

Arc User's Guide

DATAMYSTIC
5 Bond Street
MT WAVERLEY
VICTORIA 3149
AUSTRALIA

Trademarks Mentioned

Brand and product names are trademarks or registered trademarks of their respective holders.

Disclaimer

DATAMYSTIC specifically disclaims all warranties, expressed or implied, including but not limited to any implied warranty of merchantability or fitness for a particular purpose.

Copyright © DataMystic 1993-2009

All rights reserved. No part of the contents of this book may be reproduced or transmitted in any form or by any means without the express written consent of DataMystic.

Contact Information

Address:

Simon Carter
DataMystic
5 Bond Street
Mt Waverley
Victoria 3149
AUSTRALIA

Phone:

+61-3 9913-0595

Fax:

+61-3 8610-1234

Email:

sales@datamystic.com

Web:

<http://www.datamystic.com>

C O N T E N T S

Program Features.....	1	AutoBuild.....	22
<u>Getting Started</u>	<u>2</u>	Calling .BATch files.....	23
System Requirements.....	2	Editing an Option's Batch Commands	23
Installing Arc.....	2	23
Making a Backup.....	2	Editing Menus.....	24
Installation.....	2	Maximum Height.....	24
Quick Start.....	3	Editing Links.....	24
Manual Conventions.....	3	Link Path.....	24
<u>Using Arc</u>	<u>4</u>	Link Number.....	25
Log-in Screen.....	4	Automatic Link.....	25
Main Display.....	4	AutoBuild with Links.....	25
Status Line.....	5	<u>Copy</u>	<u>26</u>
Active Window.....	5	<u>Move</u>	<u>27</u>
Cursor Bar.....	5	<u>Delete</u>	<u>28</u>
Hot Keys.....	6	<u>Move Window</u>	<u>29</u>
Short Cuts.....	6	<u>First Menu Name</u>	<u>30</u>
Edit Fields.....	6	<u>User Accounts</u>	<u>31</u>
Entering Extended Characters.....	6	Edit User.....	32
System Help.....	7	User ID.....	32
User Help.....	7	User Name.....	32
Using a Mouse.....	7	Authority.....	32
Links.....	7	Password.....	32
<u>Arc Commands</u>	<u>8</u>	Allow Exit to DOS.....	32
<u>About</u>	<u>10</u>	Allow Editing.....	33
<u>Change Password</u>	<u>11</u>	Allow Password Change.....	33
<u>Preferences</u>	<u>12</u>	Allow Scheme and Preferences..	33
Date Format.....	12	LOGIN_USER.....	33
Time Format.....	12	<u>Exit</u>	<u>34</u>
Show Day of Week.....	13	<u>Batch File Commands</u>	<u>35</u>
Show Short Cuts.....	13	Definitions.....	35
Swap Mouse Buttons.....	13	Pre-defined Variables and Environment	35
Screen Save.....	13	Variables.....	35
Exit Short Cut.....	14	Editing DOS environment variables	36
Title Box.....	14	36
<u>Scheme</u>	<u>15</u>	Using a Variable's Value.....	36
Shadows.....	16	Sub-strings.....	36
EGA/VGA Palette.....	16	Verify Statement.....	37
Load Colour Scheme.....	17	MustExist Statement.....	37
Save Colour Scheme.....	18	EditField Statement.....	38
<u>Insert</u>	<u>19</u>	Command History.....	39
<u>Edit</u>	<u>20</u>	CommandButtons Statement.....	39
Editing Divider Bars.....	20	RadioButtons Statement.....	40
Title.....	20	CheckBoxes Statement.....	41
Editing Options.....	20	If... then... else... statements.....	42
Title.....	20	<u>BEdit Batch File Editor</u>	<u>43</u>
Short Cut.....	21	Notes on .BATch files.....	44
Available/ Used Short Cuts.....	21	<u>Security / System Files</u>	<u>45</u>
Password.....	22	System Security.....	45
Authority.....	22	Prevent bypassing Autoexec. Bat	45
Pause after execution.....	22	Preventing batch files from being aborted	46
Reset recovery.....	22	46
Prompt for log message.....	22	Arc's Security.....	46
Don't return to Arc.....	22		

Passwords	46
Hidden Items	47
Security and Short Cuts	47
System Files	47
Network Rights	48
Structure Consistency	48
Locking	48
Read-Only Structures	49
<u>ARC.BAT</u>	<u>50</u>
Network Installation	50
RAM drives	51
<u>Command Line Parameters &</u>	
<u>ARC.INI</u>	<u>52</u>
LogFile=logfile name	52
OnNetwork=<True False>	53
PrevShellEnvStr=environment string	
.....	54
PrimaryStructure=[pathname\]number	
.....	54
ReStartInfoEnvStr=environment string	
.....	55
ScreenSaver=command	55
TemporaryBatchFile=pathname[.BAT]	
.....	55
TextEditor=text editor	56
TextEditorFlags=text editor flags.	57
UserToLogin=user ID	57
Skipping the Login Screen.....	57
VideoMode=[Detect VGA MCGA	
EGA CGA].....	58
<u>User Help</u>	<u>59</u>
Editing User Help	59
<u>Export / Import</u>	<u>60</u>
Export.....	60
/1 Detailed Structure.....	61
Export Language	61
/2 Structure Tree.....	62
Import.....	63
<u>Error Messages</u>	<u>64</u>
I/O Errors	64
Arc-Specific Exit Codes	66
Testing Exit Codes.....	67
<u>Quick Reference</u>	<u>68</u>

I N T R O D U C T I O N

The flexibility and power of today's personal computers has led to a multitude of applications for the home and office. Remembering the directory names, program names and DOS commands to start each application is difficult for an inexperienced user, and a needless hassle for the experienced user. A large number of batch files, one or more per application, does not constitute a suitable solution.

Many programs have been written with the intention of addressing this need, but all fall well short of the mark. It is for this reason that Arc was written. It offers speed and productivity to experienced users, and insulates the novice or occasional PC user from the vagaries of DOS.

- Program Features
- Ⓟ Automatic installation, detecting over 360 known applications
 - Ⓟ Uses no memory during execution of applications (Arc is non-resident). Can be used to load/unload TSRs
 - Ⓟ Full point-and-click mouse support (Microsoft or compatible)
 - Ⓟ Simple to create, edit, copy, move and delete menu items
 - Ⓟ AutoBuild facility for adding programs
 - Ⓟ Security using authority levels and/or passwords
 - Ⓟ Hot keys - single key execution (local to each menu window)
 - Ⓟ Short cuts - single key combination execution (global to each menu structure)
 - Ⓟ Comprehensive context-sensitive system help, and user-defined help can be created for each menu item
 - Ⓟ Supports all DOS-based networks (e.g. Novell, LANtastic)
 - Ⓟ Unlimited menu depth and menu items
 - Ⓟ Extensive (and dynamic) colour configuration, attractive presentation
 - Ⓟ Screen saver with adjustable duration, support for third-party or user-written screen savers
 - Ⓟ International time and date displays
 - Ⓟ Usage tracking
 - Ⓟ Export and Import programs for modifying data structures
 - Ⓟ Fully supports EGA 43 and VGA 50 line modes, 132 and 100 column modes, and supports EGA/VGA palette editing
 - Ⓟ Very small amount of disk space used
 - Ⓟ Australian made and supported

Getting Started

This chapter outlines Arc's minimum system requirements, describes how to install Arc, and discusses the conventions used in this manual.

System Requirements

For Arc to run smoothly on your system, it should meet the following criteria:

- ✦ IBM AT/XT/PC or compatible
- ✦ 160k bytes or more RAM
- ✦ A hard disk and a floppy drive
- ✦ MS-DOS 3.1 or higher
- ✦ Microsoft or compatible mouse (Optional)

Installing Arc

Making a Backup Place a write-protect tab on your original Arc disk, and then insert it in drive A: (or the drive you are using). Make a backup of it using the **xcopy** command (see your DOS manual for more information).

```
xcopy a: b: /s /v f
```

Installation Insert your copy of the Arc program disk in drive A: (or the drive you are using) and type the drive ID. For example:

```
A: f
```

Your screen will display the prompt for the A: drive. Then type

```
install f
```

The install program will take you step-by-step through the installation process. Once installed, type

```
arc f
```


to run Arc. At the log-in screen, press f for both the user's ID and password (leave them both blank).

2 Arc is initially set up with a single user having supervisor status.

Quick Start

For information on...	See Page
how to add/modify users	31
how to skip the login screen	57
how to add new applications	19 and 22
how to create options that exit Arc	51

Manual Conventions

8	This icon indicates mouse actions.
	This icon indicates keyboard actions. Keys on the keyboard are also shown pictorially, e.g. f
2	This icon indicates a note.
M	This icon indicates a warning.
Monospace type	This typeface represents text as it appears onscreen or in a program. It is also used for anything you must type.
[information]	Items in square brackets are optional. To include optional information, type only the information within the brackets. Do not type the brackets themselves.
<i>italics</i>	You must supply the text for any item shown in italics. For example, when <i>filename</i> appears, you should type the name of your file.

Using Arc

This chapter describes the user interface of Arc and how to interact with it, as well as introducing concepts specific to Arc.

Log-in Screen

The initial screen is called the log-in screen. It is where you identify yourself so Arc can determine which menus and options you have access to.

Figure 1
Log-In
Dialog



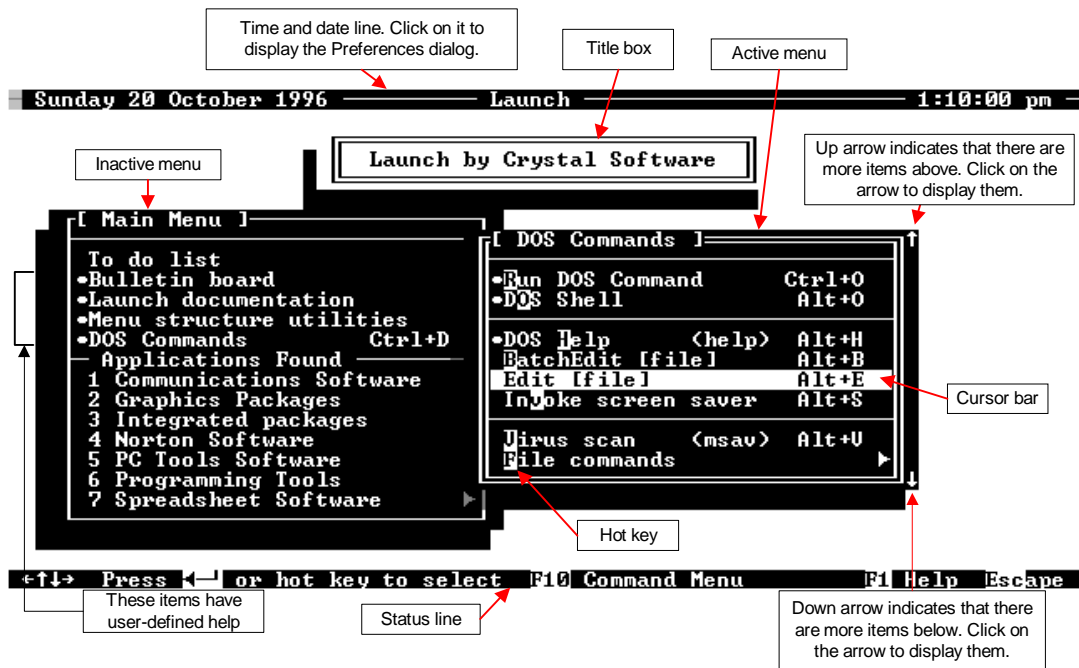
Firstly, type in your user ID and press `f`. Then you can type in your password, and press `f`. If your password is correct, you will then see the main menu displayed.

- 2
 - If project Ids have been set up in the file `PROJECT.TXT` in Arc Menu's directory, an extra field will appear called 'Project'. You must enter a valid Project Id before Arc will let you login.
 - If either the user ID or password you type is incorrect, Arc will display the text "Invalid login". It will not tell you which of the User ID or password is incorrect, because this makes the login process more secure. To clear both fields, press `^` and then enter a new user ID.

Main Display

The main display is where you select an application to run. Its components are shown below.

Figure 2
Main
Display



Status Line The status line displays information about the current action being performed by Arc. It shows available hot keys and commands, and allows mouse users to click on it to initiate actions. For example, pointing and clicking with the mouse on **F1 Help** has the same effect as pressing l on the keyboard.

Active Window Only one of the windows on the screen may be 'active' at any one time. The active window always has a double-line border, while inactive windows have a single-line border. Also, the active window is always in the foreground, and will cover up windows behind it.

Cursor Bar Within the active menu window displayed on the screen, there is a highlighted region called the cursor bar. The cursor bar is used to select the application to be run. To move the cursor bar using the keyboard, use the following keys:



Key	Action
w	Moves up one menu item (jumping over any dividers)
y	Moves down one menu item
f	If the cursor bar is positioned over an <i>option</i> , this runs the option under the cursor. If the cursor bar is positioned over a <i>menu</i> (a menu is indicated by an arrow to its right), this displays the next menu
x	Same as f, except <i>options</i> are not run
^	Returns to the previous menu if one exists. If there is no previous menu, the option to exit is given
z	Same as ^, except does not allow exiting
g	Moves to the first menu item
d	Moves to the last menu item
{	Moves six items up
}	Moves six items down

8 Clicking in the current menu will move the cursor bar to the item being clicked on. Clicking on the item currently under the cursor bar will access that item. If the current menu has more items in it than can fit on the screen, the mouse can be clicked on the up or down arrows in the window's right border to scroll more items into view.

