

# Technote 1146

## LaserWriter 8 and Fonts

By David Gelphman and Ingrid Kelly  
Updated by David Gelphman  
Apple Worldwide Developer Technical Support

---

### CONTENTS

[LaserWriter 8 Spooling and Fonts](#)

[Character Encodings](#)

[Font Keys](#)

[Font Downloading](#)

[Classic TrueType Support](#)

[OFA Font Support](#)

[Double-Byte Font Support](#)

[Font-Related Queries](#)

[Font Preferences Bits](#)

[Downloadables](#)

Many developers and prepress houses have contacted Apple for detailed information regarding how LaserWriter 8 works with fonts. They request this information so that they can better optimize their applications and workflows. This Technote gives a brief overview of font and printer driver interactions for LaserWriter 8, versions 8.6 and 8.6.5.

## LaserWriter 8 Spooling and Fonts

Prior to generating any PostScript code to image a given print job, LaserWriter 8 needs to know whether the fonts the document requires are available in the target output device. This affects the way the driver handles font queries and font downloading.

### Background printing

During background printing (2-pass printing), LaserWriter 8 gathers information about all the font requirements of a document prior to generating any PostScript code to render the document. The first pass gathers the document's font requirements and saves the information in the spool file. This data consists of the QuickDraw name of each font used in the document (that is, the resource name of its 'FOND' resource), along with any QuickDraw style bits (bold, italic, and so on.) applied to it. During the spooling pass, LaserWriter 8 knows nothing about which fonts are resident in the printer.

At the beginning of its second pass, after all the application drawing has occurred and been recorded into the spool file, the driver issues a font query to find out which of the needed fonts are available on the target output device. See the section [Font-Related Queries](#) for more information about the form of the driver's font queries. Since the driver knows about all font usage at the start of the second (printing) pass, it can include downloadable fonts more intelligently. See the section [Font Downloading](#) below for more information.

### Foreground printing

When a document is printed in foreground (one-pass printing), the driver cannot record all the font information in the document prior to generating the PostScript code to image the document. Therefore, the driver handles the font information as it goes along. Before the application does any drawing, LaserWriter 8 first obtains the complete list of fonts in the printer so it has this information before it generates any PostScript code. The [Font-Related Queries](#) and [Font Downloading](#) sections discuss in more detail how foreground printing is handled.

[Back to top](#)

## Character Encodings

Because Macintosh screen fonts and PostScript fonts may use different character encodings, LaserWriter 8 must sometimes redefine the encoding vector for each PostScript font used in the client's document to match the host font encoding. For each font, the driver's behavior is determined by bits 1 and 9 of the Font Classification word in the font's 'FOND' resource. The Font Classification word is described in [Inside Macintosh:Text, pp. 4-100 through 4-101](#).

**Table 1.** The Font Classification Word in a 'FOND' Resource

0	Font name needs coordinating (ignored by LaserWriter 8)
1	Macintosh vector reencoding scheme is required
2	Font has outline property by changing PaintType to 2
3	Disallows outlining simulation by smear and whiteout
4	Disallows emboldening by smear technique
5	Emboldening is simulated by increasing point size
6	Disallows obliquing for italic
7	Disallows automatic simulation of condensed style
8	Disallows automatic simulation of expanded style
9	Requires reencoding other than Macintosh vector encoding
10	Signifies the font family should have no additional intercharacter spacing other than the space character
11-15	Reserved

### Mac-encoded fonts

Fonts with bit 1 set are considered to be Mac-encoded fonts. For these fonts, the printer version of the font is "reencoded" so that the printer font used by the driver is encoded to correspond to the Macintosh Standard Roman Encoding. For Macintosh characters that are not included in Adobe's standard PostScript Roman character set, LaserWriter 8 generates PostScript code to use the printer's resident Symbol font and images them from there. Table 2 shows the mapping from Macintosh character codes to characters in the Symbol font.

**Table 2.** Characters substituted from Symbol font

Character	Name	Mac Encoding (decimal)	Symbol Encoding (decimal)
¬	notequal	173	185
	infinity	176	165
d	lessequal	178	163
e	greaterequal	179	179
	partialdiff	182	182
	summation	183	229
	product	184	213
Ö	pi	185	112
+	integral	186	242
©	Omega	189	87
	radical	195	214
E	approxequal	197	187
”	Delta	198	68
Ë	lozenge	215	224
	Euro	219	160
	apple	240	240

**Note 1:**

On most PostScript printers, the Symbol font contains a description of the Macintosh Apple character (☛) decimal 240 in the Macintosh character set). However, printers from vendors other than Apple typically do not include this character in the encoding vector for the Symbol font, making the character inaccessible when printing without reencoding. LaserWriter 8 explicitly reencodes the Symbol font to include the Apple character, ensuring that it will print correctly even on non-Apple printers.

