ASCII-Manual

Written by ’95

COLLABORATORS

	TITLE : ASCII-Manual		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Written by ’95	August 25, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

ASCII-Manual

1.1 The Complete Guide To ASCII-Editing

```
                Welcome to the ASCII Guide!

This guide is the manual for three programs from The Problem. Choose one
here:



                 The ASCII-Editor


                 The ASCII-Config Program


                 The ASCII-Tracer



                 The ASCII Text Format

                 The Script Format

                 The Guide Format

                 Bugs
* This guide was created with help from the ASCII-Editor.
```

1.2 The ASCII Editor

The Problems' ASCII Editor

Create ASCII text files, Scripts or even AmigaGuide databases that look like professional documentation with ease! This program was designed to make it easier to include 'complicated' ASCII drawings inside textfiles.

```
On startup you will   |o| Tools    |
be presented with a   |
                T1
                T2
                T3
                GB
                |
window  that  looks   |
                T4
                T5
                T6
                PB
                |
quite like  the one   |
                T7
                T8
                DM
                *
                |
you see here:         |
                 | | | | |
                |
                      |
                @|Front
                CLR
                |
                      |
                 Fix
                Undo
                |
                      |
                Load
                Save
                |
                      |
                Redrw
                Quit
                |
                      +------------+
```

* Extra functions of ASCII-Edit can be accessed through the
Menus
.

NOTE:
You can NOT load in SCRIPT files. Be sure to save a project file of all your work, specially when you want to continue it at later stage!

1.3 menus

```
                The Project Menu
- New    : Initialize a new project.
- Load   :

                - Save   :

                - Undo   :

                - About  : The About Requester that tells you what you are working ↩
                    with.
- Quit   : Quit the Editor (not recommended for unexperienced users!)

The Options Menu
- Draw Mode  :

                - Display  - Redraw         :

                           - XFlip          : Mirror the entire project  ↩
                              horizontally.
           - YFlip          : Mirror the entire project vertically.
           - Fill Screen    : Allows you to fill foreground or background.
           -  As AmigaGuide : Check  this  item  to  force the  display  to
                              AmigaGuide  style.   AmigaGuide  leaves  room
                              between every line so it can draw its buttons.

- Change Colour :  Brings up a window you can use to Replace or Swap
                   all  appearances of one colour by onother colour.

- Recol Hidden Chars : If  the front colour and back colour of a
                                character  is  the  same, you can not see
                                it.   This function allows you to rapidly
                                change the front/back colour of all those
                                hidden characters or kill them.

The Tools Menu
- Freehand... : Brings up a window where you can decide what happens
                         when you draw using the Freehand Tool
                         (the first tool, top left)

- Draw Box... : Brings up a window where you can decide what happens
                         when you draw using the Draw Box Tool
                         (the second tool in the second row from the top)

- Airbrush : Use this to set the size and speed of the AirBrush tool.

- Text    - Font Style : Use  this  to activate Bold,
                                  Underlined and/or Italic modes
                                  for the foreground data.
                   - Search     : Brings up a window in which you can perform
                                  several   search   actions   (like   search,
                                  replace etc.
                   - Find Next  : Use this, it's fun!
                   - Replace Next : Dito, though it's slightly less fun.
- Brush     - Show    : Shows the current brush
           - XFlip   : Mirrors brush horizontally
           - YFlip   : Mirrors brush vertivally
           - Free    : Removes current brush from memory
           - Options : Brings  up  the Brush Options window where you can
                       choose what brush information you want to use when
                       pasting.   Also  you can XFlip or YFlip the brush,
                       Invert  its  colours  or  Swap  its background and
                       foreground  colours,  Save the brush or Load a new
                       brush.
```

1.4 Freehand Tool

The Freehand Tool

The Freehand Tool is used to draw onto the screen as you would in any other paint package. You can change the way the Freehand Tool works by selecting one of the Special Freehand modes from the Tools/Freehand... menu. There is one normal freehand mode, two special modes and a combined special mode.

* In the foreground the Freehand Tool uses the character, currently displayed in the Brush Display in the Tools window.

* In background mode it will change a single character's background colour.

* The way a character responds to a freehand drawing action depends on the settings in the Drawmode window.

1.5 Fill Line Tool

The Fill Line Tool

This tool will perform a floodfill, but in a special way. It will only fill up horizontal lines in either foregound or background mode, using the settings from the Drawmode window.

* In foreground mode it will fill the selected line in the foreground, using the activated colour(drawmode) and the current brush character.

* In background mode it will fill the selected line in the background changing it's colour depending on the settings in the drawmode window.

1.6 Airbrush Tool

The Airbrush Tool

The Airbrush Tool will act like you are painting with a spraycan that is filled with text. You can change the behaviour of the Airbrush Tool in the Tools/Airbrush... menu option.