Hot Keys A hot key is a single keypress that provides a quick way of accessing a menu item from within the current menu window. Every menu item may be assigned multiple hot keys, indicated by a single character of the item's title being displayed in a different colour. In Figure 2, the menu item **File commands** has the character “F” as its hot key. Pressing the indicated hot key F results in immediate execution of that option, without having to highlight the item and press f.

Hot keys are local to each menu window, and remain lit only in the current menu window. Hot keys are case sensitive i.e. “Y” is different from “y”. If Arc cannot locate an item with an appropriate hot key, it will try again with the same key but of opposite case.

Short Cuts A short cut is a special key combination that will access an item even if it is not currently visible. In Figure 2, the menu item **Edit [file]** has the short cut **aE**. If the short cut of an option is pressed, execution follows immediately; the cursor bar remains where it is. If the short cut of a menu is pressed, the cursor bar will be placed on the item last selected in that menu (usually the first item) - Arc automatically navigates through the menu structure for you.

Edit Fields While editing, the following keys can be used:



Key	Action
z	Moves the cursor left
x	Moves the cursor right
y	A down arrow shown to the right of a text field means that a history of the values of the field has been recorded. To access the history, press y
^	Discards any changes that have been made
h	Toggles between insert/overwrite mode
c	Removes the character under the cursor
BACKSPACE	Removes the character to the left of the cursor
g	Moves to the start of the field
d	Moves to the end of the field
bz	Moves the cursor one word left
bx	Moves the cursor one word right
b BACKSPACE	Clears the entire field
v	Moves to the next field
jv	Moves to the previous field

Entering Extended Characters If you wish to enter a character that is not normally accessible via the keyboard, first find its decimal equivalent from an ASCII chart. Then, while holding the **a** key down, type in the character's decimal equivalent on the **numeric keypad**, and then release **a**.

System Help Information about the current action is always available by pressing **h**. For instance, when editing the **Short Cut** field of an item, pressing **h** will display help relating to Short Cuts. Within the Help window the following keys may be used:



Key	Action
w	Moves up one line
y	Moves down one line
{	Moves up one page
}	Moves down one page
g	Moves to the start of the help text
d	Moves to the end of the help text
h	Displays an index of available help topics. Use the keys shown in this table (except h of course) to highlight the topic you want to view, and then press o . Press h to remove the help index
^	Removes the Help window

User Help Pressing **bh** will display User-defined Help for the item under the cursor bar, if any is available. A small dot on an item's left-hand side indicates that it has user-defined help. If you have editing privileges and there is no user help, Arc will give you the option of creating it.

Using a Mouse Most operations can be performed with a mouse. Actions listed on the status line may be invoked by pointing and clicking on them with the mouse cursor. For example, clicking on **F2 Save** acts in the same manner as pressing **m** on the keyboard. Moving the mouse pointer into the active window and clicking moves the cursor bar onto the item being clicked on, and clicking on the item under the cursor bar selects that item (the same as pressing **f**).

Pressing the opposite mouse button is the same as pressing **^**. Clicking on the desktop (the background) brings up the Command menu under the mouse cursor.

A mouse can be used to drag dialog boxes to new locations on the screen. Simply place the mouse cursor on the top line of the dialog box, press and hold the mouse button, and then move the mouse. The dialog box will follow the mouse's movements unless that would take the dialog box outside the screen.

Arc can interact with all mice that are Microsoft compatible. The centre button on three-button mice is ignored.

Links A structure is a complete set of menus and options. You may have one structure for when your computer boots (a **primary** structure), and another for when you shell out of an application such as Paradox (a **secondary** structure). Even with two totally independent structures, you may want to share common elements, such as a utility programs menu. A link allows structures to be logically joined. Any changes made to a structure that is linked will be reflected in all structures that have a link to it. Links are also useful for implementing user-specific menus on a network. The primary network menu can include a link to each user's own private menu.

Arc Commands

Pressing **u** pops up the Command menu. The Command menu may also be accessed with a mouse by clicking on **F10 Commands** on the status line, or by clicking on the background (the Command menu pops up near the mouse cursor). The commands available depends on your privileges (as set by the User Accounts command). The commands allow you to alter the structure of the menu, customise Arc and display general information.

Figure 3
 Command
 Menu



The following commands are available to all users:

A	About	Displays general information
H	Change password	Changes the user's password
P	Preferences	Customises Arc
S	Scheme	Modifies Arc's presentation
X	Exit	Exits from Arc

The following commands are available to users who have **Allow Editing** ticked in User Accounts:

E,t	Edit	Edits any item
I,h	Insert	Inserts a new item
D,c	Delete	Removes any item
M	Move current item	Moves the item under the cursor bar
C	Copy current item	Copies the item under the cursor bar
W	Move window	Repositions a menu window
F	First menu name	Changes the name of the initial menu

The following command is only available to users who have their authority set to 99 in User Accounts (supervisors):

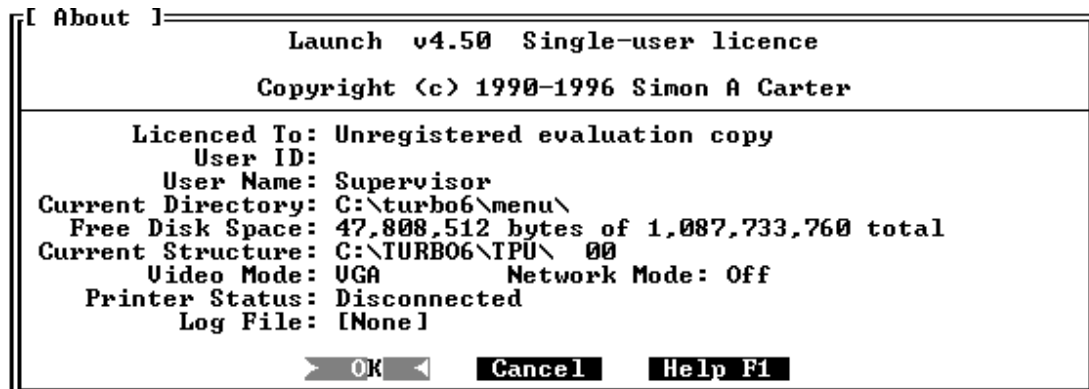
U User accounts Edits user information

About

Gives general system information

About is accessed by pressing **uA**, or by clicking **F10 Commands** on the status line and then selecting **About** in the Command menu.

Figure 4
About



Licenced To	Who owns this copy of Arc
User ID	Current user's ID
User Name	Current user's name
Current Directory	DOS's current drive and directory
Free Disk Space	Disk space used and remaining on the current drive
Current Structure	Where the currently displayed menu window comes from. This is useful when linked menu structures are used
Video Mode	The current video display mode. Possible values are VGA, MCGA, EGA and CGA.
Network Mode	Tells you if Arc is running in network mode or not.
Printer Status	Provides the current status of the printer, from DOS's point of view (so the displayed information may not always be correct)
	Disconnected not connected to the computer
	Turned Off has not been switched on
	Online ready to print
	Offline not ready to print
	Paper Out requires paper
	Busy already printing
	Error error of some description
Log File	Whether actions are being logged to a file or not. Only the supervisor can view this information, so users are unaware of whether their actions are being logged or not.

Press **^** or **f** when you have finished viewing the information (Or click on the **OK** or **Cancel** button).

Change Password

Allows the user to change their password

The Change Password command is accessed by pressing uH or by clicking **F10 Commands** on the status line and then selecting **Change Password** in the Command menu.

Figure 5
Change
Password



Before you can change your password you must be able to confirm your identity by typing in your current password. Once you have entered your current password and pressed f, you can then change your password. You must type your new password in twice, and both passwords must be identical. This confirms that you know what your password is. If your confirmation is identical to the initial password, your new password is then set, and the Command menu re-appears.

If the original password was entered incorrectly, or the confirmation was not identical to the new password, the password will remain unchanged.

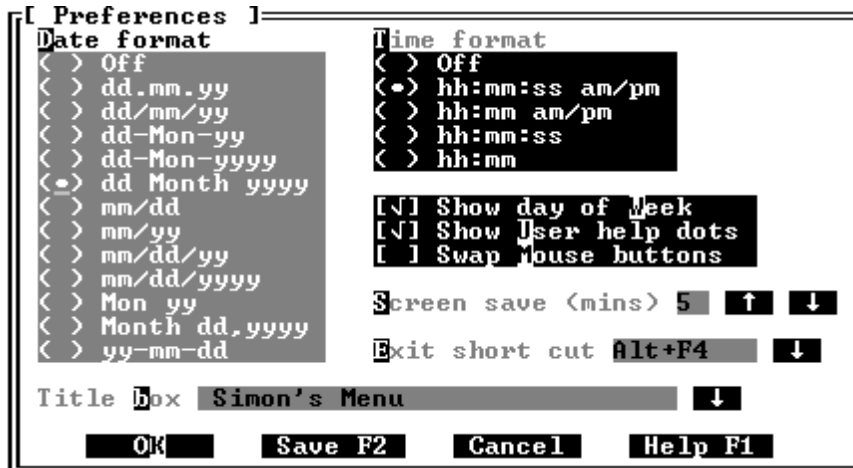
Press ^ at any stage to cancel changing the password.

Preferences

Customises various Arc features

The Preferences window is accessed by pressing **uP** or by clicking **F10 Commands** on the status line and then selecting **Preferences** in the Command menu.

Figure 6
Preferences



The Preferences dialog contains descriptions of the features that can be customised. Once you have finished making changes in the Preferences menu, you may press:



Key	Action
f	Returns to the Command menu, using the current preference settings. These will be used on a temporary basis; when you try to exit Arc you will be prompted to save them
m	Returns to the Command menu, saving any changes
^	Returns to the Command menu, discarding any changes

- 8 If you have a mouse, you can use it to point and click on the feature you would like to change.

Date Format

This controls how the current date is displayed. To prevent the current date being displayed, select **Off**. Use **w**, **y**, **g** and **d** to select a new date format.

Time Format

This controls how the current time is displayed. To prevent the current time being displayed, select **Off**. Use **w**, **y**, **g** and **d** to select a new time format.

Show Day of Week

When ticked, this field enables the day of the week display in the top left hand corner of the screen.



Key	Action
k	Toggles the setting
Y	Ticks (enables) the display
N	Unticks (disables) the display

Show Short Cuts

Arc can display short cuts next to each item in the menu window, or it can place a dot to the left of each item that indicates the presence of a short cut. A menu item's short cut is always shown on the status line when it is highlighted by the cursor bar during normal menu operations.

Use the same keys as for **Day of Week** above.

Swap Mouse Buttons

This controls the function of the mouse buttons. It allows the function normally associated with the buttons to be swapped. This is useful for left-handed users, or those who are used to having the buttons perform opposite functions.

Use the same keys as for **Day of Week** above.

Screen Save

This controls the amount of time Arc will wait before activating the Screen Saver. The Screen Saver blacks out the screen to prevent the phosphor from 'burning in' to a particular image that has been displayed continuously for an extended period of time.

If no key has been pressed and no mouse movement has occurred within the specified time period (1 to 99 minutes), the Screen Save will be activated (On a VGA system the screen will gradually 'fade' out to black). The current time and date are moved around the screen until a key is pressed or the mouse is moved to re-activate the screen.

A new time-out value may be entered directly, or the w and y keys may be used to increase or decrease the current value. Press f to accept the current value, or press ^ to leave it as it was.

A value of 0 disables Screen Save.

- 2 Screen Save only functions while Arc is active, and is turned off during scheme cycling. For information on invoking external screen savers, see page 55.

Exit Short Cut

The exit short cut is a quick way to leave Arc. It is very similar to a normal short cut put on an option or a menu. When the exit short cut is pressed during normal menu operations (if defined, it is displayed beside Exit in the Command menu), Arc will immediately exit, bypassing the normal confirmation window. If any preferences have been changed, you will be prompted to save them before exiting.

Pressing **y** pops up a list of the available/used shortcuts. You can choose a shortcut from this list. See page 21 for more detail.

As with normal short cuts, you will be warned if an item already has been assigned the short cut you press. To blank the exit short cut, press **k**.

Title Box

If not blank, this line of text appears framed in a box near the top of the screen. The title box can contain any appropriate information, such as a company or division name, or a personalised message, like 'Simon's Menu'. To access a history of previously entered Title box lines, press **y**.

Scheme

Alters Arc's presentation (colours, textures and palette)

The Scheme window is accessed by pressing **uS** or by clicking **F10 Commands** on the status line and then selecting **Scheme** in the Command menu.

Figure 7
Scheme



The Scheme dialog contains a list of screen elements whose attributes (colour or texture) may be modified. Using the cursor bar movement keys or the mouse, highlight the element you wish to modify. Pressing **v** or **aU** then allows you to modify the attribute of the element you have chosen. If the attribute you choose to modify is a colour, the right hand side of the dialog displays all possible colour combinations, and you may choose a new colour from amongst these. Elements with texture attributes display all possible textures (characters) on the right hand side of the dialog.