**Note:**

PostScript printers with a PostScript language level of 1 or 2 do not include the Euro glyph in the Symbol font. In order to print the Euro glyph on these printers, LaserWriter 8 downloads a Type 1 Euro glyph and maps it into the printer's Symbol font. The Roman fonts shipping with Mac OS 8.5 have the Euro glyph included.

**Other Encoded Fonts**

A font whose Font Classification word does not have bit 1 set is not considered to be a Mac-encoded font. In this case, LaserWriter 8 does not apply the MacEncoding vector, and in addition does not switch to the Symbol font to image any characters; all characters are imaged out of the original font. If bit 1 is not set, the driver determines how to handle the encoding by looking to see whether bit 9 of the Font Classification word is set. This bit specifies whether the font is to be reencoded on the printer to a special encoding. If bit 9 is **not** set, the driver does no re-encoding of the font in the printer. If bit 9 is set, LaserWriter 8 changes the encoding of the printer font as described by the glyph-name encoding subtable. The format and location of the glyph-name encoding subtable in the 'FOND' is described in [Inside Macintosh-Text, pp. 4-105 through 4-106](#).

If bit 9 is set, the driver uses the glyph-name encoding subtable as a list of encoding positions and character names to change the PostScript font encoding. The list is used as a delta from the PostScript font's original encoding. For example, if the glyph-name encoding subtable contains two entries: (\$A8, "diamond") (\$A9, "heart") the LaserWriter driver changes the encoding of the PostScript font so that it has the original encoding, only changed so that the encoding vector at position \$A8 now contains the name "/diamond" and that at \$A9 contains the name "/heart".

**Special Case: Both bits 1 and bit 9 Set**

The LaserWriter driver special cases the situation where both bits 1 and bit 9 are set. In this case, it uses bit 9 as an indicator that a special encoding is needed and uses the glyph-name encoding subtable as described above to apply a delta encoding to the original font encoding vector. If bit 1 is also set, LaserWriter 8 treats the font as if it must switch to Symbol in order to image the set of characters listed in Table 2.

[Back to top](#)

## Font Keys

LaserWriter 8 generates PostScript code so that it draws in the QuickDraw coordinate system. This coordinate system has the origin at the top, left corner of the imageable area of the page (as defined by the print record margins) and the *Y* coordinate increases going down the page. Before using a font defined in standard PostScript format, LaserWriter 8 must invert the font matrix by applying a vertical scale factor of -1 (using the PostScript `makefont` operator) to compensate for the fact that the font's coordinate system the font is flipped in *Y* from the standard PostScript coordinate system. After applying this modification, it saves the modified version of the font in its `md` dictionary for later use under a *font key* of the form `/fm` where *m* is a serial number denoting the particular font and style.

In addition, a font may be further modified to produce derived-style variations required by the client's document, such as synthetic bold, synthetic italic, outline, shadow, condensed, or extended. Each

derived variation is considered a separate font and is given its own serial number for use in forming font keys. Thus, for example, the key /f724 might denote 12-point Helvetica synthetic bold outline.

Outline and shadow versions of a font normally outline the characters by changing the PaintType entry in the font dictionary from 0 to 2, causing the outline of each character to be stroked instead of filled (*PostScript Language Reference Manual, Second Edition*, section 5.2). However, this technique does not work with some fonts (bitmap screen fonts in particular, as well as some PostScript fonts such as older versions of Courier). If a font's 'FOND' classification word indicates that outlining cannot be performed by using PaintType 2, the driver uses its own PostScript routines to generate the outline and shadow forms by "smearing" the character image: drawing it repeatedly with a slight offset up, down, left, and right, followed by a final time, centered in white, to "punch out" the interior of the character. This smearing can be prevented by a font creator: setting bit 3 of the font classification word prevents outline smearing, and setting bit 4 prevents smearing for emboldening. See [Table 1](#) above for more information on these bit settings.

In LaserWriter 8.6.5, users can check the "Annotate Font Keys" checkbox from the "Font Settings" panel, causing the driver to insert comments into the PostScript output stream immediately before it uses a font key. This feature was added to aid the troubleshooting of font problems in the PostScript files generated by LaserWriter 8. The comment inserted follows the form:

```
%RBI FontKey: (Geneva) 12point bold
```

In this case, the font key represents application drawing in the QuickDraw font Geneva, 12 is the pointsize, and "bold" is the font style selected.

