

Using Outline Fonts with PSF/MVS

**An Introduction for
Users and System Programmers**



Notices

References in this publication to products or services of IBM® do not suggest or imply that IBM will make them available in all countries where IBM does business or that only products or services of IBM may be used. Noninfringing equivalents may be substituted, but the user must verify that such substitutes, unless expressly designated by IBM, work correctly. No license, expressed or implied, to patents or copyrights of IBM is granted by furnishing this document.

Trademarks and Service Marks

The following terms appear in this publication and are trademarks of the IBM Corporation:

IBM Printing Systems Company
Advanced Function Printing
AFP
IBM®
BookMaster
MVS
OS/2®
Operating System/2®
PSF/MVS
PSF
Print Services Facility
IPDS
Intelligent Printer Data Stream
MO:DCA
Mixed Object: Document Content Architecture
FOCA
Font Object Content Architecture

The following terms appear in this publication and are trademarks of other companies:

- PostScript is a registered trademark of Adobe Systems, Inc.
- Times New Roman is a registered trademark of Monotype Typography, Inc.
- Fontlab is a registered trademark of Pyrus North America

The following IBM fonts are functional equivalents of fonts developed by Monotype Typography, Inc.:

- Sonoran Sans Serif
- Sonoran Serif

Related Publications

For more information about font products and tools, refer to the following publications:

- IBM AFP Font Collection for IBM Operating Systems: Licensed Program Specifications, G544-5229
- AFP Fonts: Font Summary, G544-3810
- IBM AFP Presentation Fonts: Type Transformer User's Guide, G544-3796.

For more information about PSF/MVS refer to the PSF/MVS System Programming Guide Version 2 Release 2, S544-3672.

For information on how to reference outline fonts refer to Data Stream and Object Architecture, Mixed Object Document Content Architecture Reference SC31-6802.

For information on the structure of outline fonts refer to Data Stream and Object Architecture, Font Object Content Architecture Reference, S544-3285.

Contents

1.0 Introduction	1
2.0 What Are Outline Fonts?	3
2.1 Font System Overview for MVS	5
2.2 Advantages in Printing with Outline Fonts	7
2.3 Limitations in Using Outline Fonts	8
2.4 Inside Outline Font Technology	9
2.5 What Outline Font Styles are Available	9
3.0 What Do You Need to Print with Outline Fonts?	11
3.1 Hardware	11
3.2 Host Products	11
3.2.1 PSF/MVS	11
3.2.2 Font Products	12
3.2.3 Applications	13
4.0 How Do You Specify Outline Fonts?	15
4.1 JCL	15
4.2 PPFA	16
4.3 OGL	17
4.4 DCF	18
4.5 Custom applications and formatting	18
4.6 Sample MCF2 Structured Fields	21
4.7 Coded fonts	22
4.8 Sample Coded Font Structured Fields	23
5.0 Migrating to Outline Fonts	25
5.1 Considerations for Mapping Between Outline and Raster fonts	25
5.2 IBM-Supplied Compatibility Font and Expanded Core Outlines	25
5.3 Coding an Exit to Substitute Raster and Outline Fonts	26
6.0 Summary	27
Appendix A. Sample Font Substitution Exit	29

1.0 Introduction

IBM Printing Systems Company introduced AFP outline fonts to its printers and software in 1995. Information about these fonts and how to use them is scattered throughout many AFP publications.

This informal paper is intended to pull together enough information about the AFP products and outline fonts to help our customers understand and start using this exciting new function.

This is an introduction and "cookbook" for PSF/MVS users, system programmers, the IBM Printing Systems marketing force, and field support. It explains outline fonts: what they are, why we should use them, and how to get started with them.

Note: Although double-byte outline fonts have been recently announced, this document is intended for single-byte font users. Information on double byte outline fonts is available in PSF/MVS APAR OW14128. Additional information on double-byte implementation will be provided in the near future.

Some chapters have an overview that may be helpful for understanding and promoting the use of outline fonts and outline font printers. Other chapters require knowledge of AFP applications and some basic knowledge of AFP fonts and data streams.