Key	Action
L	Loads a new colour scheme from the scheme file
S	Saves the current colour scheme to the scheme file. This option is only accessible to the supervisor (authority of 99)
P	Alters the VGA/EGA Palette
H	Changes the shadow type
m	Saves the user's scheme and returns to the Command menu
f	Returns to the Command menu using the current scheme settings
^	Discards any changes made and returns to the Command menu
w, y, z, x	Moves the attribute cursor in the attribute window
R	Reverts to the previous attribute value
C	Cycles the selection. This has the same effect as repeatedly pressing the right arrow key, except a delay is inserted between each colour change, and cycling ceases when you press k . This allows you to sit back and potentially view every selection, stopping at the one you like best. 8 Cycling can be invoked by clicking on Cycle on the status line 2 Screen Save is disabled during Cycle

- 8 If you are using a mouse, point and click on the colour or texture you want, or view a range of selections by holding down the mouse button and dragging the mouse cursor across the window.

Shadows

Pressing H or clicking on **Shadow Type** pops up the Shadow Type dialog. By moving the selection cursor with w,y,g and d you can select the way Arc displays shadows.

Figure 8
Shadow Type



Five different techniques are offered:

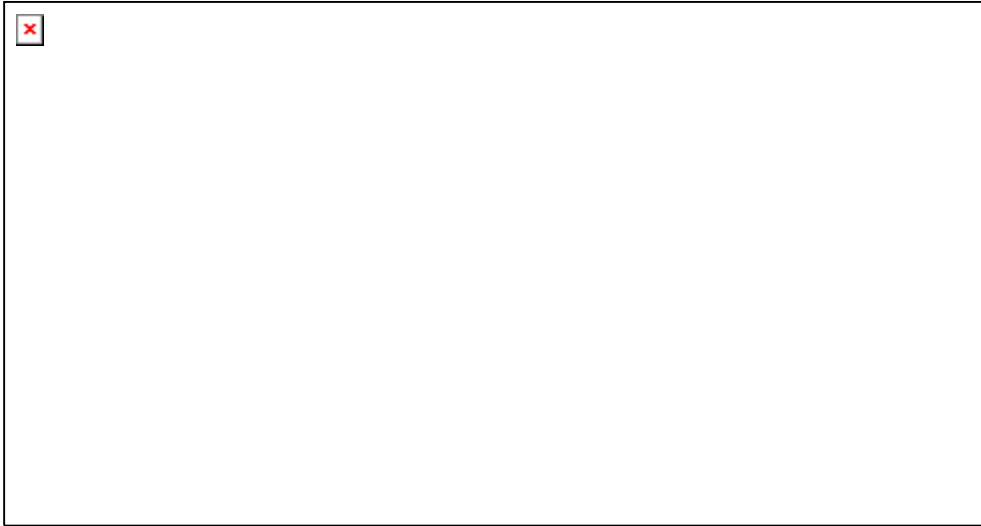
- | | |
|----------------------|---|
| (a) None | no shadows are displayed |
| (b) Left, clear | characters in the shadow region (below and to the left of windows) are 'dimmed' |
| (c) Left, [texture] | characters in the shadow region are replaced by the Shadow texture |
| (d) Right, clear | as for (b), except the shadow region is below and to the right of windows |
| (e) Right, [texture] | as for (c), except the shadow region is below and to the right of windows |

Pressing ^ discards any changes to the Shadow setting.

EGA/VGA Palette

Pressing P or clicking on **VGA Palette...** brings up the Palette dialog. The palette defines the 16 colours that are available while Arc is running (the default palette is restored on exit), as well as the screen border colour. The VGA palette feature is only available on VGA or MCGA systems, and the EGA palette feature is only available on EGA systems.

Figure 9
Palette



The following keys are used to manipulate the palette:



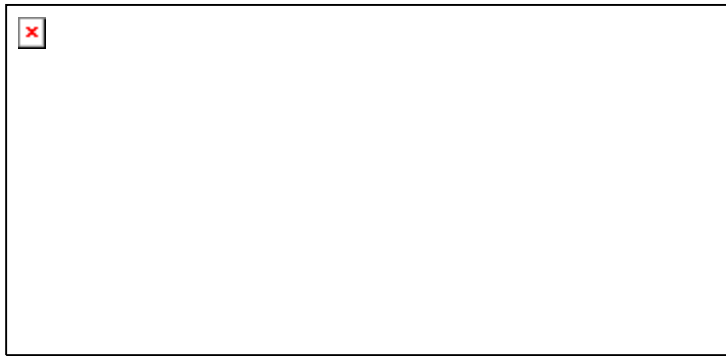
Key	Action
f	Returns to the Scheme dialog without saving any changes. Changes should be saved by pressing m on return to the main Scheme window, or when prompted to before exiting
m	Saves changes and returns to the Scheme dialog
^	Discards changes and returns to the Scheme dialog
z	Decreases the palette value (down to 0)
x	Increases the palette value (limit 63 on a VGA, 3 on an EGA)
D	Sets the current palette entry to its default DOS value
R	Restores the current palette entry to its previous value

The **Allow Blinking** field gives the user the choice between an extra 8 background colours or blinking background colours. Usually **Allow Blinking** is disabled. Pressing k toggles the setting, pressing Y enables blinking and pressing N disables blinking.

Load Colour Scheme

Load colour scheme allows a complete definition of Arc's colours and textures to be loaded. From the Scheme dialog, press L, or click on **Load Colour Scheme**. If the current colour scheme has not been saved, you will be asked if you wish to do so before continuing.

Figure 10
Load Scheme



Use **w**, **y**, **g** and **d** to select the scheme to load and press **f**. As the current selection changes, Arc's presentation will change to show you how the scheme you are about to load looks.

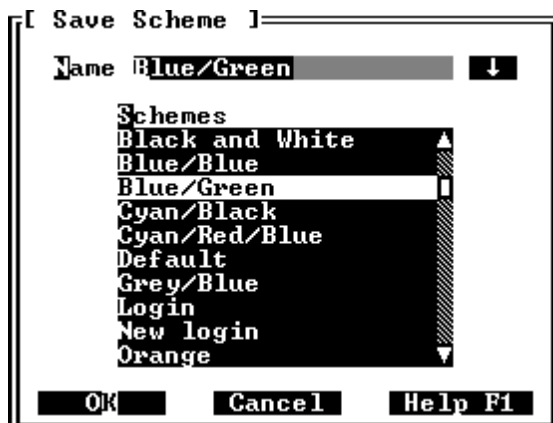
Press **^** to exit without loading a colour scheme, and revert to the previous colour scheme.

- 2 Only supervisors (users with an authority level of 99) are allowed to delete schemes saved in `SCHEMES.DAT`, all other users can only load scheme definitions.

Save Colour Scheme

Save colour scheme allows you to save a complete definition of Arc's colours and textures. From the main Scheme window, press **S**, or click on **Save Colour Scheme**.

Figure 11
Save Scheme



Enter a name for the colour scheme you are saving. Pressing **f** then saves the scheme with the given name. To access a history of previously entered Scheme Names, press **y**. Press **^** to exit without saving a colour scheme.

- 2
 - Only supervisors (users with an authority level of 99) are allowed to save scheme definitions to `SCHEMES.DAT`, all other users can only load scheme definitions.
 - The schemes that can be loaded and saved (separate to the user's scheme) are stored in the file `SCHEMES.DAT`. Each user's individual scheme is saved in `USER_INF.DAT` along with their preferences and other information.

Inserts a new item

The Insert command window is accessed by pressing **h** or by pressing **ul** or by clicking **F10 Commands** on the status line and then selecting **Insert** in the Command menu.

Insert places a new item in the current menu window, and allows you to move it to any position within any menu window (most movement keys operate as normal) :



Key	Action
[Hot Key]	Moves the new item to the position of the menu item that has the hot key you pressed.
h	Confirms the new item's position
^	Cancels the Insert command

When the new item is where you want it, press **h** (or click on **Insert Place new item** on the status line). A new window will appear, asking you which type of item you would like to insert:

Figure 12
New Item
Type



The new item can be one of:

- Divider** A horizontal line (with an optional title) that can be used to divide your menu windows up into panes of related items
- Option** Capable of executing any command
- Menu** A window into which you can place other items
- Link** A link to another structure. This has the effect of placing an entire structure from another directory into the current menu, making it look like another menu

Use **w**, **y**, **g** and **d** to select the new item's type, or use a mouse to click on the appropriate item in the *New Item Type* dialog. Once you have selected the item's type, you will be placed in the Edit command (described on the next page).

Edits any menu item

To quickly edit the item under the cursor bar, press **t**. Otherwise, press **uE**, or click **F10 Commands** on the status line and then select **Edit** in the Command menu.

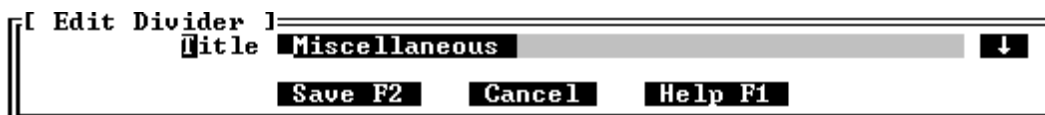
The number and type of fields is different for each type of menu item, and each is described below.

After you have finished editing, to save your modifications, press **m**. To discard any changes you have made, press **^**.

Editing Divider Bars

Normally you cannot highlight a divider bar with the cursor. Arc allows you to highlight divider bars as well as all ordinary menu items *only* when **Edit** is chosen from the Command menu. Select the item you wish to edit using the cursor bar movement keys, then press **t**. To cancel the Edit command, press **^**.

Figure 13 Edit divider bar



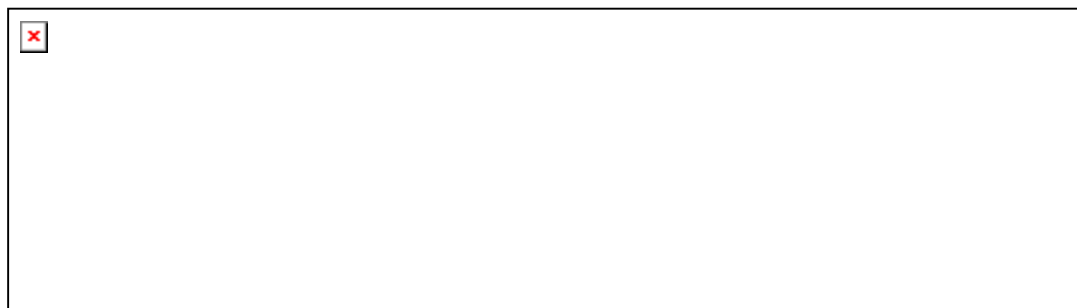
Title

The title of a divider bar cannot contain hot keys (see Edit option below). The text provided is displayed exactly as given. It is common to insert a space before and after the text to space it out from the horizontal line drawn.

Editing Options

Options include a title field as for a divider bar (see above). Options access the command editor with **t**, and AutoBuild with **u**.

Figure 14 Edit option



Title

The title of a menu item appears in the menu window, and is used to provide the user with a description of that item. An option, menu or link title may optionally specify a hot key by placing a carat **^** (j6) in front of the character to be the hot key e.g.

The letter 'F' of the title would appear in a different colour, and that item can be accessed simply by pressing F when the cursor bar is in the same menu window the item is in, in addition to moving the cursor bar over the item and pressing f. To access a history of previously entered Titles, press y.

- 2 Arc will not allow you to give an item the same hot key as a hidden item, and it will warn you if you try to give an item a hot key that has already been used

Short Cut A short cut allows you to access items even when they are not shown on the screen.

Available/
Used Short
Cuts Pressing y while editing this field displays a list of available/used short cuts. The item that uses the short cut is shown on the right hand side. Duplicate short cuts are easily identified by looking for blank short cut entries in the left hand column; this means that more than one item shares the same short cut. The first item listed against a particular short cut is the one that will be accessed when the short cut is pressed. Items that you do not have the authority to access are shown as **Hidden item**. The Exit short cut is also shown; it always has precedence over other items. You can choose a new short cut from this list by highlighting it and pressing f.

Figure 15
Short Cut List



Arc allows the following short cuts to be used:

F3..F8
Ctrl+A..Ctrl+Z (except Ctrl+M), Ctrl+F2..Ctrl+F10, Ctrl+Home
Alt+A..Alt+Z, Alt+0..Alt+9, Alt+-, Alt+=, Alt+F1..Alt+F10
Shift+F1..Shift+F10

If you press a short cut that is already in use, Arc will tell you which item has been assigned that short cut. It is up to you to decide if you want duplicate short cuts. If multiple items use the same short cut, Arc will act on the first one it finds. If you press an invalid key combination you will also be warned.

- 2 You cannot give an item the same short cut key as the Exit short cut, or the same short cut key as an item which is displayed as **Hidden item**.

Password The item password entered here will have to be entered by any user before they can access (execute, enter (if a menu), edit, delete, move or copy) the item. This password is entirely separate from the user password.

Authority The authority level determines who has access to the menu item. A user requires an authority level greater than or equal to an item in order to access it. If you do not have the authority to access an item, it will not be displayed. The highest authority level is 99 (a supervisor), and the lowest is 0. The default value is 0. The value may be changed by entering it directly or the w and y keys may be used to increase or decrease the current value.

- 2 You cannot give an item an authority level greater than your own.

Pause after execution When enabled, Arc pauses after running a program so that you can examine the program's output, and then press [Enter] to return to Arc.

Change the setting by pressing k to toggle, Y to enable or N to disable.