[Back to top](#)

## Font Downloading

When searching for a downloadable font on the host Macintosh, LaserWriter 8 looks for any of the following, in order of priority:

0. A PostScript outline font stored as a Type 1 'sfnt'
1. A PostScript outline font file in Adobe Type 1 or Type 3 format
2. An outline font in classic Apple TrueType format
3. A font with an 'sfnt' resource in a format recognized by OFA (Mac OS 8.5 and later only). See note below.
4. A QuickDraw bitmap font that can be converted to Adobe Type 3

### Note:

Prior to Mac OS 8.5, all fonts with 'sfnt' resources were either classic Apple TrueType fonts or Type 1 'sfnt' fonts. Mac OS 8.5 introduces Open Font Architecture (OFA) scalars and the ability to have additional 'sfnt' types besides classic Apple TrueType fonts and Type 1 'sfnt' fonts. (Prior to Mac OS 8.5, OFA was only available when QuickDraw GX was installed.) When this document refers to OFA-only fonts, it refers specifically to 'sfnt' fonts which are neither classic TrueType fonts or Type 1 'sfnt' fonts. See the section [OFA Font Support](#) for how OFA-only fonts are handled in Mac OS 8.5.

For fonts that are not already available on the target output device, LaserWriter 8 attempts to download a suitable version from the host Macintosh as part of the PostScript output stream. It first determines if the font is a Type 1 'sfnt' font. A Type 1 'sfnt' font has 'typ1' for the version of the 'sfnt' data. If the font is a Type 1 'sfnt' font, then the Type 1 data from the 'sfnt' is emitted into the PostScript output stream.

If the font is not a Type 1 'sfnt' font, the driver next looks for a file on disk containing a PostScript version of the font. The name of the file containing the font is constructed from its PostScript name, along with any style attributes applied to it, using the standard "5-2-3" rule described below:

with any style attributes applied to it, using the standard `— — —` rule described below.

1. Obtain the base font name from the style mapping table of the 'FOND' resource. Truncate this name to the first five characters (ignoring the leading four characters, "ITC-", if present). For example the base PostScript base font name "Palatino" becomes "Palat" while the base name "ITC-Palatino" also becomes "Palat".
2. Truncate each suffix to three characters needed to satisfy a QuickDraw style as specified by the style mapping table. For example the suffix "Bold" becomes "Bol" and "Light" becomes "Lig". Each triple is appended to the truncated base name.

Full example:

"Helvetica-Heavy-Italic-Condensed" becomes the file name "HelveHeaItaCon".

The driver searches for the font file in the following locations, in the order listed:

1. In the System's Font Folder
2. In the System's Extensions folder on the startup volume
3. In the System folder

If the PostScript font file is not available on disk and the font is a classic TrueType version, it is handled as described in the [Classic TrueType Support](#) section. If the font is not a classic TrueType font but is instead an OFA-only font, the driver uses OFA to download the font. The section [OFA Font Support](#) below describes the details.

If the font is none of the above types, the driver uses the QuickDraw bitmap (screen) version. Using the bitmap font, the driver constructs a PostScript Type 3 font using the PostScript `imagemask` operator to draw the characters from the font's bitmap data.

#### **Note:**

This description of the priority of the different font formats describes the behavior of the LaserWriter 8.6 driver but can be modified by the user when using the LaserWriter 8.6.5 driver. The "Font Settings" panel in LaserWriter 8.6.5 lets the user change the priority of the host font usage. If the user changes the Preferred Format radio button from "Type 1" (the default, as described above) to "TrueType", then LaserWriter 8.6.5 will give preference to an 'sfnt' version of a font (listed as priority 3 and 4 above) before using a PostScript output font file (listed as priority 2 above).

## **Saving to Disk**

When saving to a disk file rather than printing to a printer, LaserWriter 8 never includes bitmap versions of PostScript fonts, regardless of the "Font inclusion" setting in the Save dialog. Instead, it simply includes an Adobe Document Structuring Conventions (DSC) `%IncludeFont` comment indicating that the PostScript font is needed. By contrast, pure Macintosh bitmap fonts—those whose font description ('FOND') resource does not identify a PostScript equivalent—are always included when saving a PostScript file to disk. Such fonts are not bracketed with the DSC comments `%BeginFont` and `%EndFont`, however, since, for purposes of document management, the driver does not treat them as PostScript fonts.

## **Substitute Fonts**

If the user has checked the "substitute Fonts" option in the PostScript Options panel of the Page Setup dialog, the driver replaces all occurrences of the standard Macintosh screen fonts Geneva, New York, and Monaco with the PostScript equivalents Helvetica, Times, and Courier, respectively. The substitution occurs even if TrueType outline fonts are available for Geneva, New York, and Monaco. Text sizes in the document are appropriately scaled to compensate for size differences between the corresponding families of screen and printer fonts: for instance, each size of Times is scaled by a factor of 1.087 to approximate the corresponding size of New York. Unlike the LaserWriter 7 driver, LaserWriter 8 does not suppress

font substitution if the client program has enabled QuickDraw's fractional character widths.