2.0 What Are Outline Fonts?

An outline font is a collection of characters that can be made to print at virtually any point size and print resolution. When printed, they can look like the traditional raster fonts that have been used with PSF for many years. However, they also can be printed in custom sizes to meet your special printing needs.

Figure 1 shows how the same outline font can appear identical to a raster font you have already been using and also provide new sizes for your print job.

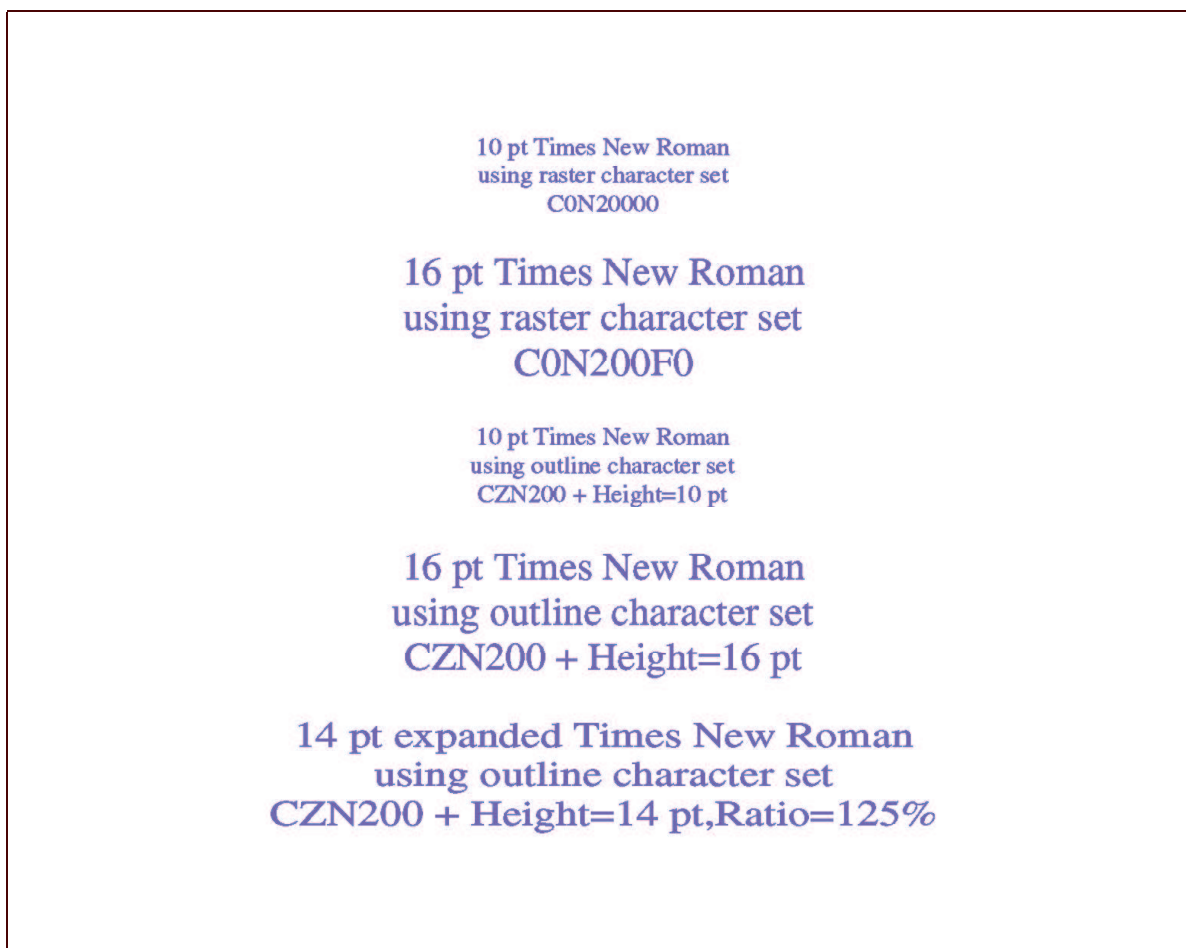


Figure 1. Examples comparing raster and outline fonts.