Reset recovery When enabled, Arc to take steps that will allow it to re-run this program if it does not finish because of the computer losing power, being turned off or rebooted. This is very useful for BBS, fax machines or other systems that must remain running. The user is given 20 seconds to abort the program before it is re-run. Note that programs that have *Don't return to Arc* flagged will actually return to Arc if they are forced to recover from a reset. This is because the normal action the user would take would be to return to Arc.

Change the setting by pressing k to toggle, Y to enable or N to disable.

Prompt for log message When enabled, Arc to ask the user for extra information to be placed in the log file. This is very useful for identifying which project was being worked on for billing customers etc. This prompt only occurs when logging is enabled.

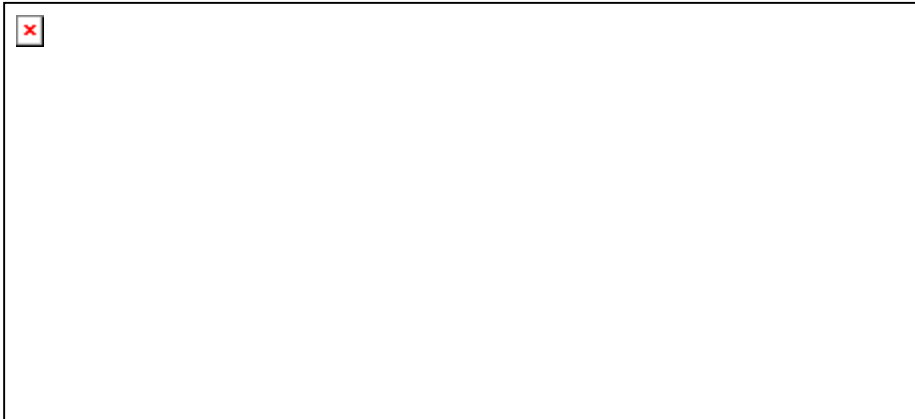
Change the setting by pressing k to toggle, Y to enable or N to disable.

Don't return to Arc When enabled, Arc will run the specified program and then exit, so Arc is not re-displayed when the program finishes. This is useful for "logout" style options where particular actions must be performed before Arc exits. If Arc is re-invoked after an option of this kind is run, it will re-display with the cursor in the same position as the user left.

Change the setting by pressing k to toggle, Y to enable or N to disable.

AutoBuild The AutoBuild function allows you to add commands to a new option by only knowing the drive, directory and application name of the program you are adding. A warning is given if the option already has commands associated with it that would be replaced.

Figure 16
AutoBuild



After **u** is pressed (make sure you are not editing the Short Cut field when you press **u**), the current directory is shown highlighted, and any executable files (files with **.COM**, **.EXE** or **.BAT** extensions) are shown below any sub-directories it might have. To examine a different drive, move to the top of the list (press **g**) and then press **f**. Arc will then show you a list of the available drives. Highlight one and press **f**.

If you know the name of the program, you can enter it directly by moving to the Name field (press **av** or **aN**) and then enter the name of the program. To access a history of previously entered Names, press **y**.

Once you have found the drive and directory containing the application you want to run, examine the list of executable files shown below the directory names. Highlight the application and press **f**. Arc will then save the option and its associated commands.

To cancel AutoBuild press **^** at any time.

Calling .BATch files

Any commands that follow the execution of a batch file from within the temporary batch file created by Arc (**\$TEMP .BAT**) will not be executed unless DOS's **CALL** statement is used. This allows control to return to **\$TEMP .BAT** when the secondary batch file is complete.

e.g.

```
CALL XTGOLDB .BAT  
PAUSE
```

Unless you use the **CALL** statement, control will not be returned to **\$TEMP .BAT** (it returns to **ARC .BAT** instead), and the **PAUSE** command will never be executed. However, Arc will reappear.

Editing an Option's Batch Commands

When editing an option, pressing **t** causes Arc to invoke a text editor that allows you to modify the batch file associated with the option. The commands contained in the batch file cause the option to be executed when the user selects it during normal usage. Once these instructions have been edited and saved by the text editor, and you have exited from the text editor, Arc will be re-displayed, and the changes will be saved. If no

alterations were made to the batch file, Arc will return to the Edit window, rather than automatically saving changes.

Editing Menus

Menus include the same fields as options (see above).

Figure 17
Edit menu



Maximum Height

Menus have an additional field not present in options - the maximum height. A maximum height of zero means that the menu will automatically resize itself to hold as many items as it can, up to the height of the screen. Any other value means that the menu can display no more items than specified. If there are less items in a menu than specified then the menu will be only big enough to display them. The default value is 0, and may be changed by entering it directly or the w and y keys may be used to increase or decrease the current value.

Editing Links

Links include the same fields as menus (see above).

You can choose to have structures linked always (an **automatic** linked structure), or only when access on them is attempted (a **manual** linked structure). The short cuts of manually linked structures are only available after that structure has been accessed, whereas the short cuts of automatically linked structures are always available. A manual linked structure does not use any memory until it is accessed.

The last three fields shown in Figure 18 below are specific to links.

Figure 18
Edit link



Link Path

This field specifies the directory in which the structure to be linked resides. For example, if you wished to link a structure in D:\LOTUS\DATA then you would place

D:\LOTUS\DATA

in the link path field. To access a history of previously entered Link Paths, press *y*. Link paths can also include embedded environment variables. This is very useful for implementing user-specific menus on a network. By specifying the link path as

```
f:\user\%USER%\
```

each user can have their own private menu structure linked into the main network menu. The environment variable *USER* changes depending on which user is logged in to the network.

Link Number This field specifies the number of the structure in the link path that should be linked into the current structure. The link number can be any integer between 0 and 99, corresponding to structures *M_STRUCTURE.D00* through *M_STRUCTURE.D99*. If this field is left blank, it is interpreted as a 0. The link number can be entered directly or the *w* and *y* keys may be used to increase or decrease the current value.

Automatic Link When ticked, this field causes Arc to always read in the structure specified. This is an **automatic** link. When unticked, the linked structure is only loaded on demand i.e. when the user tries to access the link. This is a **manual** link. This setting conserves memory. The short cut keys of a linked structure without automatic link enabled are not available until that linked structure is loaded.
Pressing *k* toggles the value, and pressing *Y* or *N* ticks and unticks the field respectively.

AutoBuild with Links Links require a drive, path and structure number to tell Arc which structure is to be loaded. Links can use AutoBuild in much the same way as options, simply by pressing *u*.

Copy

Duplicates the item under the cursor bar

Invoke Copy (after positioning the cursor bar over the item to copy) by pressing **uC**, or by clicking **F10 Commands** on the status line and then selecting **Copy** in the Command menu.

- 2** You must know the password of an item (or be a supervisor) before you can copy it. You cannot copy a menu that contains hidden items (items that you do not have the authority to access).

A new item identical to the original will be created, whether it is a menu, a link, or just an option.

Copy can be used to make the addition of similar options easier, or it may be used to copy entire menus before moving them to another structure.

- M** Unless the short cut(s) of the item copied are re-defined, Arc will only 'see' the short cuts of the original item, as it encounters them first.

Move

Moves any item to any position

Position the cursor bar over the item to move, then access the Move command by pressing uM, or by clicking **F10 Commands** on the status line and then selecting **Move** in the Command menu.

Use the cursor bar movement keys to move the item to the new position:



Key	Action
m	Confirms the item's final position
^	Cancels the Move command, returning the item to its original position
z	Moves the item to the previous menu
x, f	If the item is above a menu, this moves the item into that menu
[Hot Key]	If you press an option's Hot key, the item is moved to the current position of that option. If you press a menu's Hot key, the item is moved into that menu

A mouse can also be used to position the item.

- 2
 - Since Arc requires at least one item per menu, if you try to move the sole remaining item out of a menu, it will create a blank option within that menu.
 - You must know the password of an item (or be a supervisor) before you can copy it. You cannot move a menu that contains hidden items (items that you do not have the authority to access).

Delete

Removes any menu item

Access the Delete command by pressing **C** or by pressing **uD**, or by clicking **F10 Commands** on the status line and then selecting **Delete** in the Command menu.

Delete allows you to move the cursor bar around before selecting the item to delete - this allows divider bars to be deleted. Using the cursor bar movement keys, position the cursor bar over the item you wish to delete, and then press **C**.



Key	Action
[Hot Key]	Moves the cursor bar to the item with the Hot Key that you pressed
C	Deletes the item under the cursor
^	Cancel the Delete command

M When you choose to delete a divider bar, Arc will not ask for confirmation - it will remove it immediately.

If you selected an option, menu or link to delete, you will be asked to confirm deletion by pressing **Y** (Or click on the **Yes** button). Deleting a menu also deletes any options or menus contained within it, so use caution with this command.

- 2**
 - Since Arc requires at least one item per menu, if you try to delete the sole remaining item from a menu, it will create a blank option within that menu. If you want to remove a menu, just delete its menu item.
 - You must know the password of an item (or be a supervisor) before you can delete it. You cannot delete a menu that contains hidden items (items that you do not have the authority to access).

Move Window

Repositions a menu window anywhere on the screen

First make the menu window to be moved 'active', and then invoke the Move Window command by pressing `uW` or by clicking **F10 Commands** on the status line and then selecting **Move Window** in the Command menu. Then use the following keys to reposition it:



Key	Action
m	Confirms the window in its current position, and saves it. It then returns to the Command menu
^	Discards any changes (returning the window to its previous position), and returns to the Command menu
w	Moves the window up one screen row
y	Moves the window down one screen row
z	Moves the window left one screen column
x	Moves the window right one screen column
g	Moves the window up and to the left
d	Moves the window down and to the left
{	Moves the window up and to the right
}	Moves the window down and to the right
+	Enlarges the window by one line
-	Reduces the window by one line

If the menu window is moved off one edge of the screen, it will re-appear at the opposite edge.

- 8 If you are using a mouse, you can move the window by dragging the window's title (place the mouse cursor on the window's top line, click and hold the mouse button and then move the mouse).

First Menu Name

Edits the properties of the first menu on the desktop

First Menu Name is accessed by pressing **uF**, or by clicking **F10 Commands** on the status line and then selecting **First Menu Name** in the Command menu.

Edit the first menu as you would any other menu. The short cut of the first menu is normally set to **bg**, pressing this key combination then returns you to the first menu.

The first menu's authority level cannot be changed, hence it can always be edited by any user with editing privileges. Any password that you set on the First Menu must be entered before it can be edited. All users are able to access the contents of the first menu regardless of its password.

To finish editing, press **m** to save any changes, or press **^** to abandon any changes.

Figure 19
First Menu
Name



User Accounts

Changes the details and access privileges of Arc's users

- 2
- You must be a supervisor (authority level of 99) to access User Accounts
 - Arc requires at least one supervisor account

User Accounts is accessed by pressing uU, or by clicking **F10 Commands** on the status line and then selecting **User Accounts** in the Command menu. You must re-enter your password before you can access User Accounts; this prevents unauthorised users from accessing it if you leave your computer unattended. Once the password has been entered correctly, User Accounts displays a list of the user IDs, names and authority levels of all users.

Figure 20
User
Accounts



The following keys are used in conjunction with the User Accounts window:



Key	Action
f	Edits the selected user's information
m	Returns to the Command menu
h	Inserts a new user account (at the end of the list). The new user's scheme, preferences and authority level are a copy of the currently highlighted user's settings
c	Deletes the selected user if it is not the current user i.e. your own! Confirmation is required
w	Moves up one user
y	Moves down one user
{	Moves up one page
}	Moves down one page
g	Moves to the first user
d	Moves to the last user

Edit User

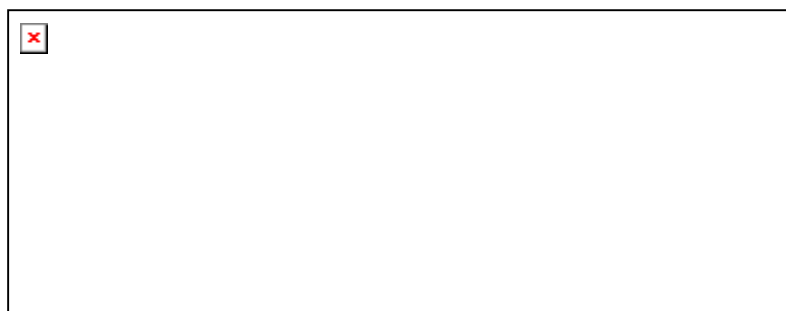
Once **f** has been pressed to edit a user's information, the Edit User window will appear over the User Accounts window. Within this window, the following keys may be used (in addition to the normal editing keys):



Key	Action
m	Returns to the User Accounts window, saving any changes
^	Returns to the User Accounts window, discarding any changes

Each user account consists of a number of fields described in detail below:

Figure 21
Edit User



User ID This field determines the unique ID this user will use to identify themselves when logging on. It is usually the user's initials or their first or last name. To access a history of previously entered User IDs, press **y**.

2 User IDs must be unique (different for each user).

User Name This field fully identifies the user, giving their full name. It can also be used to store other information, such as which department they belong to etc. To access a history of previously entered User Names, press **y**.

Authority This field specifies the user's authority level. A user can access items with an authority level less than or equal to their authority level. The maximum authority level, 99, is used to denote super-user or supervisor status. The lowest authority level is zero. A new authority may be entered directly or the **w** and **y** keys may be used to increase or decrease the current value.