* In the foreground this tool will spray the current brush character, using the current colour(drawmode) using the speed and radius you set in the Airbrush window.

* In background mode it will spray in the background, not changing any text in the foreground. What exactly happens depende on the settings in the

Drawmode
window.

* In the Tools menu, you can use the Airbrush... option to set the speed and the size of the Airbrush Tool.

1.7 Straight Line Tool

The Straight Line Tool

Use this tool to draw straight line from one point to another in a similar way to other paint packages. To draw a line simply click and hold the left mousebutton at the starting position and then drag the mouse to the ending position. Then release the mousebutton and the line is drawn.

* In the foreground the Line Tool uses the current brush character, colour and/or drawmode settings. It will not affect the background colours on the screen.

* In the background this tool uses the current colour/drawmode settings to change the background.

1.8 Draw Box Tool

The Draw Box Tool

The Box tool can be used to draw a box, either open or filled in either the foreground or the background. There are two ways of drawing boxes useing this tool. You can toggle between them using the Tools/Draw Box... menu option.

* To draw an open Box, activate the Box Tool by clicking on the upper half of the button, to draw a filled box click on the lower half.

* In foreground mode the Box Tool uses the current brush character or the special mode (depending on the settings in the Tools/Draw Box... option). The result also depends on the settings in the Drawmode window and/or the currently activated colour.

* In background mode this tool only works if the special Draw Box mode is turned off (see above). It uses the current colour/drawmode settings to change the background.

1.9 Draw Ellipse Tool

The Draw Ellipse Tool

The Ellipse tool can be used to draw an ellipse, either open or filled in either the foreground or the background.

* To draw an open ellipse, activate the Ellipse Tool by clicking on the upper half of the button, to draw a filled ellipse click on the lower half.

* In foreground mode the Ellipse Tool uses the current brush character. The result depends on the settings in the Drawmode window and/or the currently activated colour.

* In background mode this tool uses the current colour/drawmode settings to change the background.

1.10 Text Edit Tool

The Text Edit Tool

Use this tool to edit foreground text using the keyboard. This tool does not depend on wether you are in foreground or background mode, it always works in foreground mode since background text hasn’t been invented yet.

* Note that the Text Edit Tool does not perform any action that involves moving text around. This is to prevent you from destroying previously drawn foreground ’graphics’.

Therefore you will always be in overwrite mode. Insert mode would require shifting text to the right if you insert something in the midle of a line.

The Backspace key DOES delete the character preceding the cursor, but it will not shift the rest of the line to the left.

The Delete key WILL delete the character at cursor position but it will NOT shift the rest of the line to the left.

The Return key will return the cursor to the starting position at the next line, but you can not insert lines by pressing Return.

* To move text around you’ll have to pick it up as a brush and then go to the Brush Options window (via the Tools/Brush/Options menu) and set the Brush pasting mode to ’Front Only’. Then paste your text down wherever you want it. (Be sure you are in the ’Normal’ drawmode)

1.11 Clear Tool

The Clear Tool

Actually this is the same as the freehand tool, but instead of adding foreground or background information, you can use the Clear Tool to erase background AND foreground information. This tool does not depend on wether you are in foreground or background mode.

* Note that the Clear Tool clears the background to the currently activated colour. The foreground info is removed.

1.12 fix

The Fix Button

Press this when you are satisfied with what you created so far. This will update the Undo buffer. After you Fixed the changes you can not Undo them anymore. Any Undo action after this will restore your project to the state it was in when you last pressed Fix.

1.13 undo

The Undo Feature

Use this function to undo everything you changed since the last time you pressed the
Fix
button. You can change the name of the Undo file in the ASCII-Edit Configuration editor.

1.14 col

The Colour Gadgets

These are the 8 colours you can use. Simply select one by clicking on it. Activating a colour, sets that colour to be the currently used colour in the area you are editing in (Foreground or Background).

* Active colours are only used if the settings in the Colourmode window allow this.

1.15 frontback

The Front/Back switch

This switch is used to toggle between foreground (Front) and background (Back) modes. In foreground mode all tools will draw using the current ascii character and colour(mode). In background you can change the fieldcolour of any character on the screen using any tool.

1.16 clr

Clear Project

In the window that appears, select what information you want to clear. The projects' size will not change (even when you choose 'All'). To change the size, you must initialize a new project using the Project menuoption New.

1.17 redraw

Redraw

This simply forces ASCII Edit to redraw the entire screen.

1.18 drawmode

The Draw Mode Window

Selecting this feature brings up a window titled 'Draw Mode'. Use this window to determine what happens to the screen if you draw something. Be aware that the drawing mode works for Background or Foreground, although some are only useful in foreground mode.

The CycleGadget can be used to select one of the following modes:

Normal (Use Active)
This means that the currently activated colour will be used when you draw something using one of the tools.

