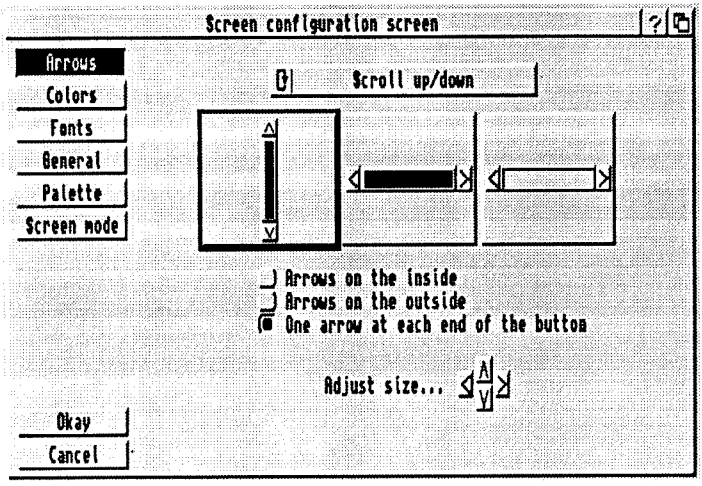

Figure 3.36: Screen/Arrows

Adjust size...

These buttons allow to adjust the width of the arrows in the selected slider type. You can certainly make these huge if you want.

3.18.2 SCREEN/COLORS

This requester allows you to modify the pen colors of almost all the components of the Directory Opus screen. The number of colors to choose from are specified in SCREEN/SCREEN MODE (see page 151) and the color palette is specified in SCREEN/PALETTE (see page 150).

A mock-up of the Directory Opus screen is shown, to allow you to see the results of your color changes. To select the item you want to modify, either click on it in the example box, or click the cycle button at the top of the screen to advance to the next item.

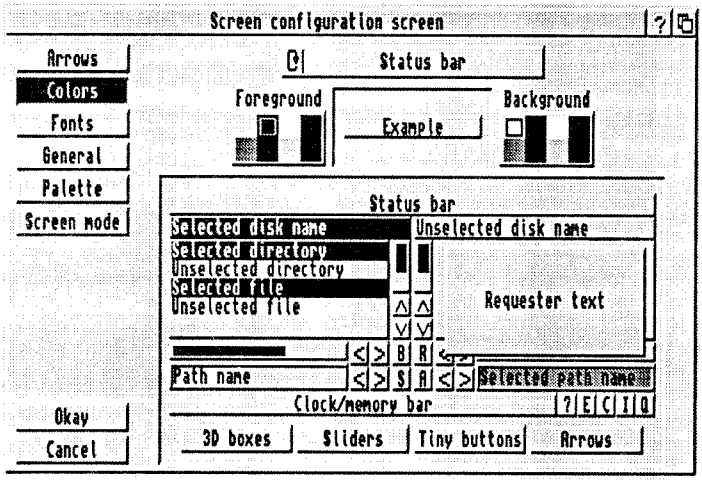

Figure 3.37: Screen/Colors

To modify the colors of an item, use the Foreground and Background palette boxes. Note that under OS1.3, you cannot modify the color of some items.

3.18.3 SCREEN/FONTS

The Screen/Fonts requester allows you to specify the fonts used in different parts of Directory Opus. When this option is selected for the very first time, there will be a delay while Directory Opus looks to see which fonts exist in the system.

Display item. . .

The items for which you may configure a font are displayed in this list. Clicking on an item displays the currently selected font and size in the Font and Size lists. You can then use these lists to select the desired values for the display item.

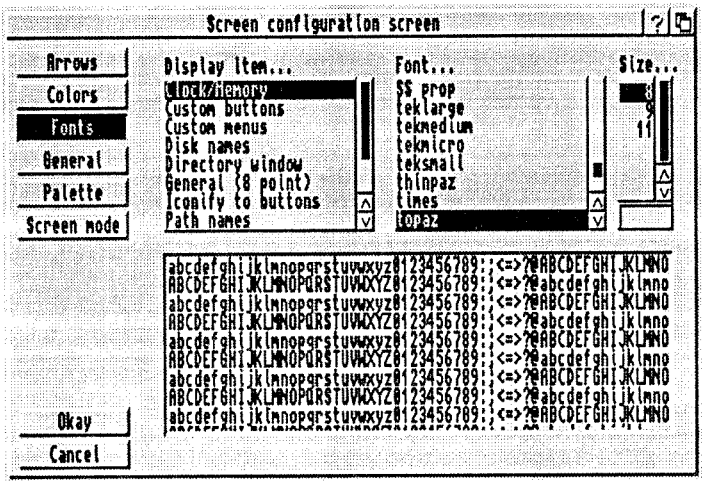

Figure 3.38: Screen/Fonts

Font...

This list contains the list of available fonts. The selected font for the Display item is highlighted. Click on a font name to change the selected font.

Size...

This list contains the sizes available for the currently selected font. The selected size is highlighted. Click on a font size to change the selected size.

An example of the currently selected font and size will be displayed below these lists.

Some Display item5•–5•–ccept non-proportional fonts, and others only accept 8-point fonts. The list of available fonts will change to filter out unacceptable fonts.

Note that under OS1.3 you cannot use a different font in the Pathname field. Setting the Pathname to a larger font will make the Pathname field larger, but under OS1.3 the string fields will still use the general 8 point font.

3.18.4 SCREEN/GENERAL

This requester allows you to specify various options for the screen's appearance.

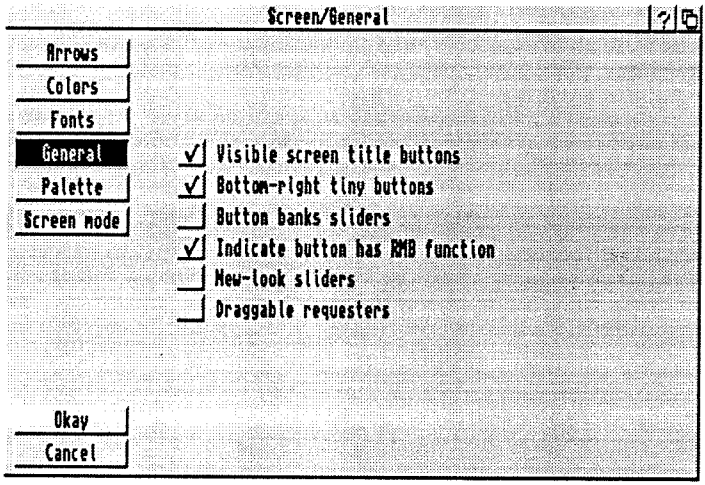

Figure 3.39: Screen/General

☑ Visible screen title buttons

Makes the close and depth buttons of the title bar visible on Directory Opus' screen.

☑ Bottom-right tiny buttons

Makes the tiny buttons in the lower right of Directory Opus' screen visible.

☑ Button banks sliders

Makes sliders appear beside the drive and custom button banks. These make it easier to scroll through the banks.

☑ Indicate button has RMB function

All buttons with Right Mouse button function will have a dog-eared upper right corner.

☑ New-look sliders[3]

The sliders adopt the new style of operating system 2.0.

☑ Draggable requesters

All requesters will have a window title bar allowing you to move it.

3.18.5 SCREEN/PALETTE

The Screen/Palette requester allows you to set the colors of the Directory Opus screen. A palette box is displayed, allowing you to select the color to modify. You can change the color by using the sliders to adjust the red, green and blue components. You can also enter color values directly, using the string fields.

Presets...

There are also several preset color palettes available, which you can select just by clicking. The **Reset** entry will return the colors to the values they had when you entered this section (an Undo

[3] only under OS2.0/OS3.0

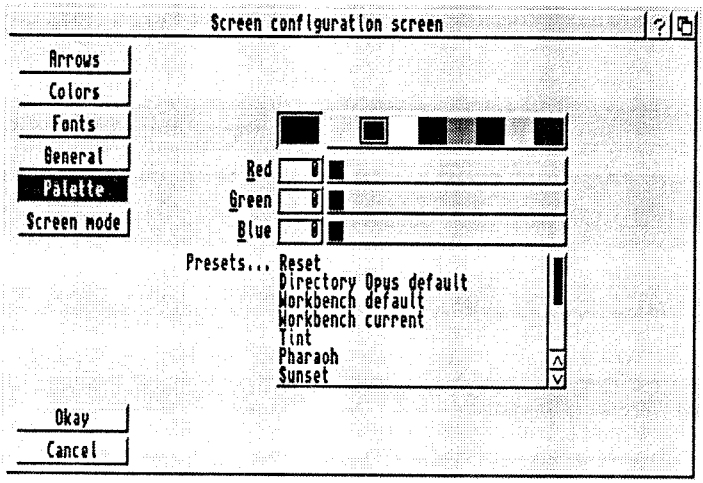

Figure 3.40: Screen/Palette

feature, basically). The **Directory Opus default** entry sets the default Directory Opus palette. The **Workbench default** entry has the default Workbench colors. The **Workbench current** entry is the colors currently set on the Workbench screen. The other presets are different settings which you can experiment with at your own leisure.

3.18.6 SCREEN/SCREEN MODE

The Screen/Screen Mode requester allows you to specify the mode, size and depth of the Directory Opus screen.

Display mode...

Contains a list of the available Display modes. Under OS1.3 this will contain High Res, High Res Laced, Workbench:Use and Workbench:Clone. Under OS2.0 the modes available will vary depending upon the monitors currently active in your system.

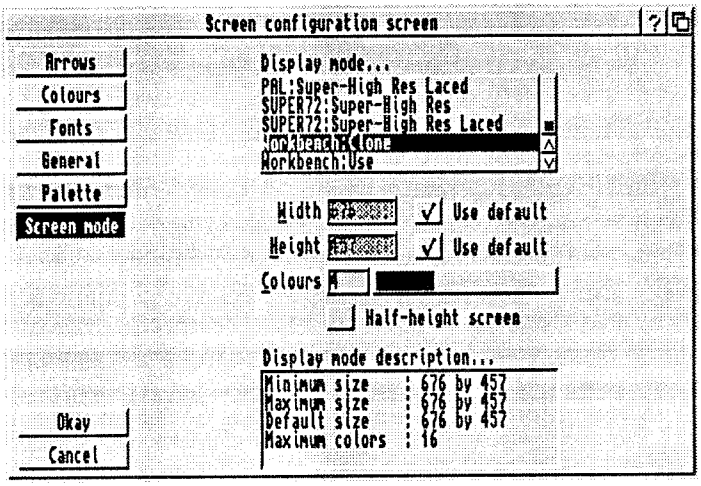

Figure 3.41: Screen/Screen Mode

The characteristics of the selected Display mode are shown at the bottom of the screen.

There are two special items on the list...

Workbench:Use Causes Directory Opus to open a window on the Workbench screen, and not to open a custom screen. The width and height of this window are editable, but the number of colors is fixed at the current Workbench depth.

Workbench:Clone Causes Directory Opus to open a screen in the same mode and exactly the same size as the Workbench screen. The size of this screen is not editable, but you may modify the number of colors.

Width

This field allows you to specify the screen width.

Use default When this button contains a check, the Width field can not be changed. It will, however, display the

default value to be used. When you click on this button, it will allow you to change the value in the Width field.

(When you are using **Workbench:Clone**, the Use default button cannot be deselected.)

Height

This field allows you to specify the screen Height.

☑ **Use default** When this button contains a check, the Height field can not be changed. It will, however, display the default value to be used. When you click on this button, it will allow you to change the value in the Height field.

(When you are using **Workbench:Clone**, the Use default button cannot be deselected.)

Colors

Depending upon the selected screen mode, the number of colors will vary from four to sixteen. There are screen modes with a maximum of less than sixteen; if this is the case, your selection will be limited.

☑ **Half-height screen**

This causes the Directory Opus screen to open at half its normal height, and to be positioned halfway down the display. With an interlaced screen, this would give you a screen the size of a normal (non-interlaced) screen, with the screen behind it still partially visible. Turning on the Half-height screen option will not modify the system's screen height setting.

3.19 SYSTEM

The System configuration screen lets you tell Directory Opus how to run several operations.

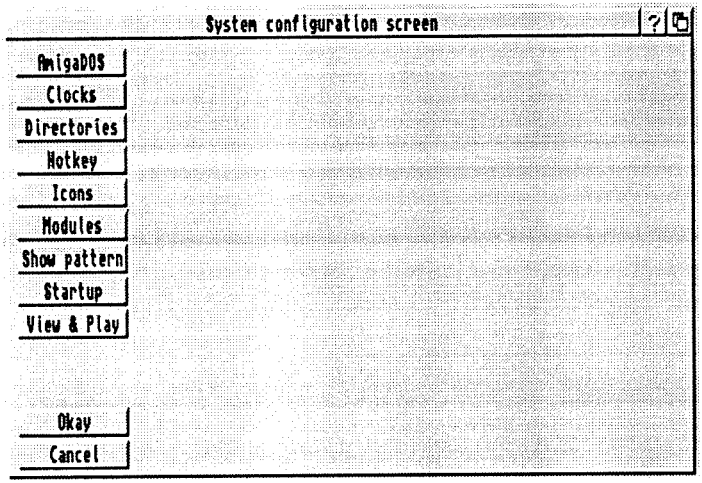

Figure 3.42: System

AmigaDOS

Allows you to specify how Directory Opus starts up AmigaDOS programs.