### Standard Font Downloading

When background printing, the driver knows the entire document's font needs prior to its generation of PostScript output. Ordinarily, when background printing, all of a document's downloaded fonts are transmitted just once at the beginning of the document, as part of the job setup section preceding the first page. LaserWriter 8, when printing directly to a printer, but not to a print queue server or to disk, can delay the downloading of a given font to just before the start of the page with that font's first occurrence. Fonts are downloaded once for each document, and are downloaded so that they remain in PostScript VM for the duration of the document.

When foreground printing, fonts are downloaded as they are encountered within a document. Each font is downloaded outside of the PostScript page level save, and therefore remains in the printer's VM for the duration of the job.

### Unlimited Downloadable Fonts

The "Unlimited Downloadable Fonts" option in the PostScript Options panel of the Page Setup dialog modifies the behavior described above for foreground and background printing so that instead each downloadable font is sent each time it is switched to within the document. This minimizes the demands on printer memory by, in effect, keeping only one downloaded font resident at a time, but it can significantly affect the time needed to print the document.

#### Note:

With LaserWriter 8.6.5, users can have the driver ignore all printer fonts and always download needed fonts. If the user checks the "Always download needed fonts" check box in the "Font Settings" panel, LaserWriter 8.6.5 will ignore the results of the font query and treat the output device as if it contains no fonts. This does not apply to save to file print jobs since the "save To File" panel has its own explicit font control settings.

[Back to top](#)

## Classic TrueType Support

If no PostScript version of a required font is available, either on the target output device or on the host Macintosh, LaserWriter 8 looks for a classic TrueType font to download from the host machine. LaserWriter 8 classifies a font as a classic TrueType font if it contains a resource of type 'sfnt' with a version of either 0x00010000 or 'true', and also contains a 'glyf' table in the 'sfnt'. If the font is an 'sfnt' font but is not a classic TrueType font (or Type 1 'sfnt'), then the font is considered to be an OFA-only font and is handled as described below in the section [OFA Font Support](#).

#### Note:

With LaserWriter 8.6, if both PostScript and TrueType versions of a font are available, the driver always gives preference to the PostScript version. In LaserWriter 8.6.5, the user can modify this behavior by choosing "TrueType" as the Preferred Format in the "Font Settings" panel. In that case, the driver does not look for the PostScript version of the font on the host if the TrueType version exists.

In order to use TrueType fonts directly for drawing text, the PostScript interpreter on the output device needs a copy of Apple's TrueType rasterizing code, known as the **TrueType scaler**. The device may have its own resident copy of the scaler, or, for some printers, it may be included in the PostScript output stream as one of LaserWriter 8's downloadable procedure sets. The downloadable rasterizer only works on some level 1 PostScript interpreters.

In order to download the scaler, all of the following conditions must hold:

file:///Monster%20Boot/Apple/Week%20of%205%2F10%2F99/tn1146%20update%20DavidG/tn1146.html

In order to download the scaler, all of the following conditions must hold.

- Properly working versions of the PostScript cexec and eexec operators are defined in the device's systemdict dictionary.
- The device has enough virtual memory (at least 120K) available to run the scaler.
- The device is not an older-model (original) LaserWriter or LaserWriter Plus.

The response LaserWriter 8 receives to its print-time \*TTRasterizer feature query tells whether the device has its own copy of the scaler or, if not, whether it can accept a copy in downloaded form. LaserWriter 8 uses this information to determine how to handle TrueType fonts:

1. If the scaler is built into the output device, LaserWriter 8 simply transmits each font in Type 42 format.
2. If the scaler is not built in, but can be downloaded, LaserWriter 8 includes the TrueType scaler in its print job and then transmits each font in a Type 42 format.
3. If the scaler is not built in and cannot be downloaded, LaserWriter 8 converts each TrueType font to an unhinted Adobe Type 1 font, and transmits it in that form.

If the status of the scaler on the output device is unknown (for example, if the PostScript output is being saved to a disk file or sent to a spooler that does not understand the \*TTRasterizer feature query), LaserWriter 8 includes the scaler in the PostScript output stream and transmits each TrueType font in both a Type 42 and an unhinted Type 1 version. In this case, the driver also emits conditional PostScript code to test the availability of the scaler at print time and load it into the device's memory only if the device can accept it and doesn't already have a built-in copy. If the scaler is available at print time (whether built-in or downloaded from the PostScript output), the conditional code uses the Type 42 version of each font and discards the Type 1 version; if not, it uses the Type 1 version and skips the Type 42 version.

**Note:**

With LaserWriter 8.6.5, users can modify the driver's default behavior when downloading TrueType fonts. If the user checks the "Never Generate Type 42 Format" check box in the "Font Settings" panel, LaserWriter 8.6.5 will never convert a TrueType font into a PostScript Type 42 format font; instead, it will only generate an unhinted Type 1 format.