In this illustration two raster fonts are needed to print two point sizes of the same typeface. A single outline Times New Roman font can be used to print the same text with many different point sizes and also with expanded or condensed characters.

The internal structure of outline fonts is quite different from raster fonts. Let's compare raster and outline font characteristics.

- *Raster fonts*
 - Character shapes are defined by bitmaps at a specific font size and printer resolution. A raster font's resolution must be compatible with the printer you are using, for example, a 240 pel (dpi) raster font can only be loaded into a 240 pel (dpi) printer.
 - You must use a different font character set in the host font library or in the printer for every size and resolution needed.
 - A raster font can either be loaded from the host font library or, in the case of AFP printers with resident fonts, activated in the printer itself.
- *Outline fonts*
 - Character shapes are defined by mathematical equations. Single-byte AFP outline fonts use Adobe Type 1 outline font technology to define the characters. (See section 2.4, "Inside Outline Font Technology" on page 9)
 - A single font character set in the host library can be used for any size and printer resolution (within the limit of your printer or application).
 - An outline font can be loaded from the host font library or activated in printers that provide resident outline fonts. Several newer model AFP printers such as the 3130 and 3900 Duplex can print using outline fonts, both resident in the printer and downloaded from host font libraries via PSF/MVS. These printers create the rasterized characters from the outline fonts at the size you specify in your application and at the printer's own resolution.

Although outline fonts are relatively new to AFP, outline font technology is already widely used and accepted in other printing environments.

Raster fonts continue to be supported by PSF and all AFP outline font printers to allow customers to have the flexibility to print using existing IBM or customer raster fonts, while allowing new and modified applications to take advantage of the efficiency and flexibility of outline font technology. You can use both outline and raster fonts in many applications for AFP data streams.

Host applications such as PPFA, OGL, and DCF have been modified to support outline fonts in page definitions, overlays, and DCF formatted documents.

2.1 Font System Overview for MVS

Figure 2 on page 6 shows the flow in which fonts are created, stored, and printed along with other AFP resources. PSF/MVS merges font and AFP resources from the host libraries with user data and sends IPDS (Intelligent Printer Data Stream) commands to an AFP outline font printer to print the job.