Clocks

Allows you to specify how Directory Opus displays Clock information.

Directories

Allows you to set various options for Directory operations.

Hotkey

Allows you to specify the Hotkey to activate Directory Opus.

Icons

Allows you to specify the Icons which will be created with the iiicAddIcon command.

Modules

Allows you to specify preloaded external modules.

Show pattern

Allows you to specify files to be shown or excluded from the directory lists.

Startup

Allows you to specify various startup options.

View & Play

Allows you to configure options for Directory Opus' viewers.

Okay and Cancel

These allow you to say either "OK, let's use the Screen settings and go back to Directory Opus", or "Oops! Forget all that, and let's get out of here!"

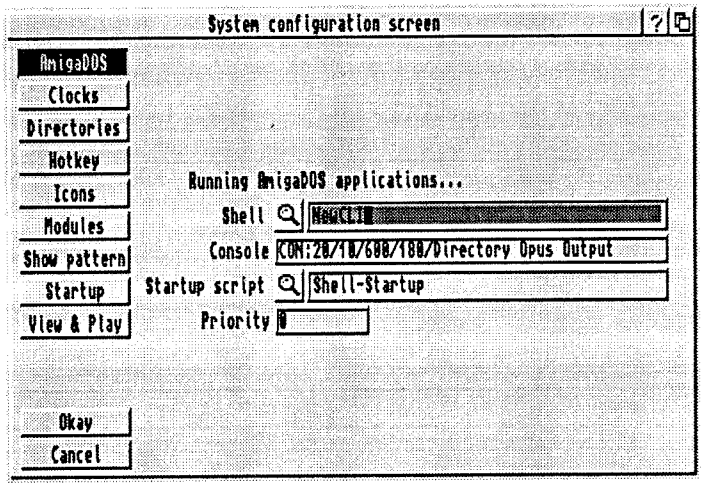

Figure 3.43: System/Amigados

3.19.1 SYSTEM/AMIGADOS

This requester allows you to specify how Directory Opus starts up AmigaDOS programs.

Running AmigaDOS applications...

Shell

This field is the AmigaDOS command used to launch programs that have output windows. It defaults to NewCLI. If you wish to use a shell, you could change the default to NewShell. If you wish to use a WShell, you could change it to NewWsh. The magnifying-glass button to the left of this field opens a requester to help you locate the command you wish to use.

Console

This is the console under which all output windows are opened. It is also used to open a CLI by the NewCLI command, and by the Run command when no files are selected. The console allows you to specify the size, position and title of the output window. It follows the standard AmigaDOS format, namely:

DEVICE:LeftEdge/TopEdge/Width/Height/Title/Options

DEVICE: This is the device you are using. It is normally CON, but under AmigaDOS 1.3 it may be NEWCON (which will open a Shell instead of a CLI).

LeftEdge The left edge of the window (x coordinate, in pixels).

TopEdge The top edge of the window (y coordinate, in pixels).

Width The width of the window in pixels.

Height The height of the window in pixels.

Title The title of the window.

Options Some devices allow you to specify optional parameters (i.e., /Close). The valid parameters vary, so consult the device's documentation for details.

An example is:

CON:20/10/600/180/Directory Opus Output/Close

Start-Up Script

This is an AmigaDOS batch script that will be performed before Directory Opus launches any programs.

Priority

Main priority allows you to specify the priority at which Directory Opus itself runs. The default priority is 0, which means that it peacefully coexists with everything else. Setting the priority higher means that while Directory Opus will not be slowed down much by other tasks, the performance of other tasks will suffer. It is suggested that you set the priority no higher than 10.

3.19.2 SYSTEM/CLOCKS

This Requester allows you to specify how Directory Opus displays Clock information.

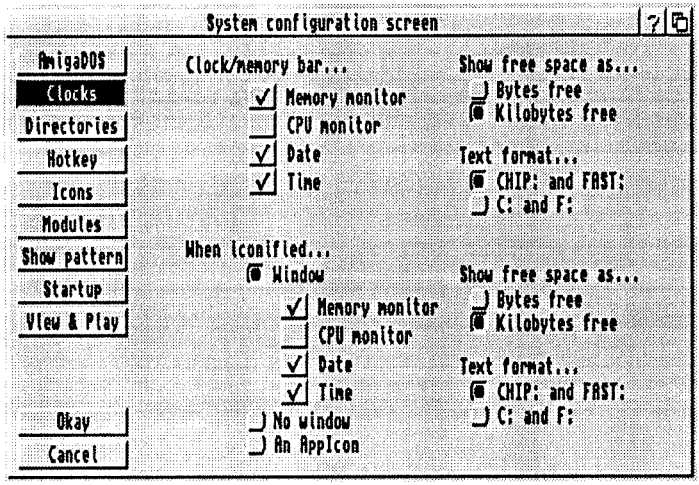

Figure 3.44: System/Clocks

Clock/Memory bar...

These flags tell Directory Opus how to format the clock bar at the

bottom of its screen. You can include any combination of these items, but in some fonts you may not be able to see all the items selected because their space may be limited.

☑ **Memory monitor**

Displays the amount of Chip Memory, Fast Memory and the Total Memory in the clock bar.

☑ **CPU monitor**

Displays the percentage of CPU time that is being used. 100% indicates that there is not any idle time and that at least one process is attempting to run continuously.

☑ **Date**

Displays the current date in the clock bar. [4]

☑ **Time**

Displays the current time in the clock bar.

Show free space as...

◉ There are two groups of buttons with this title. One applies to the Clock/memory bar, and the other applies to the iconified window. They allow you to specify how the Memory Monitor, when enabled, will display memory.

- **Bytes free**
  Displays all digits (this can be hard to read with large values).

[4]The Date Format is specified in OPERATIONS/DATE FORMAT

- **Kilo/Megabytes free**
  Displays value in either kilobytes or megabytes as applicable.

Text format...

There are two groups of buttons with this title. One applies to the Clock/memory bar, and the other applies to the iconified window. They allow you to specify how the titles for the Memory Monitor, when enabled, will be displayed. [5]

The choices are:

- **CHIP: and FAST:**
- **C: and F:**

When iconified

These buttons allow you to specify how Directory Opus should act when Iconified. [6]

Window The following flags allow you to specify what is displayed in the window.

☑ Memory monitor

☑ Date

☑ CPU monitor

[5] See your AmigaDOS documentation for an explanation of Fast and Chip Memory

[6] In addition to the methods described, you can always uniconify Directory Opus by pressing the Uniconify key sequence defined in SYSTEM/HOTKEY.

☑ Time

Directory Opus will iconify to a small window on Workbench. If this window is enabled, its last position is saved when you save the Configuration file.

To un-iconify the window, simply activate it by clicking on it with the left mouse button and then press the right mouse button.

No Window Directory Opus will close all windows.

An AppIcon Directory Opus will open an icon on WorkBench. Simply double click on the icon to uniconify it.

3.19.3 SYSTEM/DIRECTORIES

This requester allows you to set various options for Directory operations.

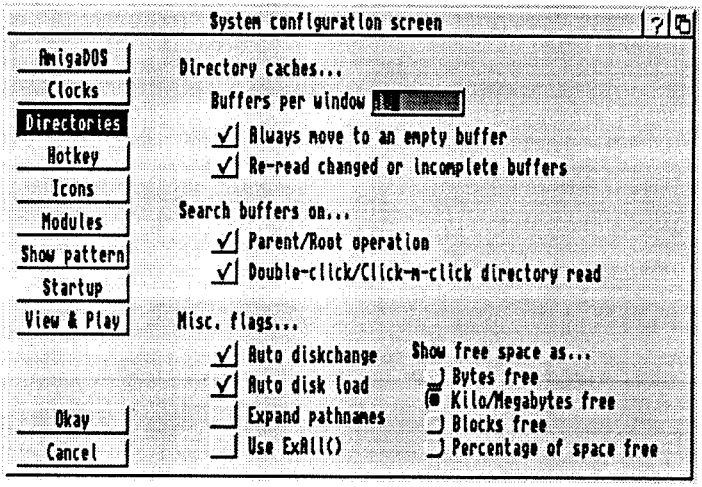

Figure 3.45: System/Directories

Directory caches...

Buffers per window

Allows you to specify the number of buffers to be allocated to each directory window. You can have any value from one to 255.

Always move to empty buffer

Causes Directory Opus to use an empty buffer (if possible) whenever a new directory is read. A new directory is read whenever you enter a path in the directory fields, double-click or click-m-click on a directory name, or select a drive button.

Re-read changed or incomplete buffers

When a buffered directory is shown, this will make Directory Opus check the directory on disk to see if it has changed. If it has, this will cause a re-read of the directory. Also, if the original reading of a directory was interrupted it will be re-read in.

Search buffers on...

Parent/Root operation

This option causes the directory buffers to be searched for the parent or root directory of the current directory, whenever the **Parent** or **Root** Commands are chosen. If the parent or root directory exists in a buffer (i.e., it has already been read), the buffered directory is retrieved from memory. This can save a lot of needless disk access.

☑ Double-click/Click-m-click directory read

When a directory is selected via Double-Click or Click-m-click this will cause Directory Opus to search the buffered directories first.

Misc. flags...

☑ Auto diskchange

Causes Directory Opus to sense disk changes in any drive. If you have the root directory open in either of the directory windows when you change the disk, then the new disk will be read automatically. This allows you to scan through different disks without having to follow the usual selection procedure.

☑ Auto disk load

This option works in a similar fashion to Auto diskchange, except that it detects any disk being inserted, not just those disks currently displayed. If you insert a disk in a floppy drive with this option turned on, you will be asked whether you wish to read the disk into the left or the right window.

☑ Expand pathnames

This flag will cause all paths to be expanded to their full pathnames. For example, if this flag was enabled and you entered S: in the directory field, the name might be expanded to Workbench1.3:s/. It will also do this when you select a drive button. For instance, if you selected a drive button marked DF0:, the actual name of the disk would be inserted in the directory field.

You should note that as the Expand pathnames flag causes the actual volume name of disks to be used, the system (and you) may get confused when using two disks with the same name. As a rule, you should avoid using identically-named disks at the same time.

☑ **Use ExAll( )** [7]

If selected, Directory Opus will use the ExAll( ) function to read directories, rather than the old Examine( )/ExNext( ) combination. ExAll() can be slightly faster than the old method.

WARNING: Operating systems 2.x and 3.x currently have a bug that causes them to skip directory entries when using ExAll( ) on certain devices. Some devices to watch out for include: CDRom, ParNet and Networks. If you have problems, simply turn off this option.

Show free space as. . .

◉ These allow you to choose how to measure the free space that is displayed alongside the disk name.

- **Bytes free**
  Displays the number of free bytes.
- **Kilo/Megabytes free**
  Displays the amount of free space in either kilobytes or megabytes (depending upon the size).
- **Blocks free**
  Displays the number of blocks free (as will also be shown by the AmigaDOS Info command).

[7] only under OS2.0/OS3.0

- **Percentage of space free**
  Displays the free space as a percentage of the total size.

3.19.4 SYSTEM/HOTKEY

This requester allows you to specify the Directory Opus activate Hotkey.

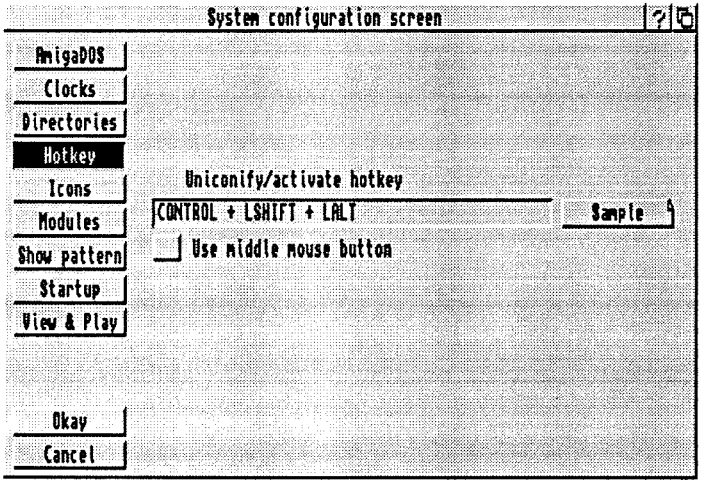

Figure 3.46: System/Hotkey

Uniconify/activate Hotkey

This hotkey is really a combination of keys that you press to do one of four things:

- If Directory Opus is iconified, it will be uniconified.
- If the Directory Opus screen is open, but at the back of the display, it will be brought to the front.

- If the screen has been pulled down using the screen drag bar, it will be brought to the top.
- If the screen is already at the front of the display, it will be moved to the back.

Sample

To modify the Hotkey, select the Sample button at the right of the Hotkey field. While the Sample button is selected, any keys that you press, along with any qualifiers, will be inserted in the hotkey field. To turn off sample mode, select the Sample button again.

To remove the Hotkey simply select the Sample button with the right mouse button.

The default Hotkeys are CONTROL + LSHIFT + LALT, meaning that pressing the control key, the left shift key and the left Alt keys together will perform the action.

Use middle mouse button

With this flag turned on Directory Opus will also use the middle mouse button as well as the Hotkey defined above.

3.19.5 SYSTEM/ICONS

This requester allows you to specify the Icons which will be created with the **AddIcon** Command.

AddIcon's icons...