Invert Bg or Fg
When you draw something, you won't be adding any new characters. What will happen is that the colour of all characters you use are inversed. This is done by substracting the number of the colour at the drawing position from the maximum number of colours available (8). You may use the 'Use x Colours' CycleGadget' to change the maximum number of colours used for this inverse mode (you can still use all 8 colours, but the outcome of an inverse calculation will be different).

Swap Bg<->Fg
When drawing, you won't be adding any new characters. In stead, all characters you touch will have their foreground and background colours swapped.

Recoulour To Active
This is one of the most usefull draw modes! In stead of adding new characters by drawing, only the colour of the existing characters will be changed.

Erase Bg or Fg
This wil erase all characters that you touch when drawing.

1.19 backfront

The Area Switch

Use this CycleGadget to switch between the two working areas.

Back : Background mode. In this mode you will be able to draw in the background area of the screen. This means that only 'Fieldcolours' will be affected by the drawing operations. All foreground text will remain the same as it was, so you can edit background without damaging the foreground.

Front : Foreground mode. Or Textmode if you like. All drawing operations will take place in the foreground area. The foreground area can contain any ASCII character, the empty space in each character will be considered to be background (so a Space will have no effect on the foreground, except that it will erase the old character if there is one).

* The Text tool will always be in foreground mode (of course).

* The Replace Colour function from the Options menu will only operate in the activated area. (Front/Back)

* The settings of this CycleGadget have no effect on the brushtools, for that you need to select the Brush Options item form the Options menu. (Left-Amiga B)

* All tools will only operate in the activated area.

1.20 getbsh

Pick Up A Brush

Press this to activate the Brush Grabbing feature. Now, while holding down the left mouse button, draw a box around the area you want to grab as a brush. This will pick up all information inside the box (Foreground, Background, Colours, Fontstyles, everything)

* After grabbing, the 'Use Brush' tool is automatically activated in the Tools window.

1.21 usebsh

Paste Current Brush

Press this to activate the Use Brush tool. This is only useful when you have already grabbed a brush once. When activated you can use the last grabbed brush which will be in memory until you pick up a new one.

If you don't want to use all the brushs' information to paste down, use the Brush Options feature from the Options menu. If there is a brush, it will be displayed in the top left corner of the Edit window, surrounded by a black border with the title 'Brush'. A window will appear (Brush Options) where you can decide what brush information to use. (Background, Foreground or Both) Also from this window you can XFlip or YFlip the brush, Invert its colours or Swap its Background and Foreground colours.

IMPORTANT:
Be aware that the currently activated Background Colour will not be visible in the brush if it is pasted down. If you picked up a brush with black in it, and you have activated black as the background colour then, when pasting down the brush somewhere, all the black will be ignored. If you want black to be displayed when the brush is pasted, you should activate another background colour, and then paste the brush. You may always change the background colour.

1.22 tgad

The Brush String Display

In this display you can see which character is currently used as a brush for all the tools (except for the Text tool, and the Paste Brush Tool).

You can change it pressing any key on the keyboard during editing. If you are in Background mode this display will show a cursor in the currently activated colour.

1.23 load

The Load Window

After selecting Load, you will be presented with a filerequester. Use it to select a file. ASCII-Edit will try to determine what type of file you have selected. The following types are known:

Custom Project files (.proj)
If you selected a project file, a requester will ask if you are sure. Note that loading a project means that all current information will be lost (if you didn't save it)!

Custom Brush files (.BSH)
When a brush file was selected it will be loaded as such. ASCII-edit will automatically activate the 'Use Brush' tool in the tools window. You can then paste the brush anywhere you want it. It will remain in memory until

you pick up a new brush, or load onther one.

AmigaGuide files (.Guide)
Selecting an AmigaGuide database will bring up a window where you can select a node. For more info about this window, read
Loading Guides
.

Textfiles (ASCII)
If the selected file was none of the above, it will be loaded as a text file. A window will pop up. Use this to select which area to load. If the file does not contain colour information (escape-codes), you can enter a default colour in this window. Unknown escape codes will be ignored.

NOTE:
Text and Guide files will be Merged into the current project if you load front or back only. This way you can load in the foreground of one file, and the background of another.
But if you load a guides' foreground and background (both), the project will be adjusted in size, which means that it will be cleared before loading (se no merging there).

1.24 save2

The Save Window

This brings up the save window. Use the CycleGadget to select the fileformat (Project, ASCII, Script or AmigaGuide) and use the checkboxes to select what to save. The settings of these checkboxes will not affect the Project fileformat!

1.25 quit

Quit

This will quit the program. You can also press the close gadget of the window (or select Quit from the Project menu in the Ascii-Editor).