Password This field specifies the user's password. Only those with the correct password can gain access to Arc. When the cursor is not on this field, it appears as a string of ampersands @ so a casual glance will not reveal the password.

Allow Exit to DOS When this field is unticked, the user will not be allowed to access DOS when they exit Arc. Instead, they will be returned to the log-in screen where a user with higher access privileges can log-off and exit to DOS, or someone else can log-in and start working.

- Allow Editing** When this field is unticked, the Command Menu items *Insert, Edit, Copy, Delete, Move item, Move window* and *First menu name* are removed, and the user is unable to alter the menu.
- Allow Password Change** When this field is unticked, the Command Menu item *Change Password* is removed, and the user is unable to alter their password.
- Allow Scheme and Preferences** When this field is unticked, the Command Menu items *Scheme* and *Preferences* are removed, preventing the user from changing colour schemes or user preferences.

LOGIN_USER

A special user **LOGIN_USER** is maintained by Arc to determine how the Login screen is displayed. The colours, textures, palette and preferences of this user are used to produce the login display. If this user name is not found before the Login screen is displayed, the scheme and preference settings of the last user in the user file are used.

To modify the settings, login as the LOGIN_USER, modify the settings and then save them.

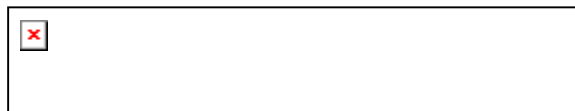
If you do not wish anyone to alter the LOGIN_USER's settings (by logging in as the LOGIN_USER), give the LOGIN_USER a password that only you know. Arc ensures that the LOGIN_USER always has an authority of zero, and cannot edit or exit to DOS. This is to prevent any security holes as a result of it being present.

Leaves Arc

Exit is accessed by pressing uX, or by clicking **F10 Commands** on the status line and then selecting **Exit** in the Command menu.

If an exit short cut is defined (it is displayed beside Exit in the Command menu) it may also be pressed to leave Arc, bypassing the normal confirmation window.

Figure 22
Save Settings



If you have changed any settings (scheme or preferences), you will be prompted to save these changes before exiting. If you exit without saving any changes they will be lost.

Figure 23
Confirm Exit



The Exit confirmation window gives you three choices:

Y Exit	leave Arc
L Log-in	return to log-in screen
^ Cancel	don't exit

- 2 If **Exiting** to DOS has been disabled (via the User Accounts command), the Log-in screen appears automatically.

Batch File Commands

Arc defines two components of a batch file:

- command section* Contains statements that are interpreted by Arc to prompt the user. A command section is included by ensuring that the **very first character** of the batch file is an open square bracket [. One or more statements are then followed by a close square bracket].
- batch section* Contains standard DOS commands.

Definitions

A *variable* holds a value provided by the user or pre-defined by Arc. Its name must start with a letter, thereafter it can contain both letters, digits, underscores _ and ampersands @ up to a maximum length of thirty characters.

Strings start and end with " characters, can be up to 255 characters long and cannot extend across lines. To embed a " character, use "".

A *field description* tells the user the purpose and use of the field being edited. You can specify a short cut key by preceding a letter character of the prompt with a tilde ~. Pressing a<letter> will then move the focus to the field. If not specified, it defaults to a generic message asking for input.

Pre-defined Variables and Environment Variables

You can use the value of most of the settings in the ARC . INI file (and a few others) anywhere within your batch file.

Variable name	Value
Arc_User	The user ID of Arc's current user.
Current_dir	Current DOS directory. For consistency, always ends with a \.
TemporaryBatchFile	Temporary batch file.
PrimaryStructure	Primary structure file.
TextEditor	Text editor.
TextEditorFlags	Command-line parameters given to the editor.
RestartInfoEnvStr	Environment variable that holds restart information.
PrevShellEnvStr	Environment variable holding prior restart information.
OnNetwork	TRUE if on a network, FALSE if not.
UserToLogin	User ID to login.
VideoMode	Current video mode, VGA, MCGA, EGA or CGA.
ScreenSaver	Command to run as a screensaver.

Variable name	Value
ResetRecovery	Name of the reset recovery file.
Logfile	Name of the log file.
ItemTitle	Title of the option running (with all hotkeys removed).

Editing DOS environment variables

You can use the value of DOS environment variables anywhere within your batch file. For example, the following batch file lets you edit the value of your PATH:

```
[EditField(P="%PATH%", "~Edit DOS path")]
set PATH=%P%
```

Using a Variable's Value

Once a variable has been defined, its value can be used anywhere in the batch file (even later in the command section). To use the variable's value, place the name of the variable in percent signs e.g.

```
%variable_name%
```

- You cannot use Arc variables in other batch files unless you pass them as parameters or copy them to a normal DOS environment variable. Arc variables are not stored in the environment - they are replaced by the actual value as the temporary batch file is being created.
 - If you wish to place a percent sign in a batch file without what follows being interpreted as a variable name, use %%.

Sub-strings

Arc allows sub-strings to be extracted in a number of ways:

```
%variable_name%
%variable_name[start]
%variable_name[start..]
%variable_name[start..end]
%variable_name[start..end,start..end]
```

where `start` is the starting index in the string and `end` is the end index in the string. An index of 0 is the end of the string, a negative value is measured back from the end of the string.

For example, if you had a variable called `copy_path` with a value of `D:\GAMES\WC`, the following constructs could be used in the batch file:

```
%copy_path%           D:\GAMES\WC
%copy_path[1]         D
%copy_path[1..2]      D:
%copy_path[9..]       \WC
%copy_path[-2..]      \WC
%copy_path[1..2,9..]  D:\WC
```

The trailing percent sign traditionally required for environment variable names is not usually required in Arc, which accepts any character that cannot be part of the variable

name as the terminator. The trailing percent can be used to maintain compatibility or when an alphanumeric character immediately follows a variable name (but not after a sub-string e.g. %copy_path[1]% is invalid unless it is immediately followed by another variable).

For compatibility with 4DOS and other COMMAND.COM extensions, Arc allows (and ignores) the environment variable-like constructs %0 through %127, %% , %_ and %@ (and what follows them) up to a following percent sign % or white space, %#, %& and %0& through %127&.

Verify Statement

The Verify statement is used to force the user to acknowledge that the program they are about to run could have harmful side effects or take a long time to complete. This is useful for options like FORMAT, FDISK, BACKUP, DEFRAG etc. An optional string (enclosed in double quotes) can be used to indicate the nature of the option being executed. The user must press f, Y or k to proceed. N or | cancels running the option. If user help has been defined for the option, it can be displayed by pressing bl or by clicking on **User Help**.

Usage:

```
verify( ["information string"] )
```

Example:

```
[verify("Use with caution! Easily destroys hard disk!")]  
fdisk
```

Figure 24
Verify



MustExist Statement

The MustExist statement is used to check that a file exists before trying to run an option. This can be used to avoid annoying *Bad command or filename* messages if a program has been deleted, moved, or is located on a file server that has not been logged in to or is currently down. If Arc can't find the file, it informs the user and asks them if they want to continue anyway. If user help has been defined for the option, it can be displayed by pressing bl or by clicking on **User Help**.

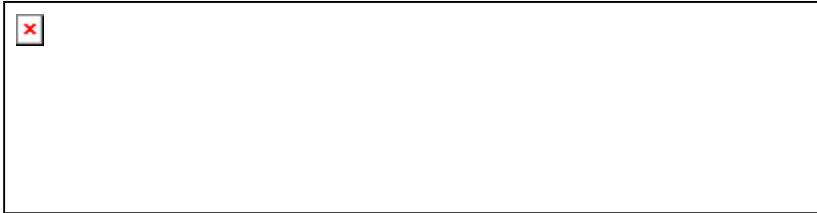
Usage:

```
MustExist("filename")
```

Example:

```
[MustExist("c:\arc\xmartin.exe")]  
c:\arc\xmartin.exe
```

Figure 25
MustExist



EditField Statement

The EditField statement is used to prompt the user with a text field. The maximum length of the field can be defined, as can the initial value, a prompt to tell the user what the field is for and a string identifying the history list associated with the field.

Use the editing keys to edit the field, then press f. If user help has been defined for the option, it can be displayed by pressing bl or by clicking on **User Help**.

Usage:

```
    EditField(  
        <variable name (must start with a letter)>  
        [ [maximum length] ]  
        [ ="initial value" ]  
        [ ,"field description"  
          [ ,"history list ID" ]  
        ]  
    )
```

variable name is the name being given to the variable. This field is not optional.

maximum length is the (optional) maximum length you would like the string to be. No strings longer than this value can be entered, and longer strings taken from the history list will be truncated. The value is enclosed in square brackets, and must lie in the range 1..255. If omitted, it defaults to fifty one characters.

initial value is the (optional) value the variable will take before the user edits it. If omitted, the string is initially blank.

field description If not specified, it defaults to a generic message asking for input.

history list ID is an (optional) identifier that associates a particular list of history strings with the field. The history list records previous values for the field, and allows them to be displayed, selected and deleted. If there are any history entries, an arrow will appear on the right hand side of the field. History lists may be shared by many options simply by providing the same history list ID (case is not significant). If not specified, it defaults to a generic history list.

Example:

```
[EditField(view[42]="%current_dir%",  
    "~File to view:",  
    "Viewed files"  
    )  
]  
more < %view%
```

Figure 26
EditField



Command
History

While an EditField is displayed, the user may press **y** to access a history of up to fifty recently entered strings (if any exist). An old entry may be selected with **f** and then edited, or removed by pressing **c**. The history list window is removed after either **f** is pressed to select an entry, or **^** is pressed.

Arc's history list IDs are:

History string ID	Holds history for...
None	The field's value will not be saved anywhere
Password	The field's value will not be saved anywhere, and will be displayed as @'s when being entered by the user.
Option parameters	Default value for any EditField that does not override the history list ID
Item title	Edit\Title
AutoBuild option	AutoBuild option\Name
AutoBuild link	AutoBuild link\Name
Link path	Edit link\Link path
Title box	Preferences\Title box
Save scheme	Scheme\Save scheme
User ID	User accounts\User ID
User name	User accounts\User name
Log message	The log message provided by the user when prompted by an option with <i>Prompt for Log Comment</i> enabled

CommandButtons Statement

The CommandButtons statement is used to provide a group of buttons for the user to choose from. Only one button can be selected, and doing so closes the window and proceeds to the next (if any). The leftmost button is always selected initially, so it should contain the default response. A string is used to tell the user what the buttons do, and each button can have an associated value string which is the value the variable takes when that button is chosen.

To change the selected button, press **v** to move right or **jv** to move left. When the button you want is highlit, press **f** to select it. You can also press the highlighted letter of the button. If user help has been defined for the option, it can be displayed by pressing **bl** or by clicking on **User Help**. Pressing **|** cancels execution of the option.

Usage:

```
CommandButtons(
  <variable name (must start with a letter)>
  ," <string> "
  ," <button description> "
  [ =" <button value when selected> " ]
```

```

    [ ,<additional button descriptions/values> ]
)

```

The button description is the text that is displayed on the button on the screen. Each button description can specify a short cut key by preceding a letter character with a tilde ~. Pressing a<letter> will then take you directly to that command button, and pressing f or k will then choose it.

The button value is the value assigned to the variable name when that particular button is chosen. Button values can be up to 255 characters in length. If no button value is given, it gets set to the number of the button, with the first button being "0".

Example:

```

[CommandButtons(source,"Drive to copy from?",
  " A:  "="A:", " B:  "="B:")
 CommandButtons(dest,"Drive to copy to?",
  " A:  "="A:", " B:  "="B:")]
diskcopy %source% %dest%

```

Figure 27
Command-
Buttons



RadioButtons Statement

The RadioButtons statement is used to provide a group of items for the user to choose from. Only one item can be selected, and doing so closes the window and proceeds to the next (if any). The uppermost button is always selected initially, so it should contain the default response. A prompt is used to tell the user what the buttons do, and each button can have an associated value string which is the value the variable takes when that button is chosen.

The dialog box shows a list of alternatives, of which only one may be chosen. Use the w, y, g, d keys to make a selection, and then press f. If user help has been defined for the option, it can be displayed by pressing bl or by clicking on **User Help**. Pressing | cancels running the option.

Usage:

```

RadioButtons(
  <variable name (must start with a letter)>
  [ = <initial value, 0 to #options-1> ]
  ," <field description> "
  ," <item description> "
  [ =" <item value when selected> " ]
  [ , <additional item descriptions/values> ]
)

```

The initial value is the number of the radio button to initially select. By default this value is 0 (ie the top item). This provides the default value if the user chooses not to edit it.

The item description is the text that is displayed on the screen.

The item value is the value assigned to the variable name when that particular radio button is chosen. Radio button values can be up to 255 characters in length.

Example:

```
[RadioButtons(which=2,"~Drives to scan for viruses?",
  "All local drives"="/ADL",
  "All network drives"="/ADN",
  "All local and network drives"="/ADL /ADN"
)
]
scan %which%
```

Figure 28
RadioButtons



CheckBoxes Statement

The CheckBoxes statement is used to provide a group of on/off items for the user to toggle. Each item can be toggled individually. Each item requires a separate variable name, description and on/off value.

Use **v** and **jv**, **w** and **y** to move between check boxes (or press **a**<the check boxes' highlighted key>), and press **k** to toggle their value, **Y** to tick it or **N** to clear it. Once they are set correctly, press **f**. If user help has been defined for the option, it can be displayed by pressing **bl** or by clicking on **User Help**. Pressing **|** cancels running the option.