This list includes three built-in icon types: Drawer, Tool and Project. In addition, each of the filetypes is included in this list. This system allows you to specify your own icons to use with the

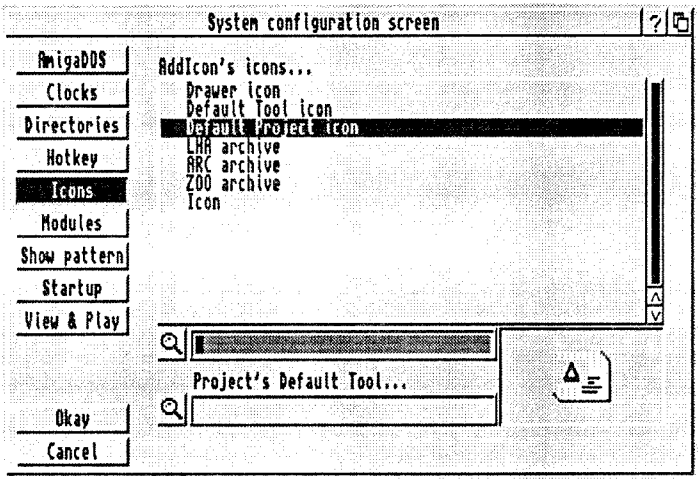

Figure 3.47: System/Icons

AddIcon Command if you do not like the default icons. Simply select the type of icon you want to change (e.g., drawer, tool, etc.), and its current icon will appear in the lower right corner. You may change the icon by modifying the field directly below the list. The magnifying-glass button is provided to allow you to find the icon you want easily.

Under OS1.3 we provide some built-in icon images but under OS2.0 we use the system standard images.

Project's Default Tool...

Allows you to modify the Default Tool of project icons added with the **AddIcon** Command. This only works with the built-in project icon or the default Workbench project icon; not with user-supplied icons. For example, you may set the Default Tool to c:FullView. Then, whenever you double-click from Workbench on an icon that you have added to a project, FullView will be invoked.

3.19.6 SYSTEM/MODULES

An external module is a small program which Directory Opus runs to perform infrequent operations. By making them external, they do not have to be in memory at all times. If you are not concerned with their memory usage and you don't want Directory Opus to load them each time you use them, this requester allows you to specify which modules you want to be preloaded.

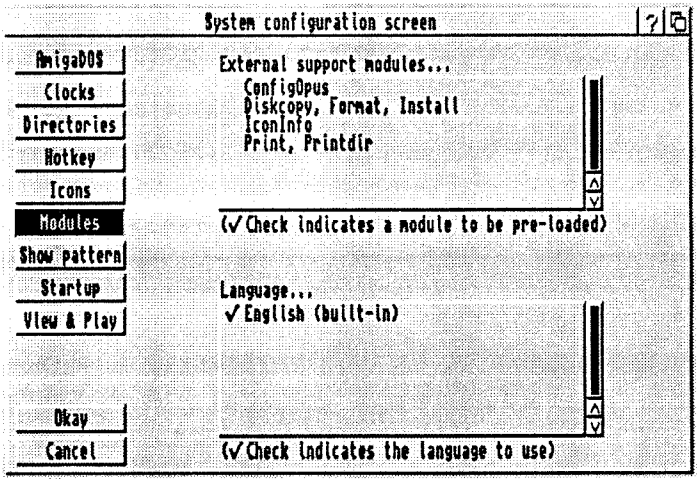

Figure 3.48: System/Modules

External support modules...

Lists all external modules supported by Directory Opus. If you want a module to be preloaded, select it and a checkmark will appear. Select it again if you do not want it preloaded. Preloaded modules take less time to start up but use additional RAM.

Language...

Allows you to select a language for Directory Opus. By default, Directory Opus uses the built-in English language, but as other

languages become available they will appear here. To select one for use, simply click on it.

3.19.7 SYSTEM/SHOW PATTERN

This requester allows you to specify files to show or exclude from the directory lists.

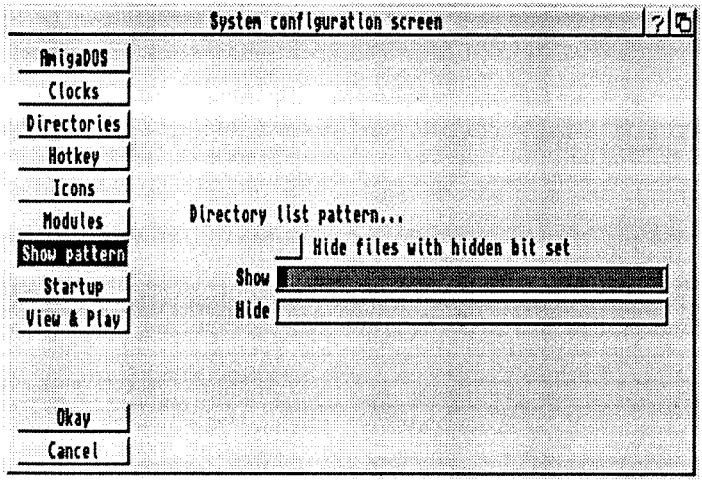

Figure 3.49: System/Showpattern

Directory list pattern...

☑ **Hide files with hidden bit set**

Tells Directory Opus to take notice of the H bit; any files that have this protection bit set will not be displayed.

Show

Allows a wildcard pattern against which all files are checked. If the file matches this pattern, it is displayed.

Hide

Allows a wildcard pattern against which all files are checked. If the file matches this pattern, it is not displayed.

3.19.8 SYSTEM/STARTUP

This requester allows you to specify various startup options.

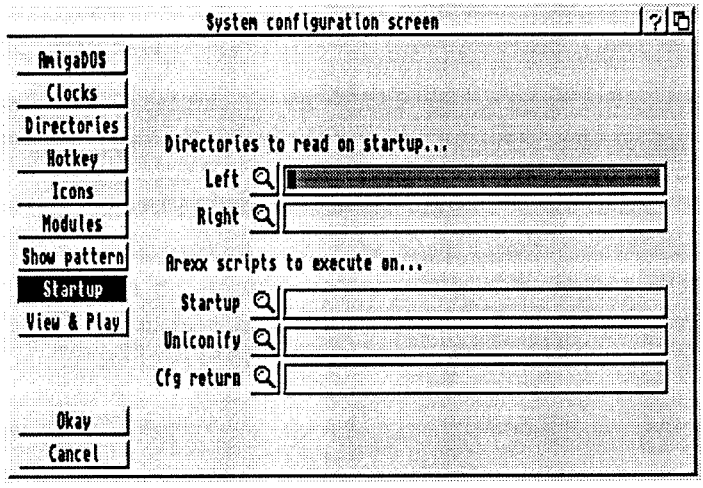

Figure 3.50: System/Startup

Directories to read on startup...

These fields allows you to specify the path name of a directory that is loaded automatically into the left or right window whenever

Directory Opus starts. The magnifying-glass button to the left of these fields opens a directory requester to help you locate the path you require.

Left

This Field specifies the startup directory for the Left window.

Right

This Field specifies the startup directory for the Right window.

ARexx scripts to execute on. . .

These allow you to specify the names of two ARexx scripts that are automatically run when Directory Opus starts (Startup), or when it returns from an iconified state (Uniconify).

Startup

This ARexx script is executed when Directory Opus first starts up.

Uniconify

The Uniconify ARexx script is executed when Directory Opus is uniconified.

Cfg return

This ARexx script is executed when Directory Opus returns from the ConfigOpus utility.

3.19.9 SYSTEM/VIEW & PLAY

This requester allows you to configure options for Directory Opus' viewers.

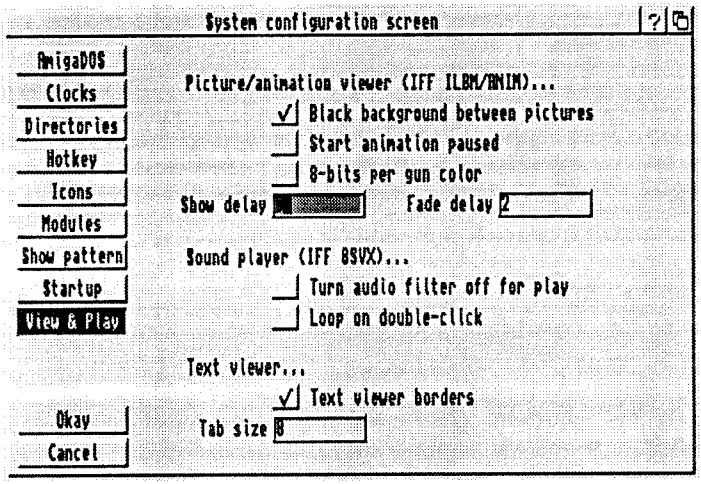

Figure 3.51: System/View & Play

Picture/animation viewer (IFF ILBM/ANIM)

☑ Black background between pictures

This flag causes a blank screen to be displayed while each picture is loading. Otherwise, the Directory Opus screen will be displayed in between each picture.

☑ Start animation paused

Makes animations come up paused. This is particularly useful when an animation is simply a collection of pictures and not really an animation.

☑ 8-bits per gun color[8]

This tells Directory Opus to treat the color information in IFF ILBMs as having 8 bits per gun. Pictures that have 4 bits per gun, pre-AGA, are shown a little dimmer than usual.

☐ Show delay

Allows you to show pictures as a simple slideshow. If set to 0, pictures will wait for you to press the left mouse button before continuing. On other settings, the pictures will continue to display one after the other for the specified time in seconds.

☐ Fade delay

Allows you to configure the speed at which pictures fade in and out. If it is set to 0, there will be no fading at all. The default is 2; the maximum is 10. The value, by the way, indicates in 60ths (50ths on PAL) of a second the time between each change in color.

Sound player (IFF 8SVX)...

☑ Turn audio filter off for play

Causes the high-pass audio filter to be turned off while each IFF 8SVX sound file is played. Note that on A1000s the audio filter may not be software-selectable.

☑ Loop on double-click

Tells Directory Opus to behave as it did in version 3.4. In that version, when you double clicked on sounds, they played in a looped fashion. Under version 4.0 the sound is played only once.

[8]Only when using OS3.0 and AGA

Text viewer...

☑ Text viewer borders

Tells the Text viewer to have a slider and fancier borders. Note that with the fancier borders and sliders you may not be able to read text files that are 80 characters unless you use a smaller font or wider screen.

☑ Tab size

Allows you to specify the number of spaces to which a tab character is equivalent in the **AnsiRead** Command. The default is 8, which is also the ANSI standard.

Chapter 4

Commands

This chapter lists the Directory Opus commands in alphabetical order. Let's begin with an example of an entry's layout, with its component parts explained as they occur:

Command ARexx only

The name of the command comes first. If it is followed by the words 'ARexx only' it means that it is a command that can only be called from another program by using the ARexx intercommunication language. Otherwise, a command can be used both in a button (menu, hotkey or filtype) and as an ARexx command.

Command This is the syntax of the command. Elements in square brackets are optional.

Below the syntax, its elements are explained in turn.

If necessary, there will follow notes pertaining to the use of the particular command.

All Directory Opus commands when called from ARexx respond

with two variables: RC (the error status code) and Result. RC will always be an integer value indicating the severity of an error, 0 being no error. Result will indicate success or failure of the command, usually expressed as an integer value, or it may be data, such as the name of a file.

AbortPrint

AbortPrint

Abort the current print process.

ARexx Results:

RC = 0
Result = 0 nothing was printing,
1 something was printing but is now aborted

About

About

This command displays copyright information and mailing addresses.

ARexx Results:

RC = 0
Result = undefined

AddCustEntry

ARexx Only

AddCustEntry text [userdata] [fgpen] [bgpen] [selectable] [show] [before]

text is the text for the entry, and may be up to 256 characters long. The dimensions of the horizontal slider are adjusted automatically to the longest entry.

userdata optional; user data for this entry. This is not displayed but can be retrieved, defaults to nothing.

fgpen optional; pen color to display entry in or -1 for default file color, defaults to -1.

bgpen optional; background color of entry or -1 for default file background color, defaults to -1.

selectable optional; indicates if entry is selectable (0 = not selectable or 1 = selectable), defaults to 1.

show optional; boolean to update directory display (0 = do not update, 1 = update), defaults to 0.

before optional; ordinal entry number before which to add this entry. A 0 adds the entry to the beginning of the list; -1 adds at the end of the list, defaults to -1.

This is similar to the AddFile command, but allows for entries of up to 256 characters. Entries added with this command are completely user-defined.

This command will fail if there are any normal entries already in the active window, so you should always call ClearWin before using it. By default, entries are not sorted in any way; the most recent entry will be shown as the last.

ARexx Results:

RC = 0

Result = 0 or 1 to indicate failure or success.

AddCustHandler

ARexx Only

AddCustHandler portname [win]

portname name of handler's message port

win optional; window to add a custom handler to (-1 = current window, 0 = left or 1 = right) defaults to -1.

This command will attach an ARexx portname to the current directory buffer. When you manipulate the buffer's entries, Directory Opus will pass messages to the specified portname. This command is only used with buffers created using AddCustEntry. Currently the only two actions supported are Double-Click and Click-M-Click. As an example, this system could be used to create a phonebook system or a NetWork manager with Directory Opus.