1.26 The ASCII Tracer

The Problems' ASCII Tacer

You can use this program to convert any IFF ILBM picture to an ASCII Edit Project. After startup you will be presented with the window as shown below. Click the buttons for more information.

```
                              |  | ILBM_2_ASCIIEdit     |  |
                              |                            |
                              |   Convert:
                        Screen
                     |
                              |                            |
                              | <Room for errormessages>   |
                              |                            |
                              |                            |
                              |
                    Convert

                     Save

                     Exit
                     |
                              |____________________________|
```

Pictures will be converted to Background-only projects, which means that there will be no text in the file. You can add text using the ASCII Editor. This is a Beta-Test version of the ASCII Tracer. Do NOT SPREAD! Try it, and you're an EX-Problem member!

1.27 screenbox

The Screen/Box Switch

Use this CycleGadget to select either "Screen" or "Box" mode:

```
Screen: Convert the loaded ILBM entirely, resizing it to fit the window.
Box   : You can draw a box around the area you want to convert. The
            contents of the box will be sized to fit the window.
```

1.28 convert

Converting An Image

Pressing this button will bring up a filerequester in which you may select the source IFF ILBM picture. ASCII text can contain no more than 8 colours, so ASCII Trace will only use 8 colours. If a picture contains more, they are simply looped around (colour 8 to 0, colour 9 to 1, colour 10 to 2 etc.)

After selecting a picture a window the size of the screen will appear. Resize this window to the size you want the project to be. In the middle of this window you can see the width and height in characters. When finished, press the 'Done' button in the top left corner.

A window showing the loaded picture appears. If you selected Box-mode rather than Screen-mode, you may now draw a box around the part of the picture you want to convert, else the conversion will start immediately. When the conversion is finished you will be back at the startup screen.

You can now save the project using the Save button.

1.29 save

Saving A Project

After pressing the save button, you will be presented with a filerequester. Use this to select a savename for your project. The file will be saved in ASCII Edits' custom Project format, so you can load it into the ASCII Editor for further editing.

1.30 The ASCII-Config Editor

```
|  | ASCII Editor Configuration                                        |  |
|  Activated Tool:
                 1
                   Default Mode:
                @|Foreground
                 Horizontal:
                 -
                 |
|Activated Colour:
                 1
                        Display:
                @|  Normal
                   Vertical:
                 |
                 |
|   Project Width:
                 77
                 Airbrush Speed:
                @|   Fast
                    Corner1:
                 +
                 |
|  Project Height:
                 29
                    Open Window:
                @| No Window
                  Diagonal1:
                 /
                 |
|   Airbrush Size:
                 2
                  Freehand Type:
                @|  Normal
                  Diagonal2:
                 \
                 |
|    Default Path:
                 Work:
                       Box Type:
                @|  Normal
                    Corner2:
                 X
                 |
|       Undo File:
                 Clips
                                                    Brush:
                 *
                 |
|                                                                          |
|
                    Save

                    Quit

                 (C)1995 by The Problem Coding Dept.
                |
|                                                                          |
```

1.31 Configuration Editor (information bar)

The Information Bar

This textgadget will display information on every button. This only works when this info is in the config-file (preceeded by a semi-column ';')

To show info, simply press any button in the window, and there it is!

1.32 Configuration Editor (default numbers)

The Numeric Default Settings

The first column of gadgets are for setting the default numeric variables. To change, simply click the gadget and type a new value. If the value is too high it will be changed to the maximum value. If your value is too low or otherwise invalid, the lowest possible value will be automatically set.

1. Activated Tool:
Can be a value ranging from 1 to 10. On startup of the ASCII Editor the tool with the specified number is already activated. In the information bar you can see which tool you activate after you filled in a number. The following tools are available:

```
        1 = Freehand                        6 = Draw Box
        2 = Fill Line                       7 = Draw Ellipse
        3 = Airbrush                        8 = Paste Brush
        4 = Pick Brush                      9 = Text Edit
        5 = Straight Line                   10= Clear
```

2. Activated Colour:
Can be a value ranging from 0 to 7, specifying which colour should be

active on startup.

3. Project Width:
The default width of the project (in characters). The minimum value is 77, the maximum is 250

4. Project Height:
The default height of the project (the maximum nnumber of lines). the minumum value is 29, the maximum is 250.

5. Airbrush Size:
A value ranging from 1 to 10 which sets the ’size’ parameter of the Airbrush Options window in ASCII-Edit.

1.33 Configuration Editor (cyclegadget defaults)

The Cyclegadget Defaults

The second column of gadgets are for setting the default settings of some cyclegadgets in ASCII Edit. To change, simply click the gadget to cycle through the options.

Default Mode:
Switch between Foreground and Background
To set the Front/Back switch in the Tools window of ASCII-Edit.

Display:
Switch between Normal and AmigaGuide
To set the display mode in the Options/Display menu.

Airbrush Speed:
Switch betweem Fast and Slow
To set the Airbrush Speed switch in the Airbrush Options window.

