

TrueType Font Explorer in general terms ...

This Applet was developed to meet the need for a handy method of exploring the makeup of any TrueType font. Reading through the several hundred pages of technical description and specifications for TrueType fonts can be very time consuming especially when all you want is a quick look at specific parameter for a specific font. This version (1.2) of **Explorer** does not decode all parameters ... that will come progressively in future versions as sufficient user interest warrants. Version 1.2 adds Panose Number decoding, identification of the Data Tables included in the selected font file, and a table listing the lookup offsets for data in each of the table names listed. In specific, **Explorer** can answer these questions for you:

- Can an appropriately licensed TrueType font be embedded in your application without first obtaining special permission from the owner of the font ?
- If a TrueType can be legally embedded, can it be used in a read-write configuration or must it be restricted to a read-only configuration?
- What are the actual DOS and Fontface names of the selected font file? DOS file names are sometimes numerically coded ... in a private code known only to the developer!

TrueType Font Explorer is simple to use and provides a wealth of information concerning the makeup of the particular font file selected for exploration.

The term 'appropriately licensed' as used in this help file refers to a software font file purchased by or for the end-user (or the end-user's employer) to be used only by the end-user. It is assumed that this particular software has been properly registered with the originating vendor of the software or that vendor's licensed distributor.

For example: MicroSoft distributed Arial, Courier New, Symbol, and New Times Roman with Windows 3.1 ... with the exception of Symbol, each of these fonts were provided in four styles ... Normal, Bold, Italic, and Bold Italic. The registered end-user (using Windows Version 3.1) normally would have these fonts available for his/her use on a single system.

It is not the intent of this program help file to offer legal opinion as to whether or not any specific font file can be embedded but to point out that embedding can become a violation of the vendor's proprietary rights and/or the end-user's licensing agreement. To help simplify the determination as to where you stand with a specific font, vendors set security bits within the font file to 1) prohibit embedding, 2) limit embedding to a read-only configuration, or 3) indicate that the font can be embedded in a full read-write configuration

Embedded in the font file are two security bits (found in the otmsType field, bits 1 and 2). Their assigned values are;

Bit #		Meaning
1	2	

0	0	<u>Read-write</u> and <u>read-only</u> embedding permitted
0	1	Embedding limited to <u>read-only</u>
1	x	<u>Embedding</u> not permitted

An application file written for Microsoft Windows 3.1 can store or 'embed' the TrueType fonts used by that application. Embedding ensures WYSIWYG in different platforms, not just on the platform upon which it was developed.

Read-write format for the embedded font places no restrictions on the use of that font in the application-user's system, and may offer the user the opportunity to install the font permanently thus making the font available to other applications installed on his/her system.

The main panel is color-coded and laid out in an intuitive mode of operation. During initiation, this application flushes out (clears) the Sample Text Printout window (light-blue), and the Data Display (white). Initially, all controls are light-gray as are the bit-set LED's, numbered 0 through 7.

To use **Explorer** effectively, follow this brief sequence:

- 1) select a font file to evaluate by clicking the Select Font Control. The Font Dialog Window listing will allow the user to scroll through the list of TrueType fonts installed on the user's system and to set the display size of the selected font. If a font file is not a valid TrueType font, a trill will sound and a warning message box will appear. If this happens, press 'OK' and the application will ready itself to try again with an other font. Assuming that a legitimate font file was selected, the font face name will appear in the Sample Text viewing window. The sample text is presented as a centered, display with as-displayed point size attached!.
- 2) single-click on the 'Process' control to start the data processing of the selected font file. If the selected font is classified as 'Embedding not permitted', there will be a warning trill followed by a warning message box statement. Press 'OK' after reading this warning. You will now see the font data appear in the data display window. Due to the considerable amount of decoding involved, this process can take several seconds ... even on a fast 486!
- 3) you now have several options to work with this data ...
 - 3.1 scroll the data in the data display window using the horizontal and vertical scroll bars attached to this window
 - 3.2 print the data report (2 pages) by single-clicking the 'Print' control. The printer dialog box will allow you to select and setup your printer and set the number of copies you wish.
 - 3.3 look at the bit makeup of the otmfsSelection field of OutlineTextMetric. The appropriate LED's will turn red if the associated bit is set to a '1'
 - 3.4 look at the bit makeup of the oftmsType field of OutlineTextMetric. The appropriate LED's will turn red if the associated bit is set to a '1'
 - 3.5 if you wish to clear the windows and re-initialize the LED's, single-click the 'Flush' control. This program will allow you to select and process another font file without the 'flush' but this control ensures a complete data re-read.
 - 3.6 the "Exit' and 'Help' controls do just that!

Read-only format imposes several restrictions: 1) the embedded fonts must not be allowed to be installed permanently, 2) the embedded fonts must not be enumerated in the using system, 3) the application should apply some form of encryption to the font data before embedding same in the application, and 4) all read-only font files installed by an application during its use must be deleted by the application's shutdown routine.

WYSIWYG stands for 'what you see is what you get'