**Protected Fonts**

LaserWriter 8.6.5 introduces special handling of protected fonts. Some TrueType fonts contain a special 'OS/2' table indicating that they are to be specially protected. Such fonts are to be specially marked so that post-processing software can protect the font in an appropriate fashion. To properly support a font containing an 'OS/2' table, LaserWriter 8.6.5 now emits special data into a Type 42 format PostScript font generated from such a host font. It does this by using the value of the fsType field of the 'OS/2' table to insert a /FSType key into the FontInfo dictionary of the generated Type 42 font. Note that the integer value stored in the /FSType key is the fsType value from the 'OS/2' table with its zero bit masked off. Adobe Acrobat Distiller 4.0 uses this /FSType key as an indicator as to whether or not such a font can be embedded in a PDF document that Distiller generates from a PostScript file.

[Back to top](#)

## OFA Font Support

Fonts which are considered to be "OFA-only fonts" are only available and supported on Mac OS 8.5 and later. For these fonts, the driver itself does not know how to generate the PostScript version of the font; it asks OFA to do so. In this case, the driver tells OFA what font it requires and what type of output the PostScript interpreter can handle; OFA is then responsible for streaming the PostScript version of the font into the output data stream.

When the driver requests a font from OFA, it indicates what the supported output types are, and OFA is then responsible for generating one of the types that can be supported. When printing to a PostScript device where it knows that Type 42 data can be directly supported, the driver indicates that Type 42, Type 1, and Type 3 output are supported. If the driver knows that the device cannot support Type 42 data, then the driver indicates that Type 1 and Type 3 output are supported. If the driver does not know whether Type 42 can be supported, it requests a portable format. When using OFA, the LaserWriter 8.6 driver never requests the downloadable TrueType scaler format described above.

**Note:**

With LaserWriter 8.6.5, users can modify the driver's default behavior when downloading OFA fonts. If the user checks the "Never Generate Type 42 Format" check box in the "Font Settings" panel, LaserWriter 8.6.5 tells OFA that Type 1 and Type 3 are the only formats supported.

As of Mac OS 8.5, OFA generates Type 1 (or perhaps Type 3) output for the portable format requested by the driver. OFA does not generate the wrapper containing both Type 42 and Type 1 fonts described above for classic TrueType font support. This means, for example, that when saving to disk, an OFA-only font is never inserted into the stream as a Type 42 font, only as Type 1 font data.

LaserWriter 8.6.5 adds support for OFA fonts that are so-called data fork fonts. These fonts do not have the traditional 'sfnt' resource, but instead contain the 'sfnt' data in the font data fork. LaserWriter 8.6.5 supports printing and downloading these fonts when they are used with by applications drawing with QuickDraw.

[Back to top](#)

## Double-Byte Font Support

The character sets for Chinese, Japanese, and Korean fonts are typically thousands of characters and require 2 bytes of data to specify a given glyph. Such fonts are frequently called "CJK fonts" and support for such fonts is called "CJK support." Because these fonts are very large (megabytes in size), it is impractical to download these fonts to PostScript printers as part of a print job. For this reason, LaserWriter 8's support for CJK fonts in the Macintosh OS is handled differently than that for traditional 1 byte fonts. This support is done by working with WorldScript II.

On the Macintosh, text drawing is done through the `textProc` QuickDraw bottleneck. As with all bottlenecks, LaserWriter 8 installs its own `textProc` bottleneck in the port so that all text imaging is done via calls to its text bottleneck. This bottleneck is responsible for turning text drawing into PostScript code to select the appropriate font and draw the requested text in that font.

In some cases, the text coming to the driver's text bottleneck needs to be intercepted and handled specially by system software. This is made possible by the driver's use of the *print action routine*. Immediately after LaserWriter 8 inserts its bottlenecks into the printing port, it calls the Script Manager's print action routine. When it calls the print action routine, it passes in the printing port and a verb indicating the action it will take with that port. The action verbs consist of `kLWSpoolAct i on` (during spooling when two-pass printing), `kLW2PassPl aybackAct i on` (during playback when two-pass printing), and `kLW1PassPl aybackAct i on` (during one-pass printing).

The driver's call to the print action routine allows the Script Manager to replace the driver's `textProc` bottleneck with its own `textProc` bottleneck in the printing `grafPort`. This allows the Script Manager to first process text drawing calls before the driver's text drawing processes them. In the case of double-byte fonts, the Script Manager, together with WorldScript II, can either pass the text through to the driver's

text bottleneck or, when appropriate, rasterize the text drawing on the host and call CopyBits on the rendered bitmaps. This results in calls to the driver's bitsProc bottleneck to send the rendered bitmaps in the printer.