Usage:

```
CheckBoxes (
  <variable name (must start with a letter)>
  [ = <initial value, 0 or 1> ]
  , " <check box description> "
  [ , " <on value> "
    [ , " <off value> " ]
  ]
  [ , <additional check box variables> ]
)
```

The initial value sets the check box to on (1) or off (0). The default is to set it on.

The description tells the user the purpose and use of the check box. You can specify a short cut key by preceding a letter character with a tilde ~. Pressing **a**<letter> will then take you directly to that check box.

The on value is the value assigned to the check box variable when that check box is turned on. If no value is provided, the variable is set to "1".

The off value is the value assigned to the check box variable when that check box is turned off. If no value is provided, the variable is set to "".

Example:

```
[ CheckBoxes (
```

```

quick    ,"~Quick format","/F",
five    =0,"~Format 360K","/4",
system=0,"~System files","/B",
copy_s=0,"~Copy system files","/S",
one     =0,"~One side only","/1"
)
]
format a: %quick% %five% %system% %copy_s% %one%

```

Figure 29
CheckBoxes



- 2 Notice how the value of **drive** is being used in the prompt for **fixup** by enclosing it in percent signs.

If... then...
else...
statements

If ... then ... else... statements are not allowed within the special batch section. However, you can make one by embedding statements within a variable and then referring to that variable within the command section. The following example asks the user which drive to format, then gives different format density options depending on which drive is chosen:

```

[radiobuttons(askformat,"~Drive to format?",
" A:  ="radiobuttons(fmt,"A: ~Density?",
"360 Kbytes"=="a: /f:360",
"1.2 Mbytes"=="a: /f:1.2"),
" B:  ="radiobuttons(fmt,"B: ~Density?",
"720 Kbytes"=="b: /f:720",
"1.44 Mbytes"=="b: /f:1.44")")
%askformat%
]
format %fmt%

```


BEdit Batch File Editor

BEDIT.EXE is a small text editor. It is used for editing batch files (up to 64 Kbytes).

Usage: BEDIT [*filename*] [/ROW *Row*] [/COL *Column*]

If *filename* is omitted, BatchEdit works with the file **untitled.txt**. BatchEdit allows the initial cursor row and column to be given as command-line parameters - this feature is used by Arc when an error is found in a batch file. BatchEdit's commands are Word Star compatible. BatchEdit supports 43/50 line EGA/VGA screen modes and 132/100 column modes.

BatchEdit has the following commands:

	Key	Action
Block commands	bK B	Marks the start of a block
	bK K	Marks the end of a block
	bK L	Marks the current line as a block
	bK V	Moves the block to the current cursor position
	bK Y	Deletes the block
	bK C	Makes a copy of the block at the cursor position
	bK H	Hides the block
	bK R	Reads a block from a file
	bK W	Writes the block to a file
	bK I	Indents the block one space
	bK U	Unindents the block one space
	bK P	Print the block
	bK n	Set bookmark n (0..9)
File commands	bK S	Saves the file
	^, bK Q	Abandons any changes to the file after confirmation, and exits
	bK A	Names file and then saves it
	m,bK X	Saves file and exits
Deletion commands	c, bG	Deletes the character under the cursor
		Deletes the character to the left of the cursor
	, bH	
	bT	Deletes a word
	bY	Deletes the line the cursor is on
	bQ Y	Deletes from the cursor to the end of the line

Cursor movement commands	w, bE	Moves up one line
	y, bX	Moves down one line
	z, bS	Moves left one character
	x, bD	Moves right one character
	bx, bF	Moves one word right
	bz, bA	Moves one word left
	bW	Scrolls display up one line
	bZ	Scrolls display down one line
	{	Scrolls display up one page
	}	Scrolls display down one page
	g, bQ S	Moves to the start of a line
	d, bQ D	Moves to the end of a line
	b{, bQ R	Moves to the top of the file
	b}, bQ C	Moves to the end of the file
	bQB	Moves to the start of the block
	bQK	Moves to the end of the block
	bg, bQ E	Moves to the top of the screen
	bd, bQ X	Moves to the bottom of the screen
	bQ n	Goto bookmark n (0..9)
	bQ L	Goto line (relative or absolute)
bQ [Find matching [] {} () to that under cursor	
Other functions	bQ F	Find
	bQ A	Find and replace
	bL	Repeat last find/find and replace
	l	Displays general help
	h, bV	Toggles between Insert/Overwrite modes
	f	Starts a new line, cursor moves onto the new line
	bN	Starts a new line, cursor remains in same position
	bP	Inserts a control character. A window will appear from which you can select the control character. Move the cursor over the one you wish to use, then press f. Press ^ to leave this window
	bQ Z	Toggles display of End of Line (EOL) markers
	bQ G	Strip unnecessary spaces from end of lines

Notes on .BATch files

All batch files should end with a Carriage Return/Line Feed combination (Decimal 13,10 Hex 0D,0A). This is normally inserted by pressing f after the end of the last line. Unless this is done, the last line of the batch file may never be executed by DOS (some versions). BEdit automatically places a CR/LF at the end of any **.BAT** file it saves.

Security / System Files

While Arc is actually displayed, the user cannot break out of it. However, while DOS is running the temporary batch file (and Arc is inactive), the user can stop the batch file running and access DOS by pressing `bC` or `b~`. Arc is not designed to prevent this.

Any solution you find-

- must resist attempts to break out of `AUTOEXEC.BAT` or any other `.BAT` file using `bC` or `b~`
- must prevent the computer from being booted from a floppy
- must not depend on a CMOS password to restrict access (since this is easily circumvented)

System Security

There are several simple steps you can perform to make your computer much more resistant to privacy invasion.

1. Most computers can boot (read the system) from either a floppy disk or the hard drive. By changing the order in which the computer checks for a bootable disk from `A:→C:` to `C:→A:`, you can prevent access being gained by someone inserting a bootable floppy disk. This setting is changed using the computer's CMOS Setup (usually accessed by pressing `C` just after the memory test).
2. Put a password on your CMOS Setup to prevent changes being made to the above setting. Keep in mind, though, that most CMOS Setup programs have a master password or other method of removing the password, such as disconnecting the battery, or shorting out specific pins.
3. If you are using DOS 6.0 or higher, put the command
`switches /n /f`
as the very first line of your `CONFIG.SYS`. This prevents the user from pressing `p` to skip all the commands in `CONFIG.SYS` and `AUTOEXEC.BAT`, or pressing `s` to confirm each line. The `/f` switch disables the normal 2 second delay after displaying "Starting MS-DOS" during startup-this is when these keys are checked for.

Prevent
bypassing
Autoexec.
Bat

You can prevent tapping of `bC` or `b~` from bypassing your `AUTOEXEC.BAT` by putting the following command at the end of your `CONFIG.SYS`:

```
shell=c:\command.com /p nul
```

Also, put

```
ctty con
```

as the last line to your AUTOEXEC .BAT. If you don't, the keyboard will not respond.

M Before you perform this procedure, make sure that you have a floppy to boot from in case something goes wrong.

Preventing batch files from being aborted

It is possible to prevent the user from interrupting a batch file by using the `ctty` command to reassign the input (and the output) device. Here is an example with a `DIR` command. Note the `<` and `>` redirections which are needed while the `ctty` has been assigned to `nul`.

```
rem Disable keyboard/monitor as input/output device
ctty nul
dir > con
rem Restore keyboard/monitor as input/output device
ctty con
```

Note that this trick does not prevent you from rebooting with `bac` while the batch file is running.

Arc's Security

Passwords

Arc's passwords (both those stored in the user file and those attached to menu items) are encrypted for maximum security. This ensures that casual viewing of these files will not reveal valuable passwords. On a network, the passwords that Arc uses are entirely independant of the network's passwords, and should in fact be made different. This is because the network goes to far greater lengths to conceal passwords, and having the same passwords used in Arc constitutes a slacker security. It is unnecessary to use passwords for each user because the automatic login process does not require them (see page 57).

When a password is required to access (view, edit, delete, move, copy) a menu item, the following dialog is displayed.

Figure 30
Password
Dialog



Supervisors (users with an authority of 99) have the option of pressing `t` to skip entering a password. This enables them to remove or access items that they have forgotten the password of. This facility is unavailable to all other users.

Hidden Items Any menu items that the user cannot access (due to a lack of authority) are not displayed. The user cannot copy, move or delete any menu that contains such 'hidden' items.

Security and Short Cuts A user can only utilise short cuts that are visible to them (not those on hidden items) i.e. they must be able to access the item in question via manual navigation through the menu structure. A user cannot give an item a short cut that is already being used by a hidden item (an item that they do not have the authority to access).

If you press an item's short cut, you will be asked for the password on all menus between your current position and that of the item in question, and then finally, the password of the item itself.

System Files

Arc's files and their locations are as follows (names that can be redefined are shown in *italics*):

File	Description
	Arc directory. Usually C:\ARC, or F:\PUBLIC\ARC
ARC.EXE	Arc application file
<i>ARC.BAT</i>	Arc's mandatory batch file. This file may be located anywhere in the user's PATH. Its first command changes to the Arc directory. The name and path of this file is user-defined, but it may reside in the Arc directory, with the PATH statement pointing to it.
USER_INF.DAT	All users' authority levels, passwords, preferences and colour schemes
SYS_HELP.TXT	Context-sensitive system help
SCHEMES.DAT	Predefined colour schemes
	Structure directory i.e. same directory as M_STRUCTURE.Dxx
M_STRUCTURE.Dxx	Menu structure file
M_STRUCTURE.Bxx	Backup menu structure file
M_OPTION.Bxx	Option batch file being edited
M_LOCK.Dxx	Lock file. Created while a user is editing the structure, and removed when the changes are saved or discarded. Prevents other users from editing the same structure at the same time
	User-defined directory (usually found in Arc's directory)
<i>\$TEMP.BAT</i>	Temporary batch file. Holds the batch file commands associated with the option being run. The name and path of this file is user-defined. It usually resides in the Arc directory, or in the user's home directory on a network
HISTORY.TXT	Histories of all edit fields, including for all options that utilise a VARIABLE statement for input. Stored in Arc's directory, or in the user's home directory on a network
<i>ARC.INI</i>	Arc configuration. The name and path of this file is user-defined, but defaults to ARC.INI in Arc's directory
<i>LOG.TXT</i>	Usage tracking file. The name and path of this file is user-defined, but it is usually created in Arc's directory

A separate temporary batch file (and environment string) must be maintained for each Arc invocation. This could be for each login from different terminals of a single user on a network, and for each session on a single machine.

Network Rights

All users should be at least able to read from all of Arc's files. The following table shows the access rights that users on a network require to use Arc's files. Each type of user inherits the access rights of the type of user shown above them. Arc gives the user the option of creating a file if it does not already exist or if it has been corrupted. This occurs with USER_INF.DAT (when Arc loads) and M_STRUCT.Dxx (when Arc attempts to load a structure).

File	Operations
Normal user - no editing rights, authority less than 99	
<i>ARC.BAT</i>	Execute
<i>ARC.EXE</i>	Execute
<i>ARC.INI</i>	Read
USER_INF.DAT	Read, write
\$TEMP.BAT	Create, execute, write, delete
SYS_HELP.TXT	Read
M_STRUCT.Dxx	Read
SCHEMES.DAT	Read
HISTORY.TXT	Create, read, write
<i>LOG.TXT</i>	Create, write
Editing user - editing rights, authority less than 99	
M_STRUCT.Dxx	Create, write, rename
M_STRUCT.Bxx	Delete
M_LOCK.Bxx	Create, read, write, delete
M_OPTION.Bxx	Create, read, write, delete
Supervisor - full editing rights, authority of 99	
SCHEMES.DAT	Write

Structure Consistency

When a user runs an application, Arc records which menu item the cursor was on so that they can be returned to that same position when Arc resumes. Any alteration to the relative position of items within the structure renders that recorded information invalid. Arc monitors the date and time of the structure file both before and after an application is run. If any change is detected, the cursor position information is reset.

Locking

When a user initiates an editing operation on a menu structure, Arc locks it to prevent other users from editing. The lock may be overridden by the supervisor at their discretion, however it is advisable to wait a reasonable time before doing so, between 5 and 20 minutes. The elapsed time since the lock was created is shown, as is the user Id of the user who created the lock.

There are four locking scenarios that can occur when more than one copy of Arc is trying to editing a structure at the same time.