Open Window:
Cycle through No Window, Load and Init New
To set which window you want to be opened on startup:

```
No Window :will do nothing.
Load      :will open the Load window after the About requester is
           closed.
Init New  :will open the Init New window after the About requester
           is closed. This is the window you get when you select
           the Project/New menu option.
```

Freehand Type:
Cycle through Normal, Special1, Special2 and Special1&2
To set the Freehand Mode cyclegadget in ASCII-Edit.

Box Type:
Switch betweem Normal and Special
To set the Box Type cyclegadget in ASCII-Edit.

1.34 Configuration Editor (text defaults)

The Text Defaults

The rightmost column of gadgets are for setting the default settings of some functions in the ASCII Editor that need text supplied to them. To change, simply click the gadget and type the new (single character-)string.

What they mean
Horizontal: ('-')
The character you choose will be used for specialmodes in the Freehand and Box tools of ASCII-Edit. The 'Horizontal' character will be used for drawing horizontal lines.

Vertical: ('|')
This character will be used to draw vertical lines in special drawing modes. (see above)

Corner1: ('+')
This character will be used to draw crossings between 'Horizontal' and 'Vertical' characters. It will also be used to draw the corners of boxes in the special boxdrawing mode.

Diagonal1: ('/')
This character will be used to draw diagonal lines in freehand drawing mode Special2 or Special1&2. It will be used when the mouse moved from bottom left to top right or vice versa.

Diagonal2: ('\')
This character will be used to draw diagonal lines in freehand drawing mode Special2 or Special1&2. It will be used when the mouse moved from top left to bottom right or vice versa.

Corner2 : ('X')
This character will be used to draw crossings between 'Diagonal1' and 'Diagonal2' characters.

Brush : ('*')
This character will be the active brush used by most of the tools in the ASCII Editor (like Freehand, Box, Circle, Line etc.).

1.35 Configuration Editor (default program path)

The Default Path

Here you can decide what the default path of ASCII-Edit should be. The path you enter will be used when you access a filerequester (for loading or saving). It will then read the directory you specify here.

1.36 Configuration Editor (name of undo file)

The Undo Filebuffer

In this stringgadget you should enter the complete path and the filename of the Undo filebuffer. When you press Fix in the ASCII Editor it will save the undobuffer to this file in the Project format.

Make sure that the undofile can always be accessed! It should not be on a floppy disk. My personal advise is to put it in the clipboard like this:

```
Clips:ASCII-Edit.UNDO
```

If you type an illegal name for the undofile (one that can’t be opened), ASCII-Edit will use the above default. When you quit the ASCII-Editor, the Undo filebuffer will be deleted.

1.37 Configuration Editor (save config file)

Saving The Configuration File

When you press this button the configuration will be saved in your S: directory as ASCII-Edit.Config

The file will be saved in a readable-by-human format. That means you may edit it using a text-editor. This is NOT recommended and should only be done when you don’t have the ASCII-Config program.

When you received the complete package, one configuration file is included. You should copy it into your S: directory before you start working with ASCII-Edit or ASCII-Config. This default file looks like this:

```
#000=001  ;Activated Tool, 01-10
#001=001  ;Activated Colour, 00-07
#002=077  ;Project Width, 77-250
#003=029  ;Project Height, 29-250
#004=002  ;Airbrush Size, 01-10
#005=001  ;Activated Mode, 00=Back, 01=Front
#006=000  ;Display, 00=Normal, 01=As AmigaGuide
#007=000  ;Airbrush Speed, 00=Fast, 01=Slow
#008=000  ;Window On Startup, 00=NoWindow, 01=Load, 02=Init
#009=000  ;Freehandtype, 00=Normal, 01=Special1, 02=Special2, 03=Special1&2
#010=000  ;Boxtype, 00=Normal, 01=Special
#011=-    ;Specialmode Freehand/Box Horizontal
#012=|    ;Specialmode Freehand/Box Vertical
#013=+    ;Specialmode Freehand/Box Corner 1
#014=/    ;Specialmode Freehand Diagonal 1
#015=\    ;Specialmode Freehand Diagonal 2
#016=X    ;Specialmode Freehand Corner 2
#017=*    ;Default Brush Character
#018=Work:Projects/Text/TP_FilesGuide/  ;Default Directory For Filerequester
#019=Clips:ASCII-Edit.UNDO  ;Name Of Undo-Buffer
```

If you do decide to edit it by hand, please make sure you have a backup copy. Furthermore make sure to keep everything on the same spot; don’t change 001 to 02, but change it to 002. ASCII-Edit does check if the values are legal, but if they're not funny things may happen. If you remove lines from the file it could even crash your machine. Also don't remove the text behind the data, since it is used by ASCII-Config's Information bar (the text is displayed there) You may however add comment lines. They must be preceded by a semi column ';'

1.38 Configuration Editor (quit)

The End

Well, you could use this to quit without saving by clicking this button before clicking the Save button, or you could use this button to Quit after saving, by clicking it after you saved the configuration... decide for yourself.