ARexx Results:

RC = 0

Result = name of old handler's message port (if there was one)

AddFile ARexx Only

AddFile name size type seconds comment protection reserved show

name name of entry

size size in bytes of entry

type type of entry (-1 = file or 1 = dir)

seconds datestamp of file, seconds from Jan 1, 1978 (date that AmigaDOS considers the beginning of time)

comment file comment

protection protection bits in ASCII form. 'RW' would mean Readable and Writable.

reserved reserved, must be 0 for now

show update directory display boolean

This command adds a file or directory entry to the active directory window.

ARexx Results:

RC = 0

Result = 0 or 1 to indicate failure or success.

AddIcon

AddIcon [name]

name optional; name of file

This will add icons to all selected files and directories in the active directory window. Directory Opus automatically senses what sort of file it is (drawer, project, tool, etc.) and will add the appropriate icon. Under OS2.0, however, we use the Workbench default icons. You may specify your own icons in the ConfigOpus if the default icons are not suitable. You may also specify the Default Tool to be used for project icons. If the name is not given, it will use the first available selected item in the current directory window.

ARexx Results:

RC = 0
Result = undefined

Alarm

Alarm

This command will sound an alarm on both or either of the sound channels. You could use this command to warn people that they have selected a dangerous command. For instance, you could modify the DELETE button to read:

```
Command   Alarm
Command   Delete
```

ARexx Results:

RC = 0
Result = undefined

All

All

This command selects all files and directories in the active directory window.

ARexx Results:
RC = 0
Result = undefined

AnsiRead

AnsiRead [name]

name optional; name of file

This command operates in exactly the same way as the READ command, except that it displays ANSI control sequences (colors and typestyles) properly, and also handles tab characters correctly. If the name is not given it will use the first available selected item in the current directory window.

ARexx Results:
RC = 0
Result = undefined

ARexx

ARexx [command]

command optional; command to execute

This allows you to launch an ARexx script or execute any Directory Opus ARexx command from within Directory Opus. If the command is not given a requester will open asking the user for one.

ARexx Results:

RC = 0
Result = undefined

Assign

Assign [name]

name optional; name to assign

This command allows you to assign a logical device name to the currently active directory. The `C:` device is an example of a logical device; it is not a physical disk itself, but is a directory that has been assigned the name `C:`. For example, a simple name such as `Text:` can be assigned to a long and complicated pathname such as:

```
DH0:Projects/Work/1989/April/Bits&Pieces/Text
```

From then on, whenever you enter `Text:` into a directory button, you will go straight to the directory at the end of the assigned path, instead of having to go through all the sub-directories on the way. You can use the assignment in any other program as well. Logical devices always end with a colon.

Assign also allows you to cancel a previously assigned logical device, or reassign a device to another directory. To do this, press the right mouse button over the S (select) button to activate the GetDevices command. Select the assigned device(s) you wish to change, and invoke the Assign command. A requester will appear, into which you can type the new pathname to be assigned each device name. If you enter a blank string, that logical device will be unassigned.

/dangerparWarning, if multiple logical devices are selected then they will be unassigned as well and you will be prompted for a new path name for each selected device.

ARexx Results:

RC = 0
Result = undefined

Beep

Beep

This command will sound a beep on both or either of the sound channels. You could use this command to signify that a command has finished. For instance, you could modify the Diskcopy option to read:

```
Command   DiskCopy
Command   Beep
```

In versions of the operating system later than 38 (2.1) this command will call the system beep routine. Under earlier versions it will sound its own beep.

ARexx Results:

RC = 0
Result = undefined

BufferList

BufferList [win]

win optional; window to scan the buffer list into (-1 = current window, 0 = left or 1 = right) defaults to -1.

This command displays a list of all the buffered directories. You may then double-click on one of the displayed buffers to jump to that buffer immediately, rather than clicking the arrows to cycle through the buffers one by one.

ARexx Results:
RC = 0
Result = undefined

Busy ARexx Only

Busy [on|off]

on|off optional; adjust the busy pointer, defaults to on.

This command turns on or off the busy pointer.

ARexx Results:
RC = 0
Result = undefined

ButtonIconify

ButtonIconify [rows] [bank]

rows optional; number of button rows. This can be 1, 2, 3 or 6. Defaults to whatever Directory Opus is set to.

bank optional; bank number to display initially, defaults to 1.

This causes Directory Opus to open a window on Workbench containing the Custom Buttons. For more details see page 49

ARexx Results:

RC = 0
Result = undefined

CD

CD [path]

path optional; new current directory path, if path not given a requester will open.

This command allows you to set the current working directory for Directory Opus. This is the directory that is read if you press return on an empty directory button.

ARexx Results:

RC = 0
Result = undefined

CheckAbort

ARexx Only

CheckAbort

This command allows you to monitor, from an ARexx script, whether the user has pressed the left and right mouse buttons together (to abort). It also resets the Abort flag every time it is called. If you plan to use the CheckAbort command, you should call CheckAbort at the start of the script (to reset the Abort flag), before actually using it to check the abort status.

ARexx Results:

RC = 0
Result = 0 if abort sequence has not been hit, 1 if it has

CheckFit

CheckFit [win]

win optional; window to a CheckFit on
(-1 = current window, 0 = left or 1 = right)
defaults to -1.

This works in the same way as **GetSizes** does except that its calculation is based on disk block sizes, and is therefore much more accurate.

The disadvantage to CheckFit is that it needs to recalculate every time the destination's block size changes. Therefore, even if all source directories already have their sizes shown, CheckFit may need to scan them all again. Operations like Copy, etc., that generate directory sizes, are not suitable for CheckFit, and so again CheckFit will need to scan all selected directories.

CheckFit displays the total number of bytes needed and the number of bytes free (i.e., block numbers multiplied by the block size), and the percentage of files that will actually fit. If they will all fit, it shows 100%, if only half of them will fit, it shows 50%.

The GetSizes command is retained as it can be used quickly to provide a rough guide. CheckFit can completely replace the GetSizes command if you don't mind it having to rescan directories when you might not have expected it.

ARexx Results:

RC = 0
Result = undefined

ClearBuffers

ClearBuffers

This command will clear the contents of all the directory buffers other than the two that are currently displayed. All unused memory will be de-allocated. This is a good way to free memory quickly if you have lots of used buffers and are running a bit low on memory.

ARexx Results:

RC = 0
Result = undefined

ClearSizes

ClearSizes [win]

win optional; window to clear the sizes from

(-1 = current window, 0 = left or 1 = right)
defaults to -1.

If a directory's size is already displayed, then the directory will not be re-scanned when selected nor the **GetSizes** command invoked. This could mean that the directory size is inaccurate, especially if any files have been created or deleted in it by another program. Selecting one or more directories, and then selecting the ClearSizes command, will cause the directory sizes to disappear. You can then reselect the directories, invoke the GetSizes command, and the directories will be re-scanned and their updated sizes displayed.

ARexx Results:

RC = 0
Result = undefined

ClearWin

ClearWin [win]

win optional; window to clear
(-1 = current window, 0 = left or 1 = right)
defaults to -1.

This command will clear the contents of a directory window.

ARexx Results:

RC = 0
Result = undefined

Clone

Clone [name] [new name]

name optional; file name

new name optional; new file name

This command allows you to make a copy of selected entries in the same directory, and rename them. A requester will appear for each entry, asking for the new name.

This can also work with a specified file. If a new name is not given then a requester will appear.

ARexx Results:

RC = 0
Result = undefined

Comment

Comment [name] [comment]

name optional; file name

comment optional; comment string

This command allows you to change the comment, or filenote, of the selected files and directories in the active directory window. When directories are selected, you are asked whether the files within them are to have their comments modified also.

For each entry, you are presented with a requester containing its current comment. To change the comment, edit it and press return. To retain the comment, just press return.

To set the comment of all selected entries you should select the All button from the comment requester; choosing Okay or pressing return will set the comments one file at a time.

ARexx Results:

RC = 0
Result = undefined

Configure

Configure

Simply starts up the configuration program.

ARexx Results:

RC = 0
Result = undefined

ContST

ContST

If there is a song paused, then it starts it playing again.

ARexx Results:

RC = 0
Result = undefined

Copy

Copy [name] [destination]

name optional; file name

destination optional; destination directory

This command copies all selected files and directories in the active directory window to the inactive directory window, keeping the same names. If a filename is given then only that file will be copied, but you may also specify a destination directory.

ARexx Results:

RC = 0
Result = undefined

CopyAs

CopyAs [name] [new name]

name optional; file name

new name optional; new file name

This command copies all selected files and directories in the active directory window to the inactive directory window, and allows you to specify a new name for each file or directory. If a filename is given, then only that file will be copied.

ARexx Results:

RC = 0
Result = undefined

CopyWindow

CopyWindow [win]

win optional; window to copy from
(-1 = current window, 0 = left or 1 = right)
defaults to -1.

This will copy the contents of a specified window or the current window into the other window.

ARexx Results:
RC = 0
Result = undefined

DateStamp

DateStamp [name] [date]

name optional; file name

date optional; date to set

This command allows you to change the datestamp of selected files and directories in the active directory window. When directories are selected, you are asked whether you also wish the files within them to have their datestamps modified.

For each entry, you are presented with a requester. If you wish the file to have its datestamp set to the current date and time, simply press return. Otherwise, enter the desired date and time.

To set the datestamp of all selected entries you should select the All button from the datestamp requester; choosing Okay will set the datestamps a file at a time.

ARexx Results:
RC = 0
Result = undefined

Defaults

Defaults

This causes Directory Opus to reset to the hard-coded, default configuration. If you have modified the configuration yourself, and have not yet saved it, you are asked if you wish to save it first.

ARexx Results:
RC = 0
Result = undefined

Delete

Delete [name]

name optional; filename

This will delete all selected entries or just the specified file in the active directory window. Be careful with this, as it is easy to wipe out valuable data.

ARexx Results:
RC = 0
Result = undefined

DirTree

DirTree [win]

win optional; window to create a directory tree in (-1 = current window, 0 = left or 1 = right) defaults to -1.

This command will map out a directory tree in the active directory window, or a specified window.

ARexx Results:

RC = 0
Result = undefined

DiskCopy

DiskCopy [source] destination [verify] [bump]

source optional; source disk

destinations destination disk

verify optional; diskcopy with verify on

bump optional; bump the revision of the disk name, as with Workbench

This command will diskcopy from source to destination, with Verify turned on if the Verify keyword is specified. If no source or destination are specified, the diskcopy requester is opened. If the Bump keyword is given then Directory Opus will bump the revision (Copy_of_DiskName) of the disk name as Workbench does.

ARexx Results:

RC = 0
Result = undefined

DiskCopyBG

DiskCopyBG same as DiskCopy

This command will execute diskcopy in background mode. This allows you to start up a diskcopy and still use Directory Opus.

ARexx Results:
RC = 0
Result = undefined

DiskInfo

DiskInfo [path]

path optional; path of a disk to get info about

This command displays some information about the disk the active directory resides on, including space used and free, datestamp and number of errors on the disk.

ARexx Results:
RC = 0
Result = undefined

DisplayDir

DisplayDir [win]

win optional; window to refresh
(-1 = current window, 0 = left or 1 = right)
defaults to -1.

This command will update the display for the active directory window or the specified window.

ARexx Results:
RC = 0
Result = undefined

DOpusToBack

DOpusToBack

Move the Directory Opus screen to the back

ARexx Results:
RC = 0
Result = undefined

DOpusToFront

DOpusToFront

Move the Directory Opus screen to the front

ARexx Results:

RC = 0
Result = undefined

Encrypt

Encrypt [name] [password]

name optional; file name

password optional; password to use

Have you ever had files that you wanted to encrypt so that only people who knew the password could understand them? This command allows you to do just that. It will encrypt all selected files, using a password that you enter, with a complex algorithm that most people will find impossible to work out. The resulting files are not written over the originals, but are instead written to the destination directory. They will be the same size as the original files, so you can ensure you have enough room in the destination directory.

To decrypt a previously encrypted file, you should enter the same password preceded by a minus sign. For example, to decrypt files you encrypted with the password FOO, select the files, choose the encrypt command and enter -FOO as the password.

ARexx Results:

RC = 0
Result = undefined

ErrorHelp

ErrorHelp [code]

code optional; error code

This command gives you more information about DOS error codes than is generally available. When you select this option, you are asked for the DOS error code you want help with. Examples of DOS error codes are 123, 205 and 225. You are then presented with a description of that error and information pertaining to the possible cause and cure of the error.

ARexx Results:

RC = 0
Result = undefined

Execute

Execute [filename]

filename optional; name of a file to execute as a batch file

This command will attempt to execute each selected file in turn or to execute a specific file as a batch file. This has the same effect as if you had run the batch file via IconX, or via the Execute command from the CLI.

ARexx Results:

RC = 0
Result = undefined

FileInfo

ARexx Only

FileInfo name [separator] [win]

name name of file to give info for (must be in the active or specified window)

separator optional; character used to separate info

win optional; window to get file from (-1 = current window, 0 = left or 1 = right) defaults to -1.

This command returns information on the entry called name (which must be in the active directory window). The information returned is separated by a separator (default separator is space), and consists of the filename, the size in bytes, the size in blocks (not used yet), the file type (less than 0 indicates a file, greater than 0 indicates a directory), the selected status (boolean), the date (days), the date (seconds), the comment and the protection bits.

The date is returned as a number of days since January 1st, 1978, and as the number of seconds since midnight. The protection bits are returned in the format HSPARWED (with a - character where the bit is not set).

