Typeface

David Kinder

COLLABORATORS			
	TITLE : Typeface		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	David Kinder	June 25, 2022	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

Typeface

1.1 Typeface Guide

```
                  Typeface 1.0
© 1995 David Kinder
```


1.2 Introduction

```
                The Amiga provides support for two types of font; bitmap and ↩
                   outline.
In bitmap fonts each character is stored as a series of bits indicating
which pixels should be on or off. A bitmap font has a specific size, and
bitmap fonts with the same design but in different sizes are usually
stored together in the FONTS: directory, e.g. for a bitmap font "foo":

foo.font    information about foo
foo/8     bitmap font of height eight pixels
```

foo/9 bitmap font of height nine pixels

Outline fonts (introduced in Release 2.04 of the operating system) store fonts as a series of curves, so that one definition can provide any height font. Such fonts usually appear in the FONTS: directory as .font and .otag files (the .otag file contains information specific to the particular font engine to be used). Since Release 3.0 the outline font engine has been separated from diskfont.library (and is now called bullet.library). This allows new font engines to be used, e.g. type1.library, which provides support for Postscript Type 1 fonts.

Typeface is an editor for bitmap fonts. In addition, outline fonts can be loaded with specific dimensions, altered if required, and saved out as bitmap fonts. Having an outline font also saved as a bitmap font means that if an application requests the font in a size stored as a bitmap, then the bitmap version is used, otherwise the outline font is converted to the correct size, which generally takes time and memory.

Typeface may be freely distributed but remains the copyright of the

author
, David Kinder. Typeface may not be distributed for more than a nominal fee to cover media, etc. Source is provided for the benefit of anyone interested in the structure of an Amiga program, and may not be used by anyone other than the author in this or any other font editing program.

1.3 Installation

Typeface requires that
bgui.library
v38 or higher be in either the LIBS:
directory or in a "libs" subdirectory of the directory where the Typeface files are kept. Other than this, Typeface requires the presence of the following standard libraries in LIBS:

asl.library v37 or higher
diskfont.library v36 or higher

Also, amigaguide.library v34 or higher must be available if on-line help is to be used in the program.

1.4 BGUI Library

bgui.library is a shared library written by Jan van den Baard which provides a font sensitive gadget layout engine in an object orientated manner.

1.5 Using Typeface

When first started, Typeface opens a character selection window in the top-left corner of the display. In this window, any of the 256 characters which make up a font can be chosen for editing. In the window only 64 characters are displayed at any one time; use the scroll bar to move through the available characters.

Clicking on any of the characters causes a character editing window to open, in which the character is displayed on a grid. Any number of these windows may be opened, as all the windows in Typeface multitask (for example, you can open up a new character editing window with the preferences window open). All windows also have a Project menu with the following options:

Open Font

Save Font

Change Dir

Edit Font

Preferences

About

Quit

1.6 Opening a font

Project Menu/Open Font

A standard ASL font requester will open from which the user can select a font to be loaded into Typeface. Outline fonts will be listed as well as bitmap fonts; remember that while bitmap versions of outline fonts can be created, but Typeface does not support direct editing of outline fonts.

The fonts listed will be those found in memory and those in the FONTS: assign. To change the FONTS: directory, use the Change Dir option.

1.7 Saving a font

Project Menu/Save Font

This window allows control over how the font is to be saved.

"Font Directory" is the directory into which the .font file and the font
directory will be placed. The default is the current FONTS: directory.
"Font" is the name of the font. Thus if the font directory is "foo:",
the font name is "bar" and the font height is 8, then the font will be
saved as the files

foo:bar.font
and foo:bar/8

"First" and "Last" control which characters will be saved. Not all
characters in a font need to be defined; those without a definition
appear in applications as the
blank
character. "Kern Info" determines
whether or not the tables of
kerning
information are saved. This gadget
is selected and ghosted if the font is proportional, as such information
must be saved in this case. For "First", "Last" and "Kern Info" the
default is to save the font with the same parameters as the last loaded
font.

1.8 The blank character

Every bitmap font contains a definition for a "blank" character, ↩
which is
used if a character requested from the font is not defined. This
character can be edited in Typeface by clicking on the "Blank" gadget in
the
Edit Font Parameters
window.

1.9 Changing the kerning information of a character

Character Menu/Kerning

In this window the spacing and kerning values for a character can be
altered. The kerning value determines how many pixels the cursor position
should be advanced before rendering the character, and the spacing value
determines how many pixels the cursor should advance after rendering.
For example, consider this character, defined on a 5x5 grid:

```
.....
.***. Spacing = 6
**.**   Kerning = 1
.***.
.....
```

If the rendering cursor starts at position 1, and the character is
printed twice:

```
        .....  .....
 .***.  .***.   1 and 2 (also 2 and 3) are
```

```
   **.**  **.**   separated by 7 pixels (i.e.
   .***.  .***.   the spacing plus the kerning).
   .....  .....
  ^      ^      ^
  1      2      3
```

2 is the position of the cursor after rendering one character, and 3 the position after rendering both characters.

Note that these values are not usually altered in fixed-wodth fonts, and are only saved in the font if the "Kern Info" checkbox is set in the

Save Font
window.

1.10 Changing the width of a character

Character Menu/Change Width

In this window the width of the character currently being edited (the character number is shown in title bar) can be changed. Enter the new width in the string gadget then use the slider gadget below to determine how the character data is placed in the new space.

The slider gadget works as follows. If the new width is greater than the old width, then the slider represents where the old data will appear in the new space (i.e. if the slider is at the right, then the data will be copied to the new space flush with the right edge of the new space). If the new width is less than the old width, then the slider represents which section of the data will be copied into the new space, as not all the data will fit.