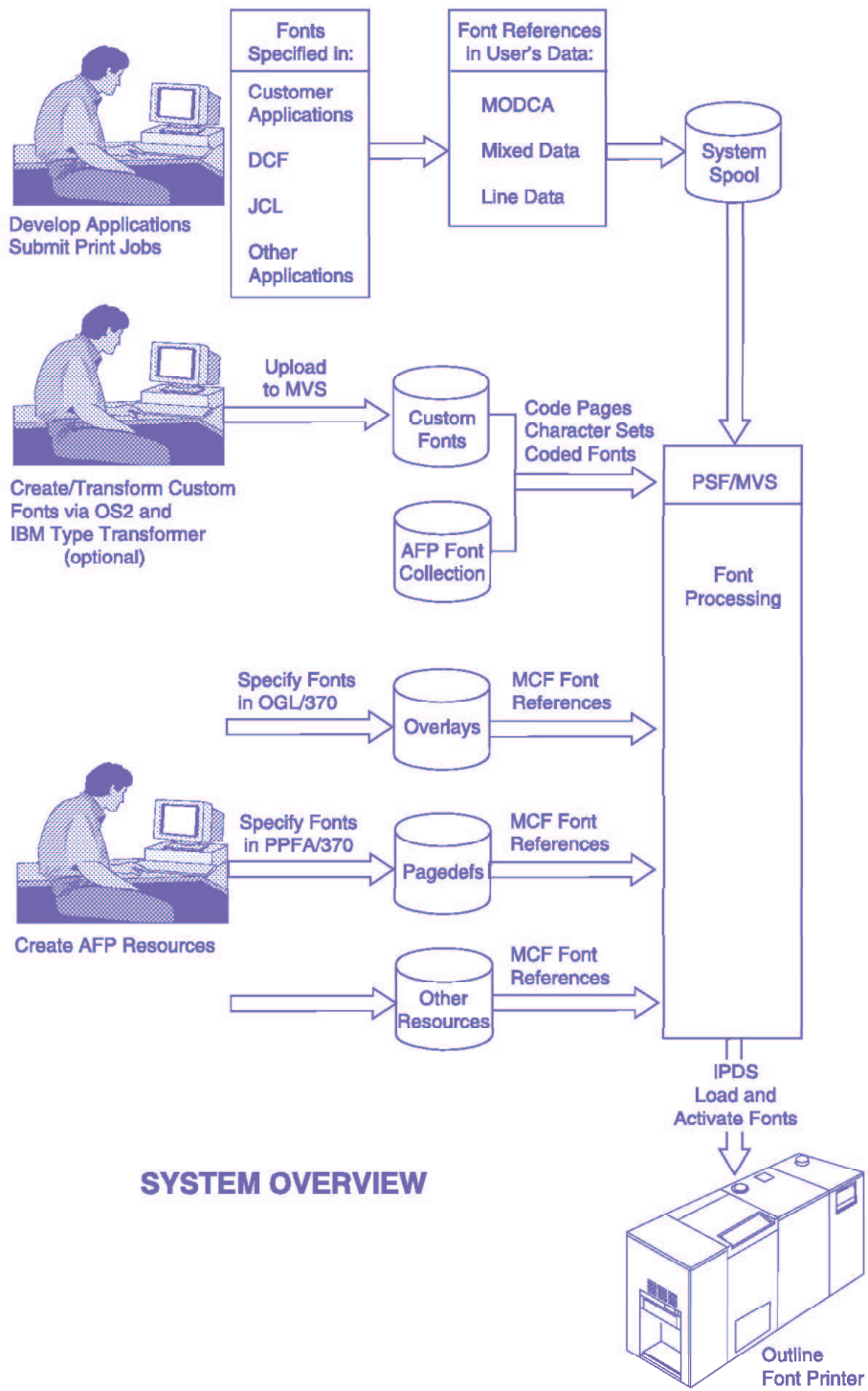


Figure 2. Outline Fonts in MVS environment.

2.2 Advantages in Printing with Outline Fonts

AFP Outline fonts offer several advantages over raster fonts that help you reduce hardware and software costs and provide easier and faster printing application development.

- **Printer resolution independence:**

IBM Printing Systems Company offers AFP outline font printers with different print resolutions such as 240 or 300 pel. Some printers allow you to select the resolution. Higher resolution AFP printers will be available in the near future. Unlike printing with raster fonts, where each resolution requires a different font library, outline font printers, regardless of resolution, can all share a common outline font library. So if you add or change your outline font printers, you do not need to convert your outline fonts to match the printer resolution.

- **More flexibility in font sizes:**

AFP raster fonts are provided in a limited number of sizes — usually 14 point sizes per typeface. AFP outline fonts allow you to specify any point size you need for your printing application. It is much easier and more efficient to simply specify custom sizes of outline fonts rather than create the individual raster fonts you need. You can also print expanded and condensed text by specifying the ratio of width to height.

- **DASD space:**

Since a single outline font can represent so many point size/resolution combinations, they are extremely space efficient. For example, a single Times New Roman outline font character set GZN200 is approximately 100Kb — similar in size to a 14 pt 240 pel raster font. This one outline font provides the same capability as the 28 separate Times New Roman raster fonts (14 point sizes, 2 resolutions), that occupy 3Mb of DASD storage. You can also print any additional sizes and expanded and condensed characters from this outline font with no additional need for disk space. The efficiency of outline fonts is especially noticeable when printing with large point sizes.

- **There are many styles available:**

The AFP Font Collection provides many font timesteps and families in outline font format. In addition, any font in Type 1 format (such as PostScript fonts) can be used to create AFP outline fonts using tools provided in the AFP Font Collection.

- **Fewer font names to learn:**

With raster fonts, a different coded font or character set name is required to identify each point size. With outline fonts a single name can be used — the size is specified separately.

2.3 Limitations in Using Outline Fonts

Outline fonts have some limitations, mainly if you are migrating from existing applications and printers that use raster fonts. Here are some limitations that you may need to consider:

- Outline fonts cannot be used on raster-only AFP printers, such as the 3820, 3825, and 3930 printers. In many situations you can get around