ARexx Results:

RC = 0

Result = information about the file

FinishSection

Internal Only

FinishSection

The FinishSection command forces any preceding programs (AmigaDOS, Workbench, Batch or ARexx) to executing before carrying on to the next command. The next command need not be a Directory Opus command; it is just more likely that it will be.

For instance, to add a beep to the end of the LHArc list filetype command, you would change the command list to read:

```
AmigaDOS  LHArc v {f}
Command   FinishSection
Command   Beep
```

ARexx Results:

RC = 0
Result = undefined

Format

Format [drive] [verify] [quick] [noicons] [ffs] [inter] [cache]

drive optional; drive to format

verify optional; format with no icons

quick optional; quick format

noicons optional; format with no icons

ffs optional; format fast file system

inter optional; format in international mode[1]

cache optional; format in directory caching mode[2]

This allows you to format a new disk. All new disks need to be formatted before the computer can write to them.

The Verify flag allows you to turn Verify on or off. If Verify is turned off, the format process will be much faster, but you will not be made aware of any errors that occur. If you don't completely trust your disks, leave Verify on.

The FFS option in Format allows you to format a disk to use the Fast Filing System of Kickstart 2.0. If you are running a Kickstart less than 2.0, you will still be able to select this option to format a disk. However, reading FFS-formatted disks is impossible without 2.0 or greater.

International mode allows file and directory names to include accented characters. Directory caching mode will decrease the capacity of your disk but the directory reading speed will be much greater.

ARexx Results:

RC = 0

Result = undefined

[1]available in OS2.1

[2]available in OS3.0

FormatBG

FormatBG same as Format

This command will execute format in background mode. This allows you to start up a diskcopy and still use Directory Opus.

ARexx Results:

RC = 0
Result = undefined

GetAll ARexx Only

GetAll [separator] [win]

separator optional; character used to separate names

win optional; window to get names from (-1 = current window, 0 = left or 1 = right) defaults to -1.

This command returns, in the RESULT variable, the names of all entries in the active directory window. If separator is specified, the names are separated by a separator. The default separator is a space. You should be prepared to handle a long string.

ARexx Results:

RC = 0
Result = all the entries

GetDevices

GetDevices [win]

win optional; window to put device list into
(-1 = current window, 0 = left or 1 = right)
defaults to -1.

This command displays a list of all devices, volumes and assigned directories present in the system. You may then read in these devices by double-clicking on them. You can also select entire devices for use with the **Hunt** and **Search** commands.

ARexx Results:

RC = 0
Result = undefined

GetDirs ARexx Only

GetDirs [separator] [win]

separator optional; character used to separate directory names

win optional; window to get directory names from
(-1 = current window, 0 = left or 1 = right)
defaults to -1.

This command returns, in the RESULT variable, the names of all directories in the active directory window. If separator is specified, the names are separated by a separator. The default separator is a space. You should be prepared to handle a long string.

ARexx Results:

RC = 0
Result = all the directory entries

GetEntry

ARexx Only

GetEntry num

num the entry number whose name is to be returned

Returns the file or directory name of entry number "num".

ARexx Results:

RC = 0
Result = name of the specified entry

GetFileType

ARexx Only

GetFileType filename [type]

filename file whose filetype is to be returned

type optional; form of filetype ID (0 = file class description or 1 = file class ID), defaults to 0

Returns the filetype (if recognized) of the specified file.

ARexx Results:

RC = 0
Result = filetype ID of the file if filetype is known

GetFiles

ARexx Only

GetFiles [separator] [win]

separator optional; character used to separate filenames

win optional; window to get filenames from
(-1 = current window, 0 = left or 1 = right)
defaults to -1.

This command returns, in the RESULT variable, the names of all files in the active directory window. If separator is specified, the names are separated by a separator. The default separator is a space. You should be prepared to handle a long string.

ARexx Results:

RC = 0
Result = all the file entries

GetNextSelected

ARexx Only

GetNextSelected [win]

win optional; window to get next selected item from
(-1 = current window, 0 = left or 1 = right)
defaults to -1.

This command returns the name of the first selected entry (in the RESULT variable) in the active directory window. The entry is NOT deselected afterwards so if you don't deselect it yourself, that is the only name you will ever get returned.

ARexx Results:

RC = 0
Result = name of the entry

GetSelectedAll

ARexx Only

GetSelectedAll [separator] [win]

separator optional; character used to separate names

win optional; window to get names from
(-1 = current window, 0 = left or 1 = right)
defaults to -1.

This command returns, in the RESULT variable, the names of all selected entries in the active directory window. If separator is specified, the names are separated by a separator. The default separator is a space. You should be prepared to handle a long string.

ARexx Results:

RC = 0
Result = all the selected entries

GetSelectedDirs

ARexx Only

GetSelectedDirs [separator] [win]

separator optional; character used to separate selected directory names

win optional; window to get selected directory names from

(-1 = current window, 0 = left or 1 = right)
defaults to -1.

This command returns, in the RESULT variable, the names of all selected directories in the active directory window. If separator is specified, the names are separated by a separator. The default separator is a space. You should be prepared to handle a long string.

ARexx Results:

RC = 0
Result = all the selected directory entries

GetSelectedFiles

ARexx Only

GetSelectedFiles [separator] [win]

separator optional; character used to separate selected filenames

win optional; window to get selected filenames from
(-1 = current window, 0 = left or 1 = right)
defaults to -1.

This command returns, in the RESULT variable, the names of all selected files in the active directory window. If separator is specified, the names are separated by a separator. The default separator is a space. You should be prepared to handle a long string.

ARexx Results:

RC = 0
Result = all the selected file entries

GetSizes

GetSizes [win]

win optional; window to work with.
(-1 = current window, 0 = left or 1 = right)
defaults to -1.

This will display in the status bar the same information that is displayed when entries are selected (namely, the number of selected files, directories and bytes, out of the total number of files, directories and bytes), along with one additional piece of information. If the selected entries occupy fewer bytes than the free space on the disk in the opposite directory window, allowing all selected files to fit on the destination disk were they to be copied, a Y will be displayed after the count. Otherwise, an N will be displayed.

If any directories are selected when GetSizes is chosen, the directories that do not already have a size displayed are scanned, and the size of the directories are calculated and displayed.

ARexx Results:

RC = 0
Result = undefined

GetString ARexx Only

GetString [message]

message optional; text for requester

This command will open a requester for the user to enter text into. If a message is given, that is used as the prompt for the requester.

ARexx Results:

RC = 0 success, 1 cancel was pressed
Result = string entered by user

Help

Help

Provided a help file has been loaded in, selecting any button or menu after enabling Help mode with this menu should result in the appearance of some helpful text. To turn Help mode off, select this command again.

ARexx Results:
RC = 0
Result = undefined

HexRead

HexRead [name]

name optional; name of file

This command will read the selected files in the same way as READ, but in hexadecimal format. This allows you to view binary and other files containing non-text characters.

ARexx Results:
RC = 0
Result = undefined

Hunt

Hunt [name] [pattern]

name optional; name of directory to hunt

pattern optional; file pattern to hunt

This command allows you to search all selected directories (and their sub-directories) for a specified file or files. A requester appears asking for the pattern to hunt for. You can use full pattern matching for this search.

If a file matching the pattern is found, you are asked if you wish to enter the directory containing it, or continue the search. If you elect to enter the directory, the directory will be read and then all matching entries will be highlighted.

ARexx Results:

RC = 0
Result = undefined

Iconify

Iconify

This command closes the Directory Opus window and screen, de-allocates as much memory as possible, and then opens a small window on the Workbench screen. This is known as iconifying, and allows you to have Directory Opus constantly available while using the minimum amount of memory.

To determine the initial position of the iconified window, especially if you start up Directory Opus in the iconified state, you should iconify Directory Opus, and position the window as you would like it to appear. Then re-enter Directory Opus, and save the configuration (described later).

To re-enter Directory Opus, simple click the left mouse button in the iconified window, then press the right mouse button. If you wish to quit Directory Opus without going back into it, just click the Close button at the far left of the iconified window.

If you have the configuration option Iconify type (described later) set to hidden, the only way to re-enter Directory Opus is with the hotkeys. These default to CTRL-SHIFT-ALT (the CTRL key, the left SHIFT key and the left ALT key held down simultaneously), although this may be changed in the configuration (explained later).

ARexx Results:

RC = 0
Result = undefined

IconInfo

IconInfo [name]

name optional; name of icon (or file with icon)

This command allows you to modify the characteristics of icons such as Stack size, Default Tool and Tool Types. It operates in a similar fashion to the Information menu item of Workbench.

To use this command you may either select the .info files themselves or the actual files or directories the icons belong to.

A requester will appear when you run this command on a valid icon. The actual appearance of the requester will vary depending on the type of icon, but in all cases the actual icon imagery will be displayed. You can click on this image with the left mouse button to display the alternative imagery (if it exists).

The Remap colors button will swap colors 1 and 2 of the icon displayed. This can be useful to remap icons designed under OS1.3 for usage under OS2.0, since OS2.0 swapped the light and dark colors around.

The information displayed for each icon type is listed below. Once you have made the desired changes to the icon, the Save button will save the changes to disk. The Skip and Cancel buttons will exit without modifying the icon on disk.

Drawer icon For a drawer icon you may edit the drawer's protection bits, comment, and tool types. The date of the last modification of the drawer is also displayed.

Project icon For a project icon you may edit the project's stack size, default tool and tool types. Also displayed are the size of the project in bytes and blocks, and the last modification date.

Tool icon For a tool icon you may edit the tool's protection bits, stack size, comment and tool types. Also displayed are the size of the tool in bytes and blocks, and the last modification date.

Disk icon For a disk icon you may edit the disk's Default Tool. Displayed are the total number of blocks, and the number of blocks used and free. The block size, creation date and disk status are also displayed.

Trashcan icon For a trashcan icon the only information available is the last modification date.

Protection bits (where appropriate) are modified in the same way as with the **Protect** command, except that the Hidden and Pure bits are not accessible.

Tool types (where appropriate) are modified in the same way as from Workbench. To edit an existing Tool type, simply select it, and press return when you have modified it. To create a new Tool type, select the New button. To delete an existing Tool type, select it and then select the Delete button.

ARexx Results:

RC = 0
Result = undefined

Install

Install [drive] [ffs] [noboot]

drive optional; drive to install

ffs optional; install a FFS bootblock

noboot optional; make disk non-bootable

This command allows you to make a disk bootable. All bootable disks have special information written on the first sector of the disk. Without this, the disk will not boot.

ARexx Results:

RC = 0
Result = undefined

InstallBG

InstallBG same as Install

This command will execute Install in background mode. This allows you to start up a diskcopy and still use Directory Opus.

ARexx Results:

RC = 0
Result = undefined

LastSaved

LastSaved

This command causes Directory Opus to read in the last saved configuration file. If you have modified the configuration, and have not yet saved it, you are asked for confirmation before this is done.

ARexx Results:

RC = 0
Result = undefined

LoadConfig

LoadConfig name

name configuration file to load

Loads the specified configuration file.

ARexx Results:

RC = 0
Result = 0 means success, 1 means some sort of failure

LoadStrings

LoadStrings name

name string file to load

ARexx Results:

RC = 0

Result = 0 means success, 1 means some sort of failure

LoopPlay

LoopPlay [name]

name optional; name of file

Unlike the Play command, the LoopPlay command plays a sound file repeatedly, requiring a mouse button click to proceed to the next one.

ARexx Results:

RC = 0

Result = undefined

MakeDir

MakeDir [name]

name optional; name of directory

This command allows you to create a new sub-directory in the active directory window.

ARexx Results:

RC = 0
Result = undefined

Modify

Modify name value

name name of item to modify

value new value

This command allows you to make quick changes in certain configuration options from buttons or menus, rather than going through the configuration program.

The Modify command allows changes to at least two characteristics the name of the option to change, and the new value(s) for it. Most values are actually a combination of flag bits; to determine the desired value, you simply add the values of the flag bits together.

You can also add and subtract bits. To do this, specify each bit (or combination of bits) that you wish to add or subtract, preceded by the appropriate sign (+ for add and - for subtract).

The options you can change are:

BankNumber allows you to specify the bank being shown.

BankNumber num

ButtonRows corresponds to Buttons/Button rows menu.

ButtonRows rows

CopyFlags corresponds to Operation/Copy

CopyFlags flags

Flags...

1 = Keep original datestamp
2 = Keep original protection bits
4 = Keep original comment
8 = Set archive bit
16 = Check destination free space.

DateFormat corresponds to Operation/Date format

DateFormat flags

Flags...

1 = AmigaDOS format (DD-MMM-YY)
2 = International format (YY-MM-DD)
4 = USA format (MM-DD-YY)
8 = English/Australian format (DD-MM-YY)
16 = Name substitution (Today, Tomorrow, etc.)
32 = 12 hour clock

DefaultTool the current Default Tool for project icons created with AddIcon.

DefaultTool toolname

DeleteFlags corresponds to Operation/Delete

DeleteFlags flags

Flags...

1 = Ask before commencing delete
2 = Ask before deleting files
4 = Ask before deleting non-empty directories
8 = Set delete protection bit

DirFlags corresponds to System/Directories.

DirFlags flags

Flags...

1 = Always move to empty buffer
2 = Use ExAll( ) to read directories
4 = Auto diskchange
8 = Auto disk load
16 = Search buffers on Double-click
32 = Re-read changed or incomplete buffers
64 = Search buffers on Parent/Root operation
128 = Expand pathnames

DisplayLength corresponds to Operation/List format's length fields. This sets the lengths of some of the directory list items.