PS:
You could also ignore this function by using the close gadget of the ASCII-Config window, or by resetting (or turning off) your Amiga before quitting the program. ...HUH? -Ed.

1.39 The ASCII Text Format

ASCII Text

"What is ASCII anyway?" you may ask. Well, to put it simple: ASCII is the text you are now reading. Yes, ASCII are all those lovely letters and numbers that magically appear on the screen when you press the matching key!

Ofcourse in a normal textfile, you won't see more than about 80 different ASCII characters, but there are 256 of those characters! So, where is the rest? You can see a lot of them by typing this in the Shell: TYPE <file> where file should be any random (non-text) file. Now you will see that the screen starts flashing while strange-looking characters are all over the place... that's them! Some of them have special functions (like flashing the screen, scrolling a line of text (line feed) or make the cursor jump forward some steps). One of those special characters is character number 27, and that's the only character that ASCII-Edit needs to use for all those special texts!

Escape Codes

Character 27 is called ESC(APE) and is usually attached to the Esc key on
your keyboard. If you enter the Shell and press escape, you will see a
funny-looking character. If you use a normal font, it should be a square
bracket ([) with inversed colours (blank character on a black background).

The ASCII format uses this code for numerous things. I will only give you a few (mainly the ones that ASCII-Edit uses), to find out more you could just experiment some.

Cursor Movements...

The syntax is very simple. An excape-code starts with the ESC character followed by a normal square bracket ([), then numeric value and the 'name' of the code you are going to use...

For example: The code for making the cursor jump a number of steps to the right (without printing spaces) is C (not c, but C!). So, to make the curser move 10 characters to the right we type:

```
  [[10C         ([ is the escape character!)
```

That's all really. We could do the same movement to the left, using code D:

```
  [[10D  will do the job...
```

```
To move the cursor down 10 lines: [[10B

To move it back up again        : [[10A
```

I used the above 2 codes to do the scrolling of the logos in the Demo and the installer. What I did is this:

1. Create a variable called $ScrollUP containing 30 escape-codes for scrolling DOWN one line ([[1B).
2. Put the data that should be scrolled UP on the screen.
3. ECHO (print) the $ScrollUP variable.

The cursor will now move down the screen very quickly on a line by line basis. When it reaches the bottom it will scroll everything aboveit upwards! (This also happens when you reach the bottom of the screen when you are typing directly into the shell...)

To scroll stuff down the screen, I did the same thing, but now with 30 times [[1A ...

To place the cursor on a certain line from the top, you need escapecode H:

```
  [[10H   will put the cursor at the start of the 10th line. Now you can
              place it anywhere on that line using the C code (see above)
```

Colours and Fontstyles

The nicest code of all has got to be m (m, not M!). You can use it to determine text and field-colours (pens) and fontstyles (bold, underlined). It works exactly the same as the above codes with the difference being that you can not see the results of these codes immediately. You only set a colour for use until a new code is entered. Some codes can be combined, other's reset the old values...

If you type this in the Shell: [[1m you will notice that the prompt gets back in a bold fontstyle. 1m means you want bold mode turned on. To turn underline mode on, enter 4m in stead of 1m. Use 3m for Italic and 0m to reset everything to normal.

```
NOTE:  [[0m   will reset EVERYTHING to normal. All settings will be
                  overrided (including colour settings).
```

To change the colour of the text in the shell, use [[3{ub}#m. The # may be any number in the range of 0 to 7. Normally this is set to 1 (black):

```
  [[31m
```

If you don't want the character itself changing colour, but only it's field (background, paper, etc.) colour, use [[4#m. Again # may be any number in the range of 0 to 7. Normally this value is set to 0 (grey/blank):

```
  [[40m
```

If you change the field colour, everytime you enter a space, you will see a cursorsized block in the specified colour. If you would want to leave a blank space between 2 of those coloured blocks, you don't need to reset the colour to 0, simply use the cursor-jumping codes. It works for me!

ASCII-Edit's Saving Methode
(You don't need to read this. It's confusing. -Ed.)

ASCII-Edit saves ASCII-Files as followes:
If a foreground colour is different from the one currently in use, the colourchange will be written to the file as soon as the foregrounddata holds a character other than a space. So if you created a project with this line in it:

```
I created     this      project
```

ASCII-Edit will save the line as

```
[[32mI created    [[31mthis      [[33mproject
```

and NOT

```
[[32mI created[[31m    this[[33m      project
```

EXCEPT when you use Underlined. If you use underlined the escapecode will follow directly after underlined mode is turned off. Otherwise all spaces lying between two areas that contain text would be underlined. (?? Confusing -Ed.) Anyway, I won't go about explaining all this. If you really are that keen on knowing how ASCII-Edit saves ASCII-files, just save some ASCII, load it in an editor and study.