In principle, all drawing of 2-byte text could be drawn on the host as bitmaps, but this would result in slow performance and would not utilize any 2-byte fonts that were actually available in the printer. Instead, if the current font used for drawing is a 2-byte font, WorldScript II's text bottleneck routine calls the LaserWriter 8 driver to determine whether the font used for this call to the text bottleneck is available in the printer. If driver reports that the font is available in the printer, WorldScript II's text bottleneck calls through to the driver's text bottleneck and the driver's bottleneck generates the PostScript code necessary to draw the text in the printer. If the LaserWriter 8 driver reports that the font is not available in the printer, WorldScript II images the text on the host and calls CopyBits to image the rendered text into the printing port.

**Note:**

This discussion applies to all versions of LaserWriter 8, up to and including LaserWriter 8.6.5.

## Background Printing

During background printing, the driver first spools all drawing to disk. The print action verb `kLWSpoolAction` is used when calling the Script Manager's print action routine. If WorldScript II's text bottleneck determines that a two-byte font is used, it calls the driver to determine if that double-byte font is available in the printer. During the spooling pass, the driver always reports that all fonts are available in the printer. This ensures that there is never any conversion of text into bitmaps during the spooling pass.

On the second pass of background printing, the driver uses the print action verb `kLW2PassPlaybackAction`. As described above in the section [LaserWriter 8 Spooling and Fonts](#), during the second pass prior to playing back the spooled application drawing, LaserWriter 8 first queries the printer to determine the availability of the fonts that were used in the drawing. Thus, during the second pass, when WorldScript II's text bottleneck routine calls the driver to determine whether a given font is available in the printer, the driver can correctly answer based on its knowledge of the document fonts in the printer.

## Foreground Printing

During foreground printing, the driver calls the print action routine with the verb `kLW1PassPlaybackAction`. As described above in the section [LaserWriter 8 Spooling and Fonts](#), during one-pass printing, prior to any application drawing, LaserWriter 8 first queries the printer to request the list of all fonts that are available. Thus, when WorldScript II's text bottleneck routine calls the driver to determine whether a given font is available in the printer, the driver can correctly answer based on its knowledge of fonts in the printer.

[Back to top](#)

## Font-Related Queries

The LaserWriter 8 driver uses two different font queries; one is for the availability of a specific set of fonts, while the other is for a full list of fonts available in the printer. The query used depends upon whether the driver is printing in foreground or background mode.

## Background Printing

When background printing, the LaserWriter 8 driver gathers the list of PostScript fonts used in the document during the spool pass, and queries the printer about the availability of each font. The font query takes the following form:

```
%%?BeginFontQuery: <list of font names>
. . . % PostScript code to find font availability
%%?EndFontQuery: Unknown
```

The %%?BeginFontQuery comment contains a list of all fonts required by the document. Fonts are listed without the slash character (/) that typically precedes a PostScript font invocation: for example,

```
%%?BeginFontQuery: Times-Roman Times-Bold Minion Helvetica
```

For each required font, the driver uses the QuickDraw name recorded in the spool file to access the font description resource, which contains a style mapping table giving the corresponding PostScript font name, which is dependent on the QuickDraw style attributes taken from the spool file. The driver then uses this PostScript name in its print-time font query. PostScript Level 1 output devices respond to the query by reporting a font available if it is permanently resident in the device's ROM, or if it is currently present in RAM or on a disk connected to the device. On Level 2 devices, the query code determines the font's availability via the PostScript `resourcestatus` operator: the font may potentially reside in memory, on a disk, in a cartridge, on a remote server accessible over the network, or on any other type of device known to the `resourcestatus` operator. See the *PostScript Language Reference Manual* for more information on the `resourcestatus` operator.

The response to the query indicates which of the specified fonts are already available on the target device; the driver then downloads the remaining fonts as part of the ensuing print job. See below for further specifics on this query.

```
%%?BeginFontQuery: font1 font2 ... fontn
```

Requests information about the availability of PostScript fonts on the target output device. LaserWriter 8 uses this information to decide which fonts to include as part of the ensuing print job. If the PostScript output is being saved to a disk file rather than sent directly to a printer, the equivalent information is provided by the user in the "Font inclusion" item of the Print dialog's Save panel.

Under version 3.0 of the Document Structuring Conventions, the correct response to this query is of the form:

```
/fontn: Yes
. . .
/font2: No
/font1: Yes
*
```

where each font name is followed by a colon and the word "Yes" if the given font is available on the target device, "No" if it is not.

Notice that each font name is preceded by a slash character (/) in the response to the query, but not in the query itself. Notice also that the fonts appear in the response in reverse order from the way they are listed in the original query, and that the entire response is terminated with an asterisk (\*). For example, the response to the query:

```
%%?BeginFontQuery: Times-Roman Times-Bold
```

```
%%+ Mi ni on Hel veti ca
mi ght be
/Hel veti ca: Yes
/Mi ni on: No
/Ti mes- Bol d: Yes
/Ti mes- Roman: Yes
*
```

**Note:**

In describing the response to this query, some printings of the *PostScript Language Reference Manual* (Second Edition, section G.8) erroneously show a space between the colon and the word "Yes" or "No". LaserWriter 8 compensates for this error by accepting responses with or without a space following the colon.