DisplayLength win type length

win is the window to act upon (-1 = current window, 0 = left or 1 = right).

Types...

0 = Name
1 = Comment
2 = Filetype

ErrorFlags corresponds to Operation/Error check

ErrorFlags flags

Flags...
1 = Disable DOS requesters
2 = Display error requester

FadeDelay corresponds to System/View & Play's fade delay.

FadeDelay time

Filter Adjust the global filename filter. If filepattern is not given then global filter will be turned off.

Filter [filepattern]

Font corresponds to Screen/Fonts. This takes three values; the ID number of the font to change, the name of the new font, and the point size of the font.

Font idnumber font size

ID Numbers...
0 = General (8 point font)
1 = Directory window font
2 = Text viewer font
3 = Buttons
4 = Menus
5 = Status bar
6 = Disk names
7 = Clock bar
8 = Requesters
9 = Path fields
10 = Button iconify

GeneralFlags corresponds to Operation/General

GeneralFlags flags

Flags...
- 2 = Display info
- 8 = File double-click
- 64 = Window slider activate
- 128 = Click-m-click drag

HelpFile allows you to specify a help file to load

HelpFile pathname

HidePattern corresponds to System/Show pattern's hide pattern

HidePattern pattern

IconFlags corresponds to Operation/Icons

IconFlags flags

Flags...
- 1 = Create icons with directories
- 2 = Perform all actions on icons as well
- 4 = Select icons automatically

IconifyFlags corresponds to System/Clocks.

IconifyFlags flags

Flags...
- 1 = Memory monitor
- 2 = CPU monitor
- 4 = Date

8 = Time
16 = No window
32 = AppIcon
64 = Memory in bytes
128 = C: and F:

Note that No window and AppIcon are mutually exclusive. If the two flags are specified together the results will be unpredictable.

ListFormat corresponds to Operation/List format. The value for this is not a combination of flags, but instead a list of numbers representing items.

ListFormat win items

win is the window to act upon (-1 = current window, 0 = left or 1 = right).

Items is up to five numbers, which indicate what data is to be displayed and in what order.

Items...
0 = Name
1 = Size
2 = Protection
3 = Datestamp
4 = Comment
5 = Filetype

OutputCmd corresponds to System/AmigaDOS/Output command.

OutputCmd shellcommand

OutputWindow corresponds to System/AmigaDOS/Output window.

OutputWindow consoledefinition

ReplaceFlags corresponds to Operation/Copy

ReplaceFlags flags

Flags...

1 = Always replace files
2 = Never replace files
4 = Replace only older files
8 = Ask before replacing

ScreenMode corresponds to Screen/Screen mode.

ScreenMode mode

Mode values...

1 = Open on Workbench's screen
2 = Open a clone of Workbench's screen

Under OS2.0/OS3.0 other mode ID's are supported. Some common mode ID's are...

32768 = High Res screen
32772 = High Res Laced screen
32800 = Super-High Res screen
32804 = Super-High Res Laced screen
233508 = Productivity screen
233509 = Productivity Laced screen
266240 = A2024 10 Hz screen
299008 = A2024 15 Hz screen

ScrDepth corresponds to Screen/Screen mode's depth. The depth can be 2, 3 or 4 bitplanes. That is the same as four, eight or sixteen colors.

ScrDepth depth

ScreenFlags Corresponds to Screen/Screen mode.

ScreenFlags flags

Flags...

1 = Use default screen width
2 = Use default screen height
4 = Half-height screen

ScrClockFlags corresponds to System/Clocks.

ScrClockFlags flags

Flags...

1 = Memory monitor
2 = CPU monitor
4 = Date
8 = Time
64 = Memory in bytes
128 = C: and F:

ScrH corresponds to Screen/Screen mode's height.

ScrH height

ScrW corresponds to Screen/Screen mode's width.

ScrW width

SeparateMethod corresponds to Operation/List format's cycle button.

SeparateMethod win method

win is the window to act upon (-1 = current window, 0 = left or 1 = right).

Methods...

0 = Mix files and directories
1 = Display directories first
2 = Display files first

ShowDelay corresponds to System/View & Play's show delay.

ShowDelay delay

ShowFreeFlags corresponds to System/Directory flags.

ShowFreeFlags flags

Flags...

1 = Bytes free
2 = Kilo/Megabytes free
4 = Blocks free
8 = Percentage of space free

These bits are mutually exclusive. If more than one is specified at a time the results will be unpredictable.

ShowPatBits corresponds to System/Show pattern's hidden bit flag

ShowPatBits flags

Flags...

1 = Hide files with hidden bit set.

ShowPattern corresponds to System/Show pattern's show pattern

ShowPattern pattern

SortFlags corresponds to Operation/List format's sort flag

SortFlags flags

Flags...

1 = Reverse sort in left window
2 = Reverse sort in right window

SortMethod corresponds to Operation/List format's checked sort item.

SortMethod win item

win is the window to act upon (-1 = current window, 0 = left or 1 = right).

Items...

0 = Name
1 = Size
2 = Protection
3 = Datestamp
4 = Comment
5 = Filetype

UpdateFlags corresponds to Operation/Update

UpdateFlags flags

Flags...

1 = Update free disk space
2 = Scroll directory window to follow operations
4 = Redraw more than a quarter of a page
8 = Directory refresh using StartNotify( )
16 = Left-justify filename in status bar
32 = Display progress indicator

ViewPlayFlags corresponds to System/View & Play.

ViewPlayFlags flags

Flags...

1 = Black background between pictures
2 = Turn audio filter off for 8SVX play
4 = 8-bits per gun color
8 = LoopPlay 8SVX sound files on double-click
16 = Text viewer borders
32 = Start animations paused

WindowSize adjusts the placement of the center bar of Directory Opus's directory windows

WindowSize offset

Any Modify command that change display characteristics (screen size, mode and depth, list format, etc.) will not take effect until you execute the **Redraw** command. For commands that only change directory display characteristics, you use the **DisplayDir** command to update the directory display. Some modifications (like date format, hide and show patterns, etc.) will not take effect until the directory is actually re-read (for instance, when the **Rescan** command is used.).

For example, you could have two buttons configured to switch between a High Res screen with 8 colors, and a High Res Interlaced screen with 4 colors. The definitions for these buttons might be:

```
LACEON - Command Modify ScreenMode 32772
         Command Modify ScrDepth 2
         Command Redraw

LACEOFF - Command Modify ScreenMode 32768
          Command Modify ScrDepth 3
          Command Redraw
```

As another example, you might wish to change the left window to be sorted by Date, but to leave the right window alone. For this, you would have the command:

```
Command Modify SortFlags 0 3
Command Rescan 0
```

ARexx Results:

RC = 0
Result = undefined

Move

Move [name] [destination]

name optional; name of file

destination optional; destination path

This allows you to move all selected files and directories from the active directory window to the inactive window. The file or directory will be copied to the destination, and the original will be deleted.

ARexx Results:

RC = 0
Result = undefined

MoveAs

MoveAs [name] [new name]

name optional; name of file

new name optional; new file name

This command moves all selected files and directories from the active directory window to the inactive window, and allows you to specify a new name for each file or directory. The file or directory will be copied to the destination under the new name, and the original will be deleted.

ARexx Results:

RC = 0

Result = undefined

NewCLI

NewCLI [handle]

handle optional; CLI console handle to use

This command opens a CLI from within Directory Opus, using the output window defined in the configuration. The new CLI will be separate from Directory Opus, so to end the CLI you must type EndCLI, as usual.

ARexx Results:

RC = 0

Result = undefined

NextDrives

NextDrives

This command simply flips to the next bank of drive buttons.

ARexx Results:

RC = 0
Result = undefined

None

None

This command deselects all files and directories in the active directory window.

ARexx Results:

RC = 0
Result = undefined

OtherWindow

OtherWindow

This is the same as pressing the space bar, and activates the inactive directory window.

ARexx Results:

RC = 0
Result = 0 left window is now active, 1 right window is

Parent

Parent [win]

win optional; window to do a parent of
(-1 = current window, 0 = left or 1 = right)
defaults to -1.

This will read the parent directory of the directory open in the active or specified directory window.

There are also two hidden parent buttons; one for each directory window. They are hidden in the outermost borders (on the left of the screen and the right of the screen) of the directory windows.

ARexx Results:

RC = 0
Result = undefined

PatternMatch

PatternMatch pattern string

pattern pattern string

string string to match to pattern

This will return a boolean value to indicate whether the supplied string matches the supplied pattern (which may include full wildcards).

ARexx Results:

RC = 0
Result = 0 if string does not match, 1 if it does

Play

Play [name]

name optional; name of file

This will attempt to play each selected file. If the file is an 8SVX format IFF sampled sound (of the type generated by AudioMaster, for instance), the speed and other information from within the file will be used. The Play routine plays both mono and stereo IFF 8SVX files. When playing a file that is not IFF 8SVX, it will assume a speed of 10000 samples/second.

This command also plays song modules. See PlayST for more information.

Each file is played once only, unlike the LoopPlay command, which plays each file continuously.

If the System/View/Play/Turn audio filter off for play option is selected, the high-pass audio filter will be turned off while a sound is playing.

ARexx Results:

RC = 0
Result = undefined

PlayST

PlayST [name]

name optional; name of file

Play a song module. Directory Opus supports Sound/Noise/ProTracker, Oktalizer and MED modules.

ARexx Results:

RC = 0
Result = undefined

Print

Print [name]

name optional; name of file

The print routine is a full-featured text formatter.

You can have a header and/or a footer printed at the top and bottom of each page if you wish. Both the header and footer can consist of the title, the current date and the page number. You may select any combination of Title, Date and Page number you wish. If none are selected, the header or footer is not printed.

You may also select the typestyle of the header and the footer, by clicking on the buttons that initially say Normal. These allow you to cycle through the various typestyles available: Normal, Bold, Italics, Underline, Doublestrike and Shadow. Note that some printers may not have all these typestyles available.

You may enter a title (to be used in the header and footer) in the Title string field. Leaving this blank will cause the name of the file to be used as a title.

The Output field allows you to specify where you want the output of the print process to go. This defaults to PRT:, but you may choose to have the output sent to a disk file. In this case, simply enter the path and the name of the file you wish to print to.

The cycle button labeled Quality allows you to specify whether all text is printed in draft mode or NLQ mode

The button below that determines whether or not the last page of the file to be printed is followed by a formfeed character, which would cause it to be ejected.

To start the print operation, select the Print button.

This command will print all selected files, one at a time. If you select only one file to print, the print routine will be started up as a separate process, allowing you to continue working with Directory Opus. To cancel this type of print, select the print command again. A requester will appear asking if you want to continue with the print or halt it. This requester will also appear if you attempt to quit Directory Opus while a print operation is in place, as you cannot quit until the print has finished.

To cancel a print job that has not been sent to the printer yet, simply press the left and right mouse buttons in the normal way.

Even if you abort a print, the printer may not actually stop for sometime. This is because most printers have print buffers, some quite large ones, which must empty the data they have stored into the printer first.

ARexx Results:

RC = 0
Result = undefined

PrintDir

PrintDir [win]

win optional; window to scan the buffer list into (-1 = current window, 0 = left or 1 = right) defaults to -1.

This command allows you to print the directory in the active directory window. You are able to choose the information you want to print (file sizes, protection bits, dates and comments). This command is not started as a separate process, unlike the **Print** command. Printdir also allows you to specify the output file to which the directory is to be printed. This can be PRT: (i.e., your printer, the default) or a disk file.

ARexx Results:

RC = 0
Result = undefined

Protect

Protect [name] [bits]

name optional; name of file

bits optional; bits to set on given file

The Protect command now allows you to specify bit-masking. This lets you mask some bits as untouchable. You can selectively add or remove certain bits while leaving other bits in their original state.

To select a mask, click on the characters following the Mask: string. If a mask bit is turned on, its character equivalent will be displayed; otherwise a hyphen (-) character will be displayed.

No mask bits that are turned on will be modified during the protection operation.

ARexx Results:

RC = 0
Result = undefined

Query

Query item [win]

item configuration item to query

win optional; window argument that some items need (-1 = current window, 0 = left or 1 = right)

defaults to -1.

Returns the current value of a specific configuration item. These items are described in the **Modify** command. Any item that can be modified can be queried.

Several of the configuration items need an additional window argument. These are ListFormat, DisplayLength, SortMethod and SeparateMethod.

ARexx Results:

RC = 0
Result = current value of the specified item

Quit

Quit [force]

force optional; do not ask user for confirmation.

This command will exit Directory Opus (providing there is not a print operation happening at the time). If the configuration has been modified and not saved, you are asked if you wish to save it first.

ARexx Results:

RC = 0
Result = undefined

Read

Read [name]

name optional; name of file

This command allows you to read selected files. At the bottom of the screen the name of the file is displayed, along with the current position within the file in lines and as a percentage, and also the total number of lines in the file.

The mouse is used to move through the file. Press the left mouse button to turn scrolling on or off. The direction and speed of the scroll are governed by the position of the mouse pointer in the window. The mouse pointer will become invisible while the text is scrolling.

No scrolling occurs if you have the mouse in the center of the window. To scroll forward, move the mouse down until the text starts to scroll. The further down you move the mouse, the faster the text will scroll. This procedure is reversed for backwards scrolling.