1.40 The Script Format

Amiga Scripting

If you already took a good look at the Australia project, you will have noticed that it contains a large amount of script files. Personally I think scripts are the best thing there are when it comes to ASCII-Editor usage. You can use them to link ASCII-text together or to create interesting and amazing Installers...

If you want to see what Scripting can do for you, just take a look at the Australia installer script. I know it has been deleted (By the installer itself, pretty clever if I may say so!), but there's a duplicate in the examples scripts directory of ASCII-Edit. Just load it into an editor like CED, or use a textviewer that can display texts without responding to ASCII escape codes (like dirwork) and read through it. If you don't get an idea or two from that...

Oh, yeah, just as clever as the installer, is the opinion poll script, with the cli-menusystem (Menu.script, also included) I'm really proud of! The menusystem was written without the need for ASCII-Edit ofcourse, but all the rest was created in ASCII-Edit! Anyway, if you haven't seen it yet... Go and do it! NOW! (and remember to send the results back to us!)

Working on this project was... learning more about ASCII, Scripts and AmigaGuides than I know about working on EP's...

You see this button? If you press it, you'll see how you should go about creating an ASCII-Animation, such as the Rotating Cube Anim example that came with the EP. If you don't, I'll be very angry. I might just get violent you know... I mean, I've spend almost an entire DAY to write that page, and you are just too damn lazy to read?! I won't accept that! Not from someone like you I won't! You hear me?! ....erm, sorry for that...

1.41 The AmigaGuide Format

Amiga Guide

Amiga Guide is so good because it's so incredibly easy to read. Reading a manual of a large program in AmigaGuide is much easier (and faster) than reading a loooong ASCII text.

Well you're reading one right now, so let's assume you know how it looks and what it does. The biggest disadvantage is the amount of colours you can use, namely only 4. You can use more, but you would probably have to mess around in the palette.prefs, using the Palette program. Well, these settings can vary with every user, and if you change something, the whole workbench looks like it has been involved in some kind of nuclear experiment. That's why ASCII-Edit will only save out 4 colours. That is, if you used 8 colours in ASCII-Edit to draw the image, only 4 will be saved (colour 4 remapped to 0, 5 to 1, 6 to 2, and 7 to 3). I'll tell you exactly what ASCII-Edit does and does not do in Guide Format.

The AmigaGuide Commands

The format of the AmigaGuide commands is simple. Every command always starts with a @. Some commands that have an immediate effect can be put right after this @. But most commands have to be put between { and }.

For instance, to change the font to topaz size 8, you type

```
 @FONT topaz 7
```

And to create a button called Wanda, that links to a node (=page) called WandaPage, you would type:

```
 @{" Wanda " LINK WandaPage}
```

But I won't go and explain all that, because ASCII-Edit has nothing to do with such commands! So, let's move on to something a bit more on the subject: bold, underline, colours and fieldcolours...

To make a word like "Hello" appear on the screen in a bold fontstyle, you use the b command (Yes! The b command!), like this:

```
 @{b}Hello
```

All text after @{b} will be bold, until you turn off bold mode again using @{ub} (let's call it unbold...) So if you type:

```
 @{b}Hello@{ub} Hello    you will see: Hello Hello
```

Now, the same thing goes for underlined mode. You can use @{u} to turn it on, and @{uu} to turn it off again. So if you type:

```
 @{u}Hello@{uu} Hello    you will see: Hello Hello
```

And ofcourse, the same goes for Italic, now just use @{i} and @{ui}. If you type:

```
 @{i}Hello@{ui} Hello    you will see: Hello Hello
```

Colours

Far more interesting are the colours! You saw the TP-Files.Guide? Well, there you have all the examples of the world of what you can do with AmigaGuide in 4 colours... The colours in the AmigaGuide format all have their own name. There are eight names (for there are 8 pens available in but as I said earlier, ASCII-Edit will only use four:

ASCII-Edit #	Description	Default	AmigaGuide Command
0 (4)	Background colour	Grey	Back
1 (5)	Text colour	Black	Text
2 (6)	Highlight colour	White	Shine
3 (7)	Fill colour	Blue	Fill

A command to change a colour looks like this: @{xx yyyy}; xx is either fg or bg for background or foreground (field or textcolour), yyyy is one of the above colourcodes. So, to display the phrase "We're The Problem, Gotta Love Us!" in white letters, you should type:

```
 @{fg shine}We're The Problem, Gotta Love Us!@{fg text}
```

If you leave out the @{fg text} at the end, all the text that follows will also be white, until you change it to another colour. To display the same phrase in white text on a blue background, you should enter:

```
 @{fg shine}@{bg fill}We're The Problem, Gotta Love Us!@{fg text}@{bg back}
```

If you would leave out the @{bg back} at the end, all following text would be on a blue background. It couldn't be easier... Just one more example to finish it. If you would want to display this:

```
 Gotta Love Us!
```

you would have to type in all this:

```
 @{b}@{u}@{fg shine}@{bg text}Gotta@{uu} Love Us!@{ub}@{fg text}@{bg back}
```

...mmm, maybe it could have been a bit easier... (@{i}!!!)