Some print queue servers (notably the AppleShare print server prior to version 5.0) use an older form of response to the font query, as defined by version 2.0 (rather than 3.0) of the Document Structuring Conventions. For compatibility, LaserWriter 8 also accepts responses of this form. Such responses simply use the integers "1" and "0" instead of "Yes" and "No" to indicate font availability, and do not include the font names themselves or the terminating asterisk: for example,

```
1
0
1
1
```

As in the first form, the order of the fonts is reversed from the original query (Minion, Times-Bold, Times-Roman, and Helvetica in the example).

**Note:**

The AppleShare print server prior to version 5.0 imposes a further restriction on the length of a single font query: the number of characters in the list of font names in the %%?BeginFontQuery comment must not exceed 128. To accommodate this restriction, LaserWriter 8 breaks any font query that exceeds this limit into two or more separate queries.

**Foreground Printing**

When foreground printing, the LaserWriter 8 driver does not know which PostScript fonts will be used in a document, and must therefore query the printer for its complete set of fonts. This font list query takes the following form:

```
%%?BeginFontLi stQuery
. . . % PostScript code to return font list
%%?EndFontLi stQuery: Unknown
```

PostScript Level 1 output devices respond to the query by reporting all fonts available in `FontDi rectory` and on the printer's disk in the directory "fonts/". If it is permanently resident in the device's ROM, or if it is currently present in RAM or on a disk connected to the device, it is listed. On Level 2 devices, the query code uses the PostScript resourceforal1 operator with the /Font resource category. This should list all fonts available to the output device. See *Adobe's PostScript Language Reference Manual* for more information on these PostScript language operators.

The response to this query is the list of printer resident fonts in which each font name is preceded by a (slash) "/". The last font is followed by whitespace and an (asterisk) "\*". For example, a printer might respond to the font list query with the following data:

```
/Hel veti ca- Bol d
/Pal ati no- Roman
/ZapfDi ngbat s
/Ti mes- Roman
```

```

/Times-Roman
/Times-Italic
/Palatio-Italic
/Times-Bold
/Palatio-Bold
/Helvetica
/Courier
/Helvetica-Oblique
/Courier-Oblique
/Symbol
*
    
```

See [Adobe Technical Note #5001, PostScript Language Document Structuring Conventions Specification](#) for more information on the font queries.

In addition to the font queries just described, LaserWriter 8 queries the printer about the availability of a TrueType scaler. If the scaler is available, the query returns "Type42", and LaserWriter 8 downloads Type 42 fonts when appropriate. If the scaler is not in the printer, but the printer can accept a downloaded scaler, the query returns "Accept68K"; LaserWriter 8 then downloads a scaler to the printer and will download Type 42 fonts when appropriate. If the printer does not have and will not accept a TrueType scaler, the query returns "None"; LaserWriter 8 then converts TrueType fonts to unhinted Type 1 fonts and downloads these when appropriate. If the printer, or more likely a spooler, returns "Unknown" to the TrueType rasterizer query, LaserWriter 8 attempts to download the rasterizer and will send TrueType fonts down as both Type 42 and Type 1 when appropriate. The PostScript wrappers around the rasterizer and the fonts allow only the appropriate formats to be executed in the printer.

The TrueType rasterizer query takes the following form:

```

%%?BeginFeatureQuery: *TTRasterizer
// PostScript code to look for the rasterizer.
%%?EndFeatureQuery: Unknown
The further specifics of this query are:
%%?BeginFeatureQuery: *TTRasterizer
    
```

Requests information about the ability of the target output device to accept fonts defined in Apple's TrueType format. Possible responses and their meanings are:

Type42	The device has a built-in copy of the TrueType scaler.
Accept68k	The scaler is not built in but the device can accept it in downloaded form.
None	The scaler is not built in and the device cannot accept it in downloaded form.
Unknown	The characteristics of the device are unknown or the responding spooler does not understand the query.

As described earlier, LaserWriter 8.6.5 allows the user to choose the "Never generate Type 42 format" setting, which causes the driver to always treat the results of the \*TTRasterizer query as 'None', regardless of the results of any query. This setting also applies to save to file print jobs.

[Back to top](#)

## Font Preferences Bits

LaserWriter 8.6 and 8.6.5 contain a preferences resource that allows certain font-related behavior to be modified outside the normal driver user interface. Editing of this preferences resource can configure some of the driver's behavior when it handles fonts. The resource of interest is the 'PRFS' resource of the LaserWriter 8 file. In LaserWriter 8.6 and 8.6.5, there are three font-related preferences bits that can be

LaserWriter 8 file. In LaserWriter 8.0 and 8.0.5, there are three font-related preferences bits that can be modified to affect the font handling. The driver itself contains a 'TMPL' resource to allow straightforward editing of these preferences in ResEdit or Resorcerer®.