The buttons in the bottom-right of the screen also allow you to move around the file. Most buttons will repeat if held down for any length of time.

The up and down arrows move up and down a line at a time. The U and D buttons move up and down a page at a time. The T and B buttons move to the top and bottom of the file.

You can also use the cursor keys to move up and down. As before, in conjunction with SHIFT these will move a page at a time, and in conjunction with CTRL they will move to the top and bottom of the file.

To jump to a specified line, click on the number showing the current line position (immediately to the left of the word Lines). You can also press the J key. To jump to a specified percentage, click on the number immediately to the left of the % sign.

To search for a string, click on the S button or press the S key. You can use full pattern matching in this search.

To print the current file, click on the P button (or press the P key). To print only the current page, press the C key (this has no button equivalent).

To leave the text viewer, click on the X button, or press either X, Q or ESC.

If more than one file was selected, the next one will be read when you exit. To exit without reading the next file, press the right mouse button.

ARexx Results:

RC = 0
Result = undefined

Redraw

Redraw

This command will completely redraw the Directory Opus screen. The screen is closed, and then re-opened.

ARexx Results:

RC = 0
Result = undefined

Relabel

Relabel [device newname]

device optional; device to relabel

newname new name to give to selected device

This command allows you to change the name (label) of the disk that is open in the active directory window.

ARexx Results:

RC = 0
Result = undefined

Remember

Remember

This command allows you to make temporary changes with the **Modify** command, and then restore all the settings later.

ARexx Results:

RC = 0
Result = undefined

RemoveEntry

RemoveEntry number show

number number of entry to remove

show boolean indicates whether to update display

Removes an entry by ordinal number

ARexx Results:

RC = 0
Result = 0 entry was not found, 1 entry was found and removed

RemoveFile

RemoveFile name show

name of the file entry to remove

show boolean indicates whether to update display

This command removes the specified entry from the active directory window, and returns a boolean indicating whether the entry was found and removed. Show is a boolean value indicating whether you want the display updated or not.

ARexx Results:

RC = 0
Result = 0 entry was not found, 1 entry was found and removed

Rename

Rename [name newname]

name optional; name of file

newname newname to give to selected file

This allows you to rename all selected files and directories. You may use a * to specify a pattern. If no * is used, you are asked for a new name for each individual file.

To add a prefix or a suffix (or both) to all selected entries, use a * in the second string field. For example, to add HIRES as a prefix and .pic as a suffix to all selected entries, enter HIRES*.pic.

To change a prefix or a suffix (or both) of all selected entries, use a * in the top string field. For example, to change HIRES to MEDRES as a prefix, and .pic to .iff as a suffix, enter HIRES*.pic in the top string field, and MEDRES*.iff in the bottom string field.

ARexx Results:

RC = 0
Result = undefined

Request

Request message

message text to print in the requester

Displays requester with given message text

ARexx Results:

RC = 0
Result = 0 cancel was selected, 1 okay was selected

Rescan

Rescan [win]

win optional; window to rescan

(-1 = current window, 0 = left or 1 = right)
defaults to -1.

This causes the directory in the active window to be re-read. This has exactly the same effect as activating the directory string field and pressing return, without modifying the directory name.

ARexx Results:

RC = 0
Result = undefined

Reselect

Reselect

This command will reselect all entries that were selected before the last operation was initiated.

ARexx Results:

RC = 0
Result = undefined

Restore

Restore

This command is used to restore the previously Remembered settings.

ARexx Results:

RC = 0
Result = undefined

Root

Root [win]

win optional; window to scan the buffer list into (-1 = current window, 0 = left or 1 = right) defaults to -1.

This command reads the root directory of the active or specified directory window.

There are also two hidden root buttons; one for each directory window. They are hidden in the outermost borders (on the left of the screen and the right of the screen) of the directory windows. With the left mouse button, these buttons perform the **Parent** command; when selected with the right mouse button they will load the root directory.

ARexx Results:

RC = 0
Result = undefined

Run

Run [name]

name optional; name of file

This command allows you to run each selected file in turn, providing that file is executable. This will have much the same effect as if you had double-clicked on the file's icon, or run it from the CLI. A requester will appear, asking for arguments (should the program require any).

ARexx Results:

RC = 0
Result = undefined

SaveConfig

SaveConfig [filename]

filename optional; configuration file to save to

This command saves the configuration to the file that it was read from initially.

ARexx Results:

RC = 0
Result = undefined

ScanDir

ScanDir path [win]

path path to read

win optional; window to read directory into
(-1 = current window, 0 = left or 1 = right)
defaults to -1.

This command allows you to read a directory in much the same way as the drive buttons. For example:

Command `ScanDir Dh0:Work`
Would read the `DH0:Work` directory into the current window.

One usage of this allows you to have an entire bank of buttons devoted to "drive buttons". You could then set up a button on every other bank that would, using the **Modify** command, jump to that bank.

ARexx Results:

RC = 0
Result = undefined

ScrollH

ScrollH chars [win]

chars characters to scroll

win optional; window to scroll
(-1 = current window, 0 = left or 1 = right)
defaults to -1

This will scroll the directory window horizontally by the specified number of characters (either positive or negative).

ARexx Results:

RC = 0
Result = new horizontal position

ScrollToShow

ScrollToShow entry [win]

entry entry to make visible. May be either a name or wildcard pattern, or the ordinal number of the desired entry.

win optional; window to scroll

(-1 = current window, 0 = left or 1 = right)
defaults to -1

This command will try to scroll the specified entry into view.

ARexx Results:

RC = 0
Result = 0 if entry was not found, 1 if it was found

ScrollV

ScrollV lines [win]

lines lines to scroll

win optional; window to scroll
(-1 = current window, 0 = left or 1 = right)
defaults to -1

This will scroll the directory window vertically by the specified number of lines (either positive or negative).

ARexx Results:

RC = 0
Result = new vertical position

Search

Search [name] [pattern] [ucnlc] [wild] [byword]

name optional; name of file

pattern optional; text pattern to search for

ucnlc optional; uppercase characters does not equal lowercase characters.

wild optional; pattern contains wildcard characters

byword optional; match only whole words

This command allows you to search all selected files for a specified string, which may contain wildcards. If a match is found, you are told which file contains the string, and you are given the option to enter the text viewer and read the file. If you opt to do this, a search for the string is automatically initiated once the file has been read by the text viewer.

If any directories are selected, all the files within those directories are also searched.

The flags ucnlc, wild and byword are off by default.

ARexx Results:

RC = 0
Result = undefined

Select

Select [pattern] [onlyfiles | onlydirs] [name | date | protection]

pattern optional; pattern to select by

onlyfiles optional; select only files, defaults to both files and directories.

onlydirs optional; select only directories

name optional; match on name, (default)

date optional; match on date

protection optional; match on protection bits

When used in name mode (default) this command allows you to select files and directories in the active directory window using wildcards. All standard wildcards are supported #, ?, |, %, (,), [,], *, ~ and '.

When used in date mode you can select all files and directories created or modified on a certain date only. Simply enter that date (e.g., 28-Mar-90 would select all entries with datestamps set to any time on the 28th of March, 1990). You can also use words like Yesterday, Today, Tuesday, etc. If you can also specify a specific time. Simply enter the time (in the format HH:MM:SS). If you enter a time, however, the datestamp must match that time **exactly** for the entry to be selected. If you enter a time but not a date it will select files created today that match that time.

You can also select a range of dates. For instance, to select all entries created or modified between the 18th of April, 1989 and the 22nd of September, 1991, you would enter:

```
18-Apr-89 > 22-Sep-91
```

The > character does not in this case signify "greater than"; it simply indicates a range of dates. It must have at least one space on either side of it.

To select all files from the beginning of time to, for instance, the 3rd of January, 1990, you would enter:

```
> 3-Jan-91
```

To select all files from, for example, the 12th of July, 1983 to the current date (today) you would enter:

```
12-Jul-83 >
```

You may also specify times when entering date ranges.

If used in protect bit mode you may specify protection bits that are allowed to be set, and bits that are not. If a bit or group of bits is allowed to be set, precede it with a + character. If a bit or group of bits is not allowed to be set, precede it with a - character. The initial + character is assumed. For instance, to select all readable and writable files, you would simply enter:

```
rw
```

Or, you could enter:

```
+rw
```

Both these would have the same effect. To expand upon this, and select all readable and writable files, but not those that have the hidden or archive bits set, you would enter:

```
rw-ha
```

Or, to select files that do not have the bits SPED set, you would enter

```
-sped
```

The Select command is additive; i.e., entries not matching the given pattern are not deselected.

ARexx Results:

RC = 0
Result = undefined

SelectEntry

SelectEntry number

number ordinal number of entry to select

state optional; selection state
(-1 = toggles, 0 = unselect and 1 = select)
defaults to -1.

show optional; boolean indicating whether to update display

Selects an entry in the active window

ARexx Results:

RC = 0
Result = previous select status of entry or -1 if it could not be found

SelectFile

SelectFile name [state] [show]

name name of entry to select

state optional; selection state
(-1 = toggles, 0 = unselect and 1 = select)
defaults to -1.

show optional; boolean indicating whether to update display

This command will select the entry specified by filename, as if it had just been clicked on.

ARexx Results:

RC = 0

Result = previous select status of file or -1 if it could not be found

SetWinTitle

SetWinTitle title [win]

title title string to use

win optional; window to modify
(-1 = current window, 0 = left or 1 = right)
defaults to -1.

Sets title of active or specified window to given string.

ARexx Results:

RC = 0

Result = undefined

Show

Show [name]

name optional; name of file

This command is very versatile indeed. It will display IFF ILBM pictures and brushes, Workbench icons, and fonts.

If the file is an IFF ILBM, it will be displayed using information from the file. Directory Opus will correctly show most pictures and brushes, including overscan, extra halfbrite (EHB), HAM pictures(4096 colors), and the new AGA display modes of the A4000 and A1200, including HAM8 with 262144 colors at once.

If the picture contains color cycling information, pressing TAB will toggle color cycling on or off.

You can press P to print the picture.

Should you wish the mouse pointer to be visible (in order to point at some part of the picture, for instance), press the .(period) key.

The show routine will also display icons. If the file has a .info suffix it is assumed to be an icon. If the icon has an alternatative image, pressing return will toggle between the two.

To display a font, you need to enter the actual font drawer, and show the size file. That is, the file 8, 12, 19, etc., NOT the .font file.

If you press the HELP key while viewing a picture, font, or icon, a screen will appear showing information about what you were looking at. You can also print what you were looking at.

ARexx Results:

RC = 0
Result = undefined

SmartRead

SmartRead [name]

name optional; name of file

This command invokes the text viewer in the same way as the **Read** command. However, it checks the file for binary characters, and if any are found, the file is hexread instead.

ARexx Results:

RC = 0
Result = undefined

Status

Status value [set newvalue]

value indicates the type of status you want

set *newvalue* optional; set the status to the specified value

This command returns the value for the specifed status command (listed below). If the set keyword is specified, this command will set the status to the specified value (if appropriate for that status command). Most status values can be set.

Some status commands require a sub-number. An example of this is status 4, which returns the total number of files in a specified directory window. The directory window may be either 0 (for the left-hand window), 1 (for the right-hand window) or -1 (for the active window). The command would therefore be status 4 0 for the left-hand window, and status 4 1 for the right-hand window.

Status values are as follows:

1 - Current directory This returns or sets the pathname of the current directory.

Status 1 [set path]

2 - Version This returns the current version number (as a string).

Status 2

3 - Active window This returns or sets the active directory window (either 0 or 1).

Status 3 [set win]

4 - Number of files This returns the total number of files in the directory window.

Status 4 win

5 - Number of dirs This returns the total number of directories in the directory window.

Status 5 win

6 - Number of entries This returns the total number of entries in the directory window.

Status 6 win

7 - Number of selected files This returns the total number of selected files in directory window.

Status 7 win

8 - Number of selected dirs This returns the total number of selected directories in the directory window.

Status 8 win

9 - Number of selected entries This returns the total number of selected entries in the directory window.

Status 9 win

10 - Total bytes This returns the total number of bytes (sum of the lengths of all files) in the directory window.

Status 10 win

11 - Total selected bytes This returns the total number of selected bytes (sum of the lengths of all selected files) in the directory window.

Status 11 win

12 - Select pattern This returns or sets the current Select pattern (used in the **Select** command).

Status 12 [type] [set pattern]

Types...
0 = Name, Default
1 = Date
2 = Bits

13 - Directory name This returns the name of the directory that is open in the specified window. Setting this to another pathname will read that directory.

Status 13 win [set path]

14 - Disk name This returns the name of the disk that is open in the directory window. This cannot be set (use the **Relabel** command to change the disk name).

Status 14 win

15 - Disk free bytes This returns the number of bytes free on the disk open in the directory window.

Status 15 win

16 - Disk total bytes This returns the total number of bytes (free+used) on the disk open in the directory window.

Status 16 win

17 - Buffered directory name This returns the pathname of the directory that is stored in a specific directory buffer (from 0 to 255).

Status 17 win buf

18 - Buffered diskname This returns the diskname of the directory stored in a specific directory buffer (from 0 to 255).

Status 18 win buf

19 - Buffered disk free bytes This returns the free space on the disk whose directory is stored in a specific directory buffer (from 0 to 255). This data may no longer reflect the actual free space on that disk.