1. User A obtains a lock because no-one else is currently editing. User B then tries to obtain a lock, but is refused because of User A's lock. User B is not a supervisor and so cannot overwrite the lock of User A.
2. User A obtains a lock because no-one else is currently editing. User B then tries to obtain a lock, but is initially refused because of User A's lock. User B is a supervisor and chooses to overwrite the lock of User A (for example, if it had been in place for over twenty minutes). User B then proceeds to edit the structure. Meanwhile, User A finishes editing and tries to save their changes, to find that their lock has been overwritten by User B. Arc then prevents User A from saving their changes, and marks their structure as invalid, because the copy they have in memory is different to the copy stored on disk. User A should reload their structure at the first opportunity.
3. User A1 obtains a lock because no-one else is currently editing. User A2 (with the same login name but from a different machine or DOS session) tries to obtain a lock, but is initially refused because of User A1's lock. Since A1 and A2 are the same user, and have supervisor rights, A2 chooses to overwrite the lock of User A1. User A2 then proceeds to edit the structure. Meanwhile, User A1 finishes editing and tries to save their changes, to find that their lock has been overwritten by User A2 (because the time and date of the lock has changed). Arc then prevents User A1 from saving their changes, and marks their structure as invalid, because the copy they have in memory is different to the copy stored on disk. User A1 should reload their structure at the first opportunity.
4. User A obtains a lock because no-one else is currently editing. User B then tries to obtain a lock, but is initially refused because of User A's lock. User B is a supervisor and chooses to overwrite the lock of User A (for example, if it had been in place for over twenty minutes). User B then proceeds to edit the structure. User B finishes editing and saves their changes. Then User A finishes editing and tries to save their changes, to find that their lock no longer exists. Arc then prevents User A from saving their changes, and marks their structure as invalid, because the copy they have in memory is different to the copy stored on disk. User A should reload their structure at the first opportunity.

The locking scenarios described above only occur when the second user to edit has supervisor authority and chooses to overwrite the first user's lock. They should only do this when that lock has been in place for a considerable time (twenty minutes). In most cases, only the supervisor will have editing rights, and so the cases described above will be the exception rather than the rule.

Read-Only Structures

Arc prevents a user from editing a structure that is flagged as read-only. This is useful if you would like each user on a network to have editing rights, but only on particular structures. For example, a common network menu structure could be flagged as read-only, and could contain a link to an editable structure in the user's home directory.

ARC.BAT

In order to operate correctly, Arc requires the presence of a specialised batch file. When the user selects an option, Arc writes the appropriate commands to the temporary batch file (usually \$TEMP .BAT). Then Arc exits with an exit code of 255, signifying that ARC .BAT should immediately execute the temporary batch file. If the user leaves Arc without choosing an option (i.e. exits from Arc), an exit code of 0 is returned, signifying the temporary batch file should not be executed. An exit code of 254 is for options that do not return to Arc - the temporary batch file is run but Arc is not-re-run. Any other exit code signifies an error of some description. A batch file operating on the above criteria is as follows:

ARC.BAT

```
@goto START
:_LABEL255
call c:\arc\temp.bat
del c:\arc\temp.bat
:BEGIN
c:
cd\arc
arc arc.ini /Z
for %%e in (0 254 255) do if errorlevel==%%e
    goto _LABEL%%e
echo Unknown exit code
goto end
:_LABEL254
call c:\arc\temp.bat
del c:\arc\temp.bat
goto end

:START
@ECHO OFF
rem ** Put initialisation code down here
    to speed up batch file **
goto BEGIN

:_LABEL0
:end
rem End actions go here
```

Network Installation

On a network, the temporary batch file used by Arc must be different for each user and for each of their logins. Several small changes are required for the Arc batch file and the ARC .INI file:

ARC .BAT Initialisation section:

```
ADD_USER %USER%
```

ARC .BAT Temporary batch file name used is:

```
%temp%\$temp%LB%.bat
```

And the corresponding settings in ARC .INI:

```
OnNetwork=True  
UserToLogin=%USER%  
TemporaryBatchFile=f:\usr\%USER%\$temp.bat
```

RAM drives

An alternative to the network solution shown above is to create a very small RAM drive for each login, and cause the temporary batch file to be written to it. This has the advantage that if a machine is turned off or rebooted, no files remain to be periodically deleted; it is also much faster than writing back to the file server or a local hard disk. It can also be used on a network where the file server is read only. Its disadvantage is that it consumes a small amount of memory, and that it may complicate debugging when there is a network problem.

Add the following line to your CONFIG.SYS:

```
device=ramdrive.sys 4
```

This creates a small (less than 4k) RAM drive.

Command Line Parameters & ARC.INI

Arc allows its default mode of operation to be changed through the use of a configuration file called ARC.INI. You can tell Arc to use another file by specifying it on the command line e.g.

```
arc net\lnch_net.ini /z
```

You do not need to do this if ARC.INI is in the same path as Arc (ARC.EXE) itself. Arc also requires the use of the command line parameter */z*, this reassures Arc that it is being run from a batch file, since if it is not, options will not be run. Users who run Arc directly from the command line are told which batch file to run.

The ARC.INI file contains a list of statements in the following format:

```
keyname=value
```

These statements define the value of each setting. **Keyname** is the name of a setting. It must be followed immediately by an equal sign =. The value can be an integer, a string or a Boolean (True or False), depending on the keyname.

- 2 You can include comments in the ARC.INI file. You must begin each line of a comment with a semicolon ;. You can also include environment variables by enclosing them in percent signs e.g.

```
TemporaryBatchFile=F:\USR\%USER%\$TEMP.BAT
```

LogFile=logfile name

Default: No usage tracking (commented out)

Usage tracking allows a supervisor to see what actions a user has been performing, and for which project. Each action is recorded in the log file specified using this option. Omitting a LogFile entry in ARC.INI (or leaving it blank) disables usage tracking, otherwise the filename given is used to track Arc events. If the filename given does not exist, it is created, if it does exist, new events are added (appended) to the file.

The events recorded are:

<i>Event</i>	<i>Description</i>
Logged in	User started Arc via log-in screen

<i>Event</i>	<i>Description</i>
Logged in automatically	User started Arc, bypassing log-in screen
Logged out	User left Arc
<Option name>	User ran option <Option name>
Returned to Arc	User returned to Arc after running an option
Login denied	Login was denied because of an incorrect password
Item access denied	Access to an item (e.g. edit, move, copy etc.) was denied because of an incorrect password
User Accounts denied	Access to User Accounts was denied because of an incorrect password

Projects

Projects can be tracked by creating the file PROJECT.TXT in Arc Menu's directory. Use one line per project, with a maximum of 10 characters per project name. When this file exists, Arc will prompt the user as they login, and the user must select a valid project before login can complete successfully.

Information is reported in the following format:

```

User Name      Time      Date      Project      Event
-----
1234567890    hh:mm:ss  dd-mm-yyyy Accounts      Event description

```

Sample usage file:

Arc Log File

```

User      Time      Date      Project      Event
-----
SUPERVISOR 22:11:11 12-09-1994 Accounts      LOGGED IN
SUPERVISOR 22:11:27 12-09-1994 Accounts      \Utilities\BatchEdit
SUPERVISOR 22:12:08 12-09-1994 Accounts      RETURNED TO ARC
SUPERVISOR 22:12:13 12-09-1994 Accounts      \Application Programs\MicroSoft
Windows
SUPERVISOR 22:12:54 12-09-1994 Accounts      RETURNED TO ARC
SUPERVISOR 22:12:55 12-09-1994 Accounts      LOGGED OUT
FRED      22:13:11 12-09-1994 Sales         LOGGED IN
FRED      22:13:33 12-09-1994 Sales         \Utilities\Directory Sizes
FRED      22:18:00 12-09-1994 Sales         RETURNED TO ARC
FRED      22:18:00 12-09-1994 Sales         LOGGED OUT

```

OnNetwork=<True | False>

Default: False

On a network, the temporary batch file used by Arc must be different for each user and for each of their logins. When OnNetwork is set to True, Arc interprets the TemporaryBatchFile setting differently.

As an example, assume that user SIMON has logged in, and that his home directory is F:\USR\SIMON.

Firstly, Arc appends the digits 000 to the temporary batch filename given e.g. if the ARC.INI setting was

```
TemporaryBatchFile=F:\USR\%USER%\$TEMP.BAT
```

the filename constructed would be F:\USR\SIMON\\$TEMP000.BAT. Arc then checks to see if this file exists. If it does exist, this implies that another invocation of Arc is currently running an application using that temporary batch filename. In this case, Arc increases the number by one (giving F:\USR\SIMON\\$TEMP001.BAT) and then checks if that file exists. It continues searching until it finds a temporary batch filename that does not exist.

Once found, it writes the batch commands to that file. It also sets the environment variable **LB** in the parent environment to the digits of the temporary batch filename that it has chosen (e.g. LB=001). This allows the Arc batch file to know which temporary batch file to CALL (and to DELeTe once it has been used).

PrevShellEnvStr=*environment string*

Default: None (is commented out)

This setting allows you inform Arc that a particular user has already logged on at another invocation of Arc (in a previous shell), and that they need not re-log.

For instance, your primary menu structure may use an environment string of **PRIMARY**, which is used to store information regarding the status of that user whilst within the primary menu structure. An option within the primary structure may take the user into a separate, secondary menu structure, which uses an environment string of **SECONDARY**. To avoid the hassle of re-entering your user ID and password when entering the secondary structure, and to prevent the secondary copy from affecting the first, the setting is used as follows:

```
PrevShellEnvStr=PRIMARY  
ReStartInfoEnvStr=SECONDARY
```

This says "Use the environment string SECONDARY to store information about the user's status in this menu level, but by-pass the log-in screen by assuming the same user as that stored in PRIMARY is still using the system."

PrimaryStructure=*[pathname]*number

Default: Structure 0 in the current directory (.\00)

This setting tells Arc the path and number of the primary structure (the structure that is loaded first). If Arc cannot find or successfully load this structure, it will offer to create a new one. If the user declines, Arc will exit.

e.g.

```
PrimaryStructure=c:\menudata\3
```

ReStartInfoEnvStr=*environment string*

Default: ARC

This setting tells Arc which environment string to use. When an option is executed, Arc uses the DOS environment to store information about the current user and their position within the menu. To avoid conflicts with multiple invocations of Arc, you may use this parameter to change the environment string Arc uses for this purpose.

e.g.

```
ReStartInfoEnvStr=MENU
```

ScreenSaver=*command*

Default: None (commented out)

An external program may be used as a screen saver through the use of this setting. When the screen save time-out is reached, the command assigned to this setting is executed instead of Arc's own screen saver.

- 2 It is the responsibility of the screen saver program to return the screen to the correct text mode (if it changes the video mode).

This method may be used to get Arc to invoke a third-party or user-written screen saver. It can also be used to get Arc to invoke tape backup utilities, hard disk unfragmenters etc.

e.g.

```
ScreenSaver=c:\arc\xmartin.exe -perturb
```

If the screen saver that you are using relies on a batch file, then you must invoke COMMAND.COM to run the batch file. For example, if you had a screen saver batch file called C:\SCRNSSAVE.BAT, you would specify it as follows:

e.g.

```
ScreenSaver=%COMSPEC% /c c:\scrnsave.bat
```

The %COMSPEC% is just another way of saying COMMAND.COM, except if you use a shell other than COMMAND.COM, COMSPEC will change to reflect this.

TemporaryBatchFile=*pathname* [.BAT]

Default: \$TEMP .BAT (in the current directory)

This setting tells Arc the name of the batch file that will contain the commands for executing the option selected by the user. This temporary batch file is then executed by ARC.BAT. A .BAT extension is added automatically.

e.g.

```
TemporaryBatchFile=c:\$temp\menu.bat
```

- 2 If you change this setting, you will have to change your ARC.BAT so that it executes a different temporary batch file.

TextEditor=*text editor*

Default: BEDIT (in the current directory)

This setting tells Arc which text editor to invoke when an option's commands or an item's user help are being edited. When invoking the editor, Arc provides the full path name of the file being edited as the first parameter. If an error in an option's batch file is being corrected, the third and fifth parameters are the row and column where the error was found.

e.g.

```
BEDIT m_option.b03 /ROW 3 /COL 18
```

If the text editor is not in the current directory or DOS's path, you must specify an alternative path and program name.

e.g.

```
TextEditor=c:\utility\bedit.exe
```

If you wish to use an editor beside the supplied text editor BEDIT, you must specify it using this setting.

e.g.

```
TextEditor=c:\dos\edlin
```

You may also use this setting to specify the name of a batch file that causes your text editor to be executed, but if so you must invoke COMMAND.COM to run the batch file. For example, if you had an editor batch file called C:\EDIT_IT.BAT, you would specify it as follows:

e.g.

```
TextEditor=%COMSPEC% /c c:\edit_it.bat
```

The %COMSPEC% is just another way of saying COMMAND.COM, except if you use a shell other than COMMAND.COM, COMSPEC will change to reflect this.

TextEditorFlags=*text editor flags*

Default: /row @ROW /col @COL

When Arc detects an error in an option's batch file, it can tell the editor being used to place the cursor at the position of the error (if the user chooses to edit the batch file). To allow for the various methods of supplying this information to the editor, Arc provides two macros:

@ROW the row or line of the error

@COL the column of the error

So if your editor required a /POS parameter, you could specify

```
TextEditorFlags=/pos(@ROW,@COL)
```

UserToLogin=*user ID*

Default: None (commented out)

This setting enables you to skip the log-in screen by automatically logging in a given user. Arc scans the user file for the user ID given, and if found, that user is logged in automatically.

This feature may be used on a network to facilitate the sharing of user information between the network and Arc

e.g.

```
UserToLogin=%USER%
```

Skipping the Login Screen

By specifying this setting without a value i.e.

```
UserToLogin=
```

and having a user account with a blank user ID field, you can cause Arc to completely bypass the log-in screen.

The precedence for log-in methods is as follows:

- 1) Environment string information
- 2) ARC.INI information
- 3) Log-in screen

VideoMode=[Detect | VGA | MCGA | EGA | CGA]

Default: Detect

This setting is used if Arc does not correctly determine your video card type. Simply use the value that correctly identifies your system's video card. Valid values are CGA, EGA, MCGA and VGA and DETECT.

e.g.