### **Use Symbol Font for Euro Currency Symbol**

This preference setting in the 'PRFS' resource controls whether the driver switches to the Symbol font to image the Euro character encoded at 0xDB in fonts marked as MacEncoding fonts by the Font Classification word in the 'FOND'. The driver preference setting is "ON" by default, and the driver switches to Symbol when imaging this character for MacEncoding fonts. If this preference setting is "OFF", the driver does not switch to Symbol but images the character in the 0xDB position of the encoding vector in the original font. For fonts using the Mac encoding vector, this character name is '/Euro'.

### **Ignore Fonts in Printer When Printing**

This driver preference setting is "OFF" by default, and the driver treats any printer fonts it finds in the query as available for usage. If this driver preference is set to "ON", the driver ignores printer fonts and downloads all fonts as if the printer contained no fonts. This preference does not apply to save to disk jobs since the user can explicitly set this in the Print dialog.

In LaserWriter 8.6.5, there is explicit UI support for this feature with the "Always download needed fonts" checkbox in the "Font Settings" panel in the Print dialog. With LaserWriter 8.6.5, the default setting of this checkbox comes from the user's saved defaults. If the user has not saved a default value for this checkbox, then its default value comes from the 'PRFS' value as described above.

### **Always Generate Type 1 Versions of TrueType Fonts**

This driver preference setting is "OFF" by default, and the driver will download Type 42 versions of classic TrueType fonts or OFA fonts if the printer can support them as described above. If this driver preference is set to "ON", the driver treats the printer as if it cannot support Type 42 fonts and can only accept Type 1 versions.

In LaserWriter 8.6.5, there is explicit UI support for this feature with the "Never generate Type 42 format" checkbox in the "Font Settings" panel in the Print dialog. With LaserWriter 8.6.5, the default setting of this checkbox comes from the user's saved defaults. If the user has not saved a default value for this checkbox, then its default value comes from the 'PRFS' value as described above.

### **PrintingLib Issues**

The "PrintingLib" file, version 8.6 and 8.6.5, contains a similar preferences resource of type 'PRF2' as well as an appropriate 'TMPL' resource to allow straightforward editing of these preferences in ResEdit or Resorcerer®. The two preference settings "Ignore Fonts in Printer When Printing" and "Always Generate Type 1 Versions of TrueType Fonts" are supported, and have meanings and defaults as described above and can be changed appropriately. Any user saved defaults made with LaserWriter 8.6.5 for these settings are respected; otherwise, the values from the 'PRF2' resource are used if there are no saved defaults. The preference settings in PrintingLib apply only to drag-and-drop printing through the Download Manager. See Technote 11xx, "The Download Manager Specification" (Coming Soon) for more information.

For the shipping configuration of Mac OS 8.5, which includes only PostScript, EPS, PICT, and JPEG downloaders, these settings affect drag-and-drop printing of PostScript, EPS and PICT files. When doing drag-and-drop printing of PostScript or EPS files, the PostScript downloader supplies fonts to satisfy the '%IncludeFont' and '%IncludeResource: font' DSC comments, and it respects these font preference settings accordingly. Font usage in PICT files is handled in the same manner as a standard print job but the PICT downloader uses the settings of the preferences in the PrintingLib file to control this special behavior.

**Note:**

The information about the ' PRFS' and ' PRF2' resources in LaserWriter 8 and PrintingLib versions 8.6 and 8.6.5 is for informational purposes only. The existence of these resources and bits control special behaviors that are subject to change with further releases of the LaserWriter 8 driver.

[Back to top](#)

## Summary

We hope that the detailed information contained in this Technote gives developers a good overview of how LaserWriter 8, versions 8.6 and 8.6.5, interacts with fonts.

## Further References

- [Adobe's PostScript Language Reference Manual](#)
- [Inside Macintosh: Text](#)
- [Technote 1140: The Euro Currency Symbol](#)
- [Technote 11xx: The Download Manager Specification \(Coming Soon\)](#)
- [Adobe Technical Note #5001, PostScript Language Document Structuring Conventions Specification](#)

[Back to top](#)

## Downloadables



[Acrobat version of this Note \(51K\).](#)

[Back to top](#)

## Change History

- Originally written in November 1998, as Technote 1146, "LaserWriter 8.6 and Fonts" by David Gelpman and Ingrid Kelly.
- In May 1999, this Technote was updated to reflect changes made for LaserWriter 8.6.5.

## Acknowledgments

Thanks to John Blanchard, Rich Blanchard, and Paul Danbold.

---

To contact us, please use the [Contact Us](#) page.  
Updated: 10-May-99

[Technotes](#) | [Contents](#)  
[Previous Technote](#) | [Next Technote](#)