1.42 How to create an ASCII-Animation!

ASCII Animation

Did you know that you too can create an ASCII-Animation without too much fuzz. Before you read this, first read the pages about ASCII format and Scripting!

Remember the Australia installer? When it started you saw a Problem logo appearing in vertical columns, from the center of the screen. I will now fully explain how this was done...

1. Load up ASCII-Edit
2. Select Load from the Tools window, and in the filerequester go to the Examples/Projects directory. Then select the file called Logo.Proj
3. Press the Fix button to store the current screen

4. Go in Background mode, select colour 0 (The first one, grey) and activate the Draw Box tool. Now draw boxes over the screen to remove everything but the three center columns of characters. So, when you're done, there's only one verticle bar left, in the center of the screen.
5. Press the Save button and in the Save window, select the ASCII fileformat. Set the ASCIISpace cyclegadget to Escape-codes. Now save it as Frame1
6. Press the Undo button. You should now have the full logo back on your screen.
7. Now remove the centre column you saved in step 5, and leave everything else on screen.
8. Press Fix to store this screen.
9. Now again remove everything from screen, except for three columns to the right of the center columns you removed before, and three columns at the left of it. So, when you're done, all that's left are two columns, with an empty space between them
10. Press Save again and Save it as Frame2
11. Press Undo. Now you should have the full loge back on screen, but with some columns missing in the middle (fixed in step 8).
12. Remove the columns you just saved (to the left and right of the center) and press Fix.
13. Repeat from step 9 until you have reached the outer left and right limits of the logo. The last frame you save is an almost empty screen with only the utmost left, and utmost right columns of the logo.
14. Quit ASCII-Edit. Yes Really...

15. Load a Text Editor a la CED that can handle escapecodes.
16. Load Frame1
17. Move the cursor to the bottom of the text.
18. If you're in CED, press ESC and the [, then 0 and then H (capital, you know with Shift) so you see [[0H
19. Select Include File from the menu and load Frame2.
20. Repeat from step 17 until all frames are in and then save it as Logo.ASCII

21. Clear the screen and type this:

```
SETFONT topaz 8
TYPE Logo.ASCII
```

Save it as Logo.ANIM in the same directory as the Logo.ASCII file Make sure that an Icon is saved with the file (CED: turn on Icon Creation)

22. Leave the editor to return to Workbench. Now go to the directory where you saved Logo.Anim and look for it's icon and press it ONES to select it. From the Icon menu, select Information. Now enter this in the Tool Types area:

```
 WINDOW=CON:0/0/640/256/Logo
```

and in the Default Tool area:

```
 IconX
```

and the press save.

23. Double click the icon and if all went well, you should be looking at an animated logo, appearing from the center outwards... If it doesn't work, load the Logo.ASCII file in the editor and look for errors.

1.43 Bugs

Bugs In ASCII-Edit?

ASCII-Edit has no real bugs in it, but there are some inconveniences.

1. Redraw
Everytime you press a slidergadget, the entire screen redraws itself, even if you didn't move the slider. I could tell you that this is an extra feature of ASCII-Edit in that you have two extra Redraw Buttons, but I won't.

If you find any other bugs contact the Bug-Extermination crew!

1.44 How to load a node

Loading Guides...

ASCII-Edit can load in a node from an AmigaGuide file. If you selected an AmigaGuide from the Load filerequester, you will see a window pop up. In the window are the following things:

List-o-nodes

In the listviewer area you will see a list of all the nodes in the AmigaGuide file. If you want to read one, simply click on it's name in the list. In the background you will see a text-only version (no colours, no fontstyles etc.) This way you can search for the node you would like to load.

Textfields
Next to the list, are two textgadgets. The top one will display the selected node. The bottom one shows the real title of the node (the one between " and " in the nodes title).

Front&Back Cyclegadget

Use the cyclegadget to select what you want to load. If you select "Front&Back" ASCII-Edit will load the complete node and all on-screen information will be lost because the projects' size will be adapted to the size of the node, so remember that. If you select Front or Back then only that information will be inserted into the current project. The size of the project will not be changed, so set it up properly first.

Load New Button

Use this to bring up a filerequester which you can use to select another guide.

PREV and NEXT buttons

Press one of these to select the following or the previous item in the list.

Load Selected Button

Pressing this, will get ASCII-Edit to load in the selected node, and close the "Load Guide" window.

The Cancel Button

Pressing this, will get ASCII-Edit not to load in the selected node, and then close the "Load Guide" window.