VideoMode=EGA

User Help

User help can be created by the administrator to provide information about the purpose and usage of each item in the menu. To display the user help for an item, move the cursor bar onto the item and press **bl**. The user help information may be scrolled using **w**, **y**, **{**, **}**, **g** and **d**. When you have finished viewing the information, press **|** or **f**.

If the user has editing privileges, they can press **aE** to edit any existing user help information. If no user help information exists, they are given the option of creating it.

Editing User Help

When you choose to edit or create user help information, Arc invokes the editor with any existing user help information (none if you are creating new user help). It places a margin at the top and bottom of the page to show you the text width limit of 48 characters. These ruler lines are automatically removed by Arc.

It is important that the editor you specify does not save special characters in its files. BEdit and the MS-DOS editor EDIT both adhere to this requirement. They both save the file as ASCII text.

Export / Import

Arc comes with two additional programs, EXPORT and IMPORT, that allow you to make major modifications to your menu structure by editing it in its entirety in an ASCII text file format.

Export

Export takes M_STRUCTURE.Dxx and creates one of a number of different reports, described further below.

Usage: EXPORT [*path*][*number*] [/*format*[**F**]]

path is the (optional) directory showing where the structure files reside. When not given, the path is assumed to be the current directory

number is the (optional) number of the structure file. When not specified, it is assumed to be 00. In the text that follows, this number is referred to as **xx**

format the report format. This is either 1 or 2. If followed by an **F** it forces EXPORT to overwrite any existing output file when it would normally stop. If omitted, this parameter defaults to /1 i.e. Detailed structure report with no overwriting

Each type of report is output to a file with a different name. If the file already exists, EXPORT will not continue unless you have provided the extra **F** parameter which forces overwriting.

e.g.

```
export /2f c:\arc\02
```

forces EXPORT to overwrite the file C:\ARC\M_TREE.D02 if it exists, producing a graphical tree report.

e.g.

```
export /1 > list.txt
```

directs the detailed structure report of the default structure (structure 00 in the current directory) to the file LIST.TXT.

/1 Detailed Structure

This report is used to convert an entire menu structure to an editable ASCII text file form. Once edited, this file can then be read by `IMPORT` to regenerate the appropriate menu structure files. This report is output to `M_EXPORT.Dxx`. Note that comments can be included by beginning each line with a semi colon `;`.

e.g.

```
;EXPORT v4.50 detailed structure file
;Created at 12:40:27 pm on Tuesday 4 June, 1996
;from C:\MENU\M_STRUCT.D00

#MENU Main Menu
XPOS=6
YPOS=7
MENUITEMS=

    #DIVIDER

    #MENU ^Applications Programs
    XPOS=10
    YPOS=9
    SHORTCUT=Ctrl+A
    MENUITEMS=

        #DIVIDER

        #OPTION WordStar ^2000
        SHORTCUT=Alt+2
        COMMANDS=
        C:
        CD\WS2000
        WS2
        ###ENDCOMMANDS###
        #ENDOPTION

        #OPTION ^XTree Gold
        SHORTCUT=Alt+X
        COMMANDS=
        C:\XTGOLD\XTGOLD
        ###ENDCOMMANDS###
        #ENDOPTION

    #ENDMENU (Applications Programs)

    #OPTION Disk cop^y
    COMMANDS=
    [VARIABLE(source[2]="a:","Source drive:")
     VARIABLE(dest[2]="a:","Destination drive:")]
    DISKCOPY %source% %dest%
    ###ENDCOMMANDS###
    #ENDOPTION

#ENDMENU (Main Menu)
```

Export Language

When producing a detailed structure report, `Export` uses a 'language' to describe the various types of menu item and their fields. Sections of text that you can change are shown in *italics*. In an exported menu structure, sections that have the default value are not output.

A **divider** description is simply

```
#DIVIDER title of item (any hot keys are ignored)
```

An **option** description is

```

#OPTION title of item (with optional hot keys)
AUTHORITY=authority level required, 0-99. Default 0
SHORTCUT=short cut, None if no short cut. Default None
RESETPRECOVERY=Y to enable, N to disable
PAUSEAFTEREXEC=Y to enable, N to disable
PROMPTFORLOG=Y to enable, N to disable
EXITARC=Y to enable, N to disable
HELP=
User help information. Default nothing.
###ENDHELP###
COMMANDS=
Batch file commands. Default nothing.
###ENDCOMMANDS###
#ENDOPTION

```

A menu description is:

```

#MENU title of item (with optional hot keys)
XPOS=The menu window's upper left screen x-coord
YPOS=The menu window's upper left screen y-coord
AUTHORITY=authority level required, 0-99. Default 0
SHORTCUT=short cut, None if no short cut. Default None
MAXHEIGHT=link window's maximum height. Default 0
HELP=
User help information. Default nothing.
###ENDHELP###
MENUITEMS=
All of the items in this menu. Must be included.
#ENDMENU (optional menu title repeated in brackets)

```

A link description is:

```

#LINK title of item (with optional hot keys)
DRVPATH=path to linked structure
NUMBER=number of linked structure. Default 0.
AUTOLINK=link immediately, Y or N
AUTHORITY=authority level required, 0-99. Default 0
SHORTCUT=short cut, None if no short cut. Default None
XPOS=The link window's upper left screen x-coord
YPOS=The link window's upper left screen y-coord
MAXHEIGHT=link window's maximum height. Default 0
HELP=
User help information. Default nothing.
###ENDHELP###
#ENDLINK

```

/2 Structure Tree

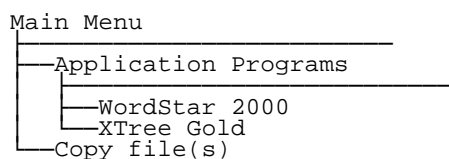
This report provides a graphical tree showing the hierarchical relationship of your menu structure. It is very similar to a tree display of the directories on a drive. It is output to M_TREE.Dxx.

e.g.

```

;EXPORT v4.50 menu structure tree
;Created at 12:40:27 pm on Tuesday 4 June, 1996
;from C:\MENU\M_STRUCT.D00

```



Import

Import takes the (edited) detailed structure file M_EXPORT.Dxx and, after checking it for errors, produces M_STRUCT.Dxx.

Usage: IMPORT [*path*][*number*] [/F]

path is the (optional) directory showing where M_EXPORT.Dxx resides. When not given, the path is assumed to be the current directory

number is the (optional) number (xx) of M_EXPORT.Dxx. When not specified, it is assumed to be 00

/F this parameter may be used to force Import to overwrite existing M_STRUCT.Dxx files. Without this parameter, Import will not overwrite existing files

e.g.

```
import c:\arc\7 /F
```

converts the detailed structure file C:\ARC\M_EXPORT.D07 to Arc's data format in C:\ARC\M_STRUCT.D07, overwriting it if it already exists.

Error Messages

This section provides descriptions of error messages you may encounter while using Arc. Information enclosed in square brackets [] represent sections of the error message that may change.

I/O Errors

A number of critical errors can occur which prevent Arc from continuing:

2. File not found

The file does not exist in the specified directory.

3. Path not found

The specified path does not exist. Check to see if the path you have specified is correct.

4. Too many open files

The CONFIG.SYS file includes a FILES=xx entry, where xx specifies too few files for Arc to operate correctly. Increase the number xx to some suitable value, such as thirty (30).

5. File access denied

The file has its read-only/system/hidden attribute set, preventing Arc from deleting, renaming or writing to this file

or

Arc is creating a file in the root directory and the root directory is full

or

Arc is renaming a file and a file with the new name already exists (this can only occur when Arc cannot delete the original file of the new name).

100. File is wrong format

Arc unexpectedly encountered the end of the file when attempting to read this file. The file is of the wrong format, and may have been corrupted. You can

recover from this error by deleting M_STRUCTURE.Dxx, and then renaming M_STRUCTURE.Bxx to M_STRUCTURE.Dxx.

101. Disk full

While writing to this file, the disk became full. Increase the amount of free space on the disk by removing some other files.

150. Disk is write-protected

Arc cannot write to the specified disk. This is probably due to a write-protect tab on the disk. If you are sure you wish to write to the disk, remove the tab.

151. Internal DOS error

The DOS device driver reported an internal error.

152. Drive not ready

There is no disk in the drive, or the drive door is open. Alternatively, the drive may not be powered-on or on-line.

157. Unknown media type

DOS can't recognise the type of disk in your drive. This message usually indicates you have placed an unformatted disk into the drive or the hard disk has not been formatted.

Any of the following error messages may indicate you have a defective disk or drive:

154. Data CRC error

DOS detected an error in the recorded data on your disk.

156. Disk seek error

158. Sector not found

DOS reported an error while attempting to find data on your disk.

160. Device write fault

DOS encountered a hardware error when attempting to write to your disk.

161. Device read fault

DOS encountered a hardware error when attempting to read from your disk.

162. Hardware failure

DOS encountered a hardware failure on your disk drive.

Arc-Specific Exit Codes

The following exit codes are specific to Arc:

0. User leaving Arc

This occurs when the user leaves Arc for DOS.

32. Primary structure invalid

This occurs when Arc encounters an invalid (corrupted) primary structure file (the first structure read). To revert to the backup, first delete M_STRUCTURE.Dxx. Then, rename M_STRUCTURE.Bxx to M_STRUCTURE.Dxx.

33. No USER_INF.DAT file

Arc needs a user definition file to tell it the security status of the people using it.

203. Out of memory. Terminating

Arc has run out of memory, and cannot continue correct operation. This may be due to too many linked structures, menu nesting too deep, structures too large etc. Free some memory e.g. by removing TSRs.

254. Run temporary batch file, then exit

The user has selected an application to be run, the batch file should CALL the temporary batch file to run the commands in it. The batch file will not return to Arc because the "Don't return to Arc" flag has been set.

255. Run temporary batch file

The user has selected an application to be run, the batch file should CALL the temporary batch file to run the commands in it.

Could not find the Help file [SYS_HELP.TXT]

Help I was invoked, and Arc was unable to find the system help file. Check the help file exists, and is in the expected directory. SYS_HELP.TXT can be copied from your original disk.

Warning! Insufficient environment space

Arc uses the DOS environment to store information about the current user during the time it is absent from memory. If there is not sufficient space to do this, it provides this warning message. Increase your environment space by reducing the number and size of environment variables (such as your path), or by increasing the environment space using the CONFIG.SYS SHELL command (see your DOS manual).

Unable to find the topic [item] in [SYS_HELP.TXT]

Help I was invoked and Arc found the help file, but it was unable to locate the specified topic.

Testing Exit Codes

Errors that cause Arc to terminate can be trapped using DOS's **ERRORLEVEL** command within a batch file

e.g.

```
:
arc arc.ini /Z
for %%e in (0 30 32 33 202 203 254 255) do if errorlevel==%%e goto
_LABEL%%e
echo Unknown exit code
goto end
:_LABEL254
call %TEMP%\$temp%LB%.bat
del %TEMP%\$temp%LB%.bat
goto end

:_LABEL33
echo User file missing or corrupted
goto end

:_LABEL0
goto end

.. etc ..

:START
@ECHO OFF
rem ** Put initialisation code down here to speed up batch file **
c:
cd\arc
goto BEGIN

:end
rem End commands go here
```

Quick Reference

	Key	Action
Cursor Movement	w	Up one item
	y	Down one item
	{	Up six items
	}	Down six items
	f	Enters menu or executes item under cursor bar
	x	Enters menu under cursor bar
	^	Returns to the previous menu or exits Arc
	z	Returns to the previous menu, if any
	g	Moves to first item in menu
	d	Moves to last item in menu
	l	System help
	bl	User help for item under cursor bar
	t / uE	Edit item under cursor bar / any item
	u	Command menu
	h, ul	Inserts a new item
	c, uD	Delete any item
	uA	About
	uP	Preferences
	uS	Scheme
	uM	Move item under cursor bar
uC	Copy item under cursor bar	
uW	Move menu window	
uU	User Accounts	
uF	First menu name	
uX	Exit	
Edit	y	Show available/used short cuts (on Short cut field)
	t	(For options) Edit batch commands
	u	<i>Options:</i> AutoBuild batch commands <i>Links:</i> AutoBuild link structure information
User Accounts	f	Edits a user's account
	h	Inserts a new user account based on the current highlighted account
	c	Deletes a user account

Show Short Cuts	13
Skipping the Login Screen	58
Statements	
CheckBoxes	42
CommandButtons	40
EditField	39
MustExist	38
RadioButtons	41
Verify	38
Status line	5, 7
Structure	7, 20, 25, 55, 60
Structure Consistency	49
Sub-strings	37
Supervisor	3, 18, 23, 33
Swap Mouse Buttons	13
System Files	48
System help	7, 48
System Requirements	2
System Security	46

T

TemporaryBatchFile	56
Testing Exit Codes	67
TextEditor	57
TextEditorFlags	57
Time Format	12
Title	21
Title Box	14

U

Used short cuts	22
User	
Allow editing?	34
Allow menu ESC?	33
Authority	33
Edit User	33
Login user	34
Password	33
User ID	33
User name	33
User Accounts	32
User help	7, 59
User ID	4, 10, 33, 55, 58
User Name	33
UserToLogin	58
Using a Mouse	7
Using a Variable's Value	37
Using Arc	4

V

Verify	38
VGA	58
VideoMode	58