Status 19 win buf

20 - Buffered disk total bytes This returns the space used on the disk whose directory is stored in a specific directory buffer (from 0 to 255). This data may no longer reflect the space used on that disk.

Status 20 win buf

21 - Buffer displayed This returns the number of the buffer that is displayed in the directory window, a number from 0 to 255. If this is set to a specific buffer that buffer will be the one shown in the directory window.

Status 21 win [set buf]

24 - unused in Directory Opus 4.0

23 - unused in Directory Opus 4.0

24 - Entries per page This returns the number of entries that can be displayed per page.

Status 24

25 - Configuration changed This returns a boolean value to indicate whether the configuration has been changed since it was last saved.

Status 25 [set state]

26 - Okay string This returns the name of the Okay string in requesters. This also allows you to select the text of the positive button for the request and getstring commands.

Status 26 [set string]

27 - Cancel string This returns the name of the Cancel string in requesters. This also allows you to select the text of the negative button for the **Request** and **GetString** commands.

Status 27 [set string]

28 - Currently iconified This returns a value indicating whether Directory Opus is currently iconfied and if so what type of iconification it is. This will return 0 if not iconified, 1 if iconified and 2 if button iconified.

Status 28

29 - (unused in Directory Opus 4.0)

30 - Top text justification This returns a value indicating how the text in the status bar is rendered.

Status 30 [set type]

Types...
0 = left
1 = right
2 = center

30 - (unused in Directory Opus 4.0)

31 - (unused in Directory Opus 4.0)

32 - (unused in Directory Opus 4.0)

33 - (unused in Directory Opus 4.0)

34 - (unused in Directory Opus 4.0)

ARexx Results:

RC = 0
Result = depends on which status was requested)

StopST

StopST

Stops playing a song module.

ARexx Results:
RC = 0
Result = undefined

SwapWindow

SwapWindow

Swaps the contents of the two directory windows.

ARexx Results:
RC = 0
Result = undefined

TechSupport

TechSupport

This command displays some information on how to contact technical support.

ARexx Results:

RC = 0
Result = undefined

Toggle

Toggle

This command reverses the state of all files and directories in the active directory (i.e., all selected entries are deselected, and vice versa).

ARexx Results:

RC = 0
Result = undefined

TopText

TopText text

text text string to display

Displays given string in the status bar.

ARexx Results:
RC = 0
Result = undefined

Uniconify

Uniconify

Returns to main program from iconified state.

ARexx Results:
RC = 0
Result = undefined

User1, User2, User3, User4

User1, User2, User3, User4

These commands will invoke the four user definable commands associated with your filetypes. In the default configuration the command User1 is defined to extract files from archives.

ARexx Results:

RC = 0
Result = undefined

Verify

Verify [text]

text optional; text to display in requester

Displays "Are you sure?" requester. If this is included in a multi-command operation, Cancel will abort the rest of the commands.

ARexx Results:

RC = 0
Result = 0 is cancel, 1 is okay

Version

Version

This command displays version information about Directory Opus, Kickstart, Workbench and your computer.

ARexx Results:

RC = 0
Result = undefined

4.1 Simple Argument Functions

{d} Destination directory window's path

This will insert the path of the destination directory window.

{f} First selected entry (with path)

Insert the path and filename of the first selected entry. The entry will then be deselected.

{F} All selected entries (with paths)

Insert the path and filenames of the all selected entries. The entries will then be deselected.

AmigaDOS has limited the length of the command line to 256 characters. Therefore, if the length of a command line exceeds 256 characters because of a {F} sequence (or for any reason), another command line will be created for the next lot of file entries until there are no more selected file entries.

{fu} First entry (with path, do not deselect)

The same as {f} except that the entry is not deselected.

{Fu} All entries (with paths, do not deselect)

The same as {F} except that the entries are not deselected.

{o} First selected entry (name only)

The same as {f} except that only the name of the first selected entry is used, without the pathname.

{O} All selected entries (names only)

The same as {F} except that only the names of the selected entries are used, without the pathname.

{ou} First entry (name only, do not deselect)

The same as {o} except that the entry is not deselected after it is used.

{Ou} All entries (names only, do not deselect)

The same as {O} except that the entries are not deselected after they are used.

{p} ARexx port/Public screen name

This will insert the name of the rexx port for this invocation of Directory Opus. Every Directory Opus running will have a unique rexx port name. This is the same name used for the public screen name.

{s} Source directory window's path

This will insert the path of the source directory window.

4.2 Requester Functions

`{RdTitle:Default}` Directory requester

Default can be any valid pathname

`{RfTitle:Default}` File requester

Default can be any valid pathname or filename

`{RFTitle:Default}` Multi-select file requester

Default can be any valid pathname or filename

`{RoTitle:Default}` Font requester

Default can be any fontname. A pointsize can be specified with a slash followed by a number (ie. topaz/8)

`{RsTitle:Default}` Requester that asks for some text

Default can be any text.

The Title string is the title that will appear on the requester. You can just omit the Title string, in which case a default title will be used.

The Default string (preceded by a colon) is the default string that is to appear in the requester. This too can be omitted.

The Default string can contain bracketed letters. These tell Directory Opus where to insert additional information into the requester. The supported bracketed letters include...

`[o]` inserts the last referred filename, only the filename

`[f]` inserts the last referred filename, both path and filename

`[s]` inserts the source directory window's path

`[d]` inserts the destination directory window's path

For example, it would be possible to have your own custom rename button with the following call function:

```
c:rename {f} {RsEnter new filename:Foo.[o]}
```

In this case a string requester would open with the title "Enter new filename" and the default of the string field would be "Foo." followed by the selected filename.

Chapter 5

Appendices

5.1 Glossary

Some terms used in this manual that you may need to know are:

Active (directory window) . . . denoted by a highlighted disk name. This is the source directory, the directory that all commands will act upon (see page 12).

Buffers (directory) . . . A Directory Buffer is an internal storage area for directory information. Whenever Directory Opus reads directory information from a disk, it stores the information in a Directory Buffer. By storing the directory information in a buffer, Directory Opus can avoid unnecessarily re-reading the disk (see page 20).

Click-M-Click . . . this is a Directory Opus term. To click-m-click is to click once with the mouse button, move the mouse to another part of the screen, and press the mouse button again, all within the double-click time set from

Preferences (see page 16).

Directory window ... one of the two windows on the screen in which entries are displayed (see page 12).

Double-click ... an Amiga term, this means to click twice with the mouse button (usually the left button, but sometimes the right button) on an object, within a pre-determined time. The time is set from within Preferences, and you should consult your Amiga manual for more information on this.

Drive gadgets ... the 30 gadgets, of which only six are visible at once, that enable you to easily read any directory.

Entry ... either a file or a directory.

File types ... a file type is a set of user-defined tests to see whether a file is of a certain type or not. Actions can then be performed on the file, depending on its type.

Highlight ... highlight is the term given to the changing or reversal of colors, usually when an object is selected.

Inactive (directory window) ... disk name is not highlighted. This is the destination directory, and files may sometimes be written here from the source directory.

Menu ... a pull-down menu is like a list of gadgets. A menu is accessed by moving the mouse to the top of the screen and holding the right mouse button.

Open ... a directory that is currently displayed is said to be open.

Requester ... a requester is usually a box that appears on the screen, soliciting a response. Some requesters require a

string input, others simply ask you to choose yes or no.

RMB Button ... a Directory Opus term. This is a button that is selected with the right mouse button, instead of with the left.

Selected ... entries are selected when they are highlighted. This is usually done by clicking on them with the left mouse button.

5.2 How we start external programs

Directory Opus uses the small (4k) AmigaDOS program, DOpusRT, to help it launch AmigaDOS, WorkBench, Batch and ARexx programs. When you start up an external program a script file is created in the T: directory. This script file contains all the initialization and execution code needed to start up and manage your programs.

The DOpusRT program is called at the end of the script file, and communicates with Directory Opus to inform it when the script file has finished running. This is needed for functions that are not run asynchronously. It also doubles as a wait command, which is needed for the adjustable close delay.

To take advantage of the current directory changing, adjustable stack, and priority you will need the commands CD, Stack and ChangeTaskPri in your computer's C: directory. You will also need the commands Run, Execute, FailAt, NewCLI and EndCLI. These are almost always present in your C: directory anyway, so you should not need to install them.

There are a couple of other points to be aware of. Because external programs are run via a script file, they can be aborted like any other

script file by pressing CTRL-D. This is only possible if you have an output window selected. If the script file is aborted in this fashion, the output window effectively turns into a CLI, so you must type EndCLI to remove the window.

The second point is that if the program was not run asynchronously, Directory Opus will be frozen, waiting for the signal back from the DOpusRT program. If you abort the script file before it reaches DOpusRT, Directory Opus will never unfreeze. You can force Directory Opus to unfreeze by activating the Directory Opus window and pressing SHIFT-ESC (the shift key in conjunction with the escape key). This also allows you to turn any non-asynchronous program into an asynchronous program, while the program is running.

Note that when using NewCLI or NewShell as the Output command, an error such as unknown command will abort the script file as well. You should therefore make sure that all your buttons, filetypes, hotkeys or menus do not try to run nonexistent programs.

5.3 Wild Card Patterns

Directory Opus can use *wild cards* to make some tasks easier. Wild card characters are expressions that tell Directory Opus to match (or not match, in some cases) more than just one character. You may already have encountered pattern matching in the discussion of the **Rename** command.

? This token will match any single character, but does not include none. For example, `FI?E` would match `FILE`, `FINE` and `FIRE` but not `FIE`.

This character tells Directory Opus to match the following expression 0 or more times. For instance, `FRE#D` would match

FRE, FRED, FREDD, FREDDD, etc. "#?" will match anything, just like "*" (see below).

() is in effect an or. (p1|p2|p3) will match on any one of the patterns p1, p2 and p3. For instance, (D*G|C*T) would match DOG, CAT, DIG, COT, etc. You may include as many patterns inside the parentheses as you like, separated by vertical bars (|).

| Use this token within parentheses to separate different expressions you wish to match at the same time. For example, "(ab|cd)" will match any one of the items separated by "|" (here, "ab" and "cd").

% Matches 0 characters. This is always (useful in an instance like "(foo|bar|%)"). Another example: "CA(M|%)P" will match CAMP and CAP.

* Synonym for "#?", an asterisk will match any number of repetitions of any characters, including none. For instance, "F*" will match F, FR, FRE, FRED, FREDD, FREDDY, etc.

~ The tilde can be used to negate an expression. Anything that would normally match, now will not match. For example, "F~(ROG)" will match FROM but not FROG. "~J*" will not match anything starting with J.

[ ] Brackets enclose a list of characters (a "character class") that match any of the characters in the class. "[abc]" tells Directory Opus to match either "a", "b", or "c" in this position.

~ A tilde at the beginning of a character class matches any characters that are not in the class. For example, "[~bc]" says "match any character except "b" or "c".

- A hyphen indicates a character range (only within character classes). The expression "[a-z]" represents all the lowercase letters from "a" to "z".

' (apostrophe or single quote) Use this character to declare the following character to be a literal character, used as it stands instead of as a wildcard. This permits you to use the characters #, ?, |, %, (,), [,], *, ~ and even an apostrophe as themselves rather than as pattern-matching control characters. For instance, YEAR*'% will match YEAR1989%, YEAR1990% but not YEAR1989 or YEAR1990 as would normally be the case, with % meaning match on nothing. Note that this means that you must use '' to match a apostrophe.

Note: "Expression" in the above table means either a single character (ex: "#?"), or an alternation (ex: "#(ab|cd|ef)"), or a character class (ex: "#[a-zA-Z]").

5.4 Shortcut Keys

These are the default shortcut keys. Each of these can be redefined using the ConfigOpus utility.

Command	*Shortcut Key*	
Add Icon	Left Shift Ctrl	I
All	Ctrl	A
Arc Extract	Ctrl	E
Check fit	Ctrl	F
Clone	Ctrl	L
Comment	Left Shift Ctrl	C
Configure	Right Amiga	C
Copy	Ctrl	C
Date Stamp	Ctrl	D
Delete	Left Shift Ctrl	D
Edit	Left Shift Ctrl	E
Get Sizes	Ctrl	G
Hex Read	Ctrl	H
Hunt	Left Shift Ctrl	H
Iconify	Right Amiga	I
Icon Info	Ctrl	I
Make Dir	Ctrl	M
Move	Left Shift Ctrl	M
None	Ctrl	N
Parent	Ctrl	P
Play	Ctrl	Y
Print	Left Shift Ctrl	P
Protect	Ctrl	T
Quit	Right Amiga	Q
Read	Ctrl	R
Rename	Left Shift Ctrl	R
Root	Ctrl	O
Run	Ctrl	U
Search	Left Shift Ctrl	S
Show	Ctrl	S

5.5 Simple Shortcut Keys

Action	*Shortcut Key*
Toggle Help Mode	Help
Active Directory Field	Return Key
Max Right Window	-
Max Left Window	\
Center Windows	=
Change Active Window	spacebar
Scroll Window Up One Line	Up Arrow
Scroll Window Down One Line	Down Arrow
Scroll Window to Top	Shift Up Arrow
Scroll Window to Bottom	Shift Down Arrow
Scroll Window Right One Column	Right Arrow
Scroll Window Left One Colmun	Left Arrow
Scroll Window to Far Right	Shift Right Arrow
Scroll Window to Far Left	Shift Left Arrow

Index

INOVATRONICS

interactive media