1.11 Changing the font directory

Project Menu/Change Dir

When
opening a font
, the FONTS: directory is scanned. If the font to be edited is not in the FONTS: assign, then this window can be used to change or add to FONTS:.

The name of the new directory should be entered in the string gadget, optionally by using a file requester (obtained by clicking on the folder gadget to the right of the string gadget). To change the FONTS: assign to this directory, click on "Assign". To add this directory to the FONTS: assign, click on "Assign Add".

1.12 Editing the font parameters

Project Menu/Edit Font

This window allows editing of the various flags and parameters that define the font. These are:

Font Type

This gadget determines whether the font is fixed-width or proportional. In a fixed-width font, all characters have the same width, whereas in a proportional font, different characters can have different widths.

Note that for Amiga fonts this distinction is not absolute, as "fixed-width" fonts can have characters with widths different from the nominal (e.g. the standard courier font). However, it is not a good idea to have fixed-width fonts with characters whose widths vary by more than a few pixels from the nominal, as this will confuse most software.

Height

The height (in pixels) of the font.

Width

The width (in pixels) of the font. For a proportional font this should be an approximate average width for all the characters in the font. For a fixed-width font, all characters should have this width (but see above). If the font is fixed-width, then changing this number will change the width of every character in the font.

Baseline

The font baseline, measured (in pixels) from the top of the font. When the font is rendered into a RastPort, the vertical cursor position (cp_y) will correspond to the font baseline. The baseline must be less than the font height.

Bold Smear

When a bold font is algorythmically generated, the font data is binary or'd with itself, shifted right by this many pixels.

The next set of parameters do not affect how the font is displayed; they only provide information on how the font should be used.

Normal The font does not have any of the following attributes set.

Bold Specifies that the font is designed so that it appears as bold.

Italic Specifies that the font is designed so that it appears as italic.

Underline Specifies that the font is designed so that

it appears underlined.

Extended Specifies that the font is designed so that it appears wider than it is high.

Reversed

Specifies that the font is designed to be rendered right-to-left (i.e. the reverse of the usual).

Aspect

Specifies the pixel aspect ratio that the font is designed for. "Normal" indicates that the pixels should be as wide as they are high; "Thin" that the pixels should be higher than they are wide; and "Wide" that the pixels should be wider than they are high.

The "Blank" gadget allows editing of the font's blank character.

1.13 Editing a character

Each character in a font can be edited in its own window, which can be opened by clicking on the character in the character selection window. The main section of the window is given over to an expanded display of the character. Clicking on any pixel in the window toggles the pixel between on and off.

The window also contains a row of gadgets along the top and a Character menu. The functions of the gadgets are:

Hollow Left Arrow Decrease the character width by one pixel.

Hollow Right Arrow Increase the character width by one pixel.

Filled Arrows Move the character data one pixel in the direction specified by the arrow.

Triangles Open the kerning window.

The Character menu items are:

Copy Copy the character into the internal clipboard.

Paste Replace the character with the last character stored in the internal clipboard.

Move Left/Right/Up/Down Move the character data one pixel in the specified direction.

Change Width Open the

width
window.

Kerning Open the
kerning
window.

1.14 Preferences

Project Menu/Preferences

The preferences window can be used to control aspects of Typeface's appearance and behaviour. The window contains a number of "pages", which are accessible by clicking on the gadgets at the top of the window. The pages available are:

Screen

Edit

Misc
Selecting "Save" will save the preferences to either ENVARC: or ↩
PROGDIR:
(the directory containing the Typeface files), depending on the selection made in the
Misc
page. "Use" makes Typeface use the current preferences but does not save them. "Cancel" ignores any changes made to the preferences and just closes the preferences window.

1.15 Preferences - Screen Page

If "Custom Screen" is selected, then Typeface uses its own screen, else it uses the Workbench screen. In the case of a custom screen, the screen type is displayed in the "Screen Type" box. Clicking on the "Choose" gadget allows the user to select which screen mode Typeface should use (the default is the Workbench screen mode). Note that the user can only select the screen mode if asl.library v38 or higher is present (Release 2.1 of the operating system).

1.16 Preferences - Edit Page

This page controls the appearance of the
character editing windows
.
"Pixel Width" and "Pixel Height" determine the size of the representation of each bit of the character's definition. "Tool Bar Height" determines the height of the tool bar (the line of gadgets along the top of the character editing windows). The minimum is 11 (which looks good for a

non-interlaced PAL screen), but if you use higher resolution displays you may want to increase this value. The "Pixel Border" checkbox determines whether or not each character pixel is edged with a border, or is displayed completely solid.

Note that changing these options does not affect character editing windows currently open, only those opened after "Save" or "Use" has been selected.

1.17 Preferences - Misc Page

In this page the location of the preferences file is determined. The possibilities are:

ENVARC: "Typeface.prefs" is written to ENV: and to ENVARC:. This is the standard place to put preferences files.

PROGDIR: This is the directory containing all the Typeface files. Saving "Typeface.prefs" here saves memory, as there will not be a copy in ENV: (usually assigned to RAM:env).

1.18 Contacting the author

If you have any comments, queries or suggestions about Typeface, I can be contacted via Internet email at either

kinder@teaching.physics.ox.ac.uk
or david.kinder@physics.ox.ac.uk

Future versions of Typeface supporting colour fonts and with more features may materialise, depending on the level of feedback that I receive. If you like Typeface, let me know.

1.19 About

Project Menu/About

Gives the version number, copyright date and author of Typeface.

1.20 Quit

Project Menu/Quit

Quits Typeface. If the font loaded into Typeface has not been saved, it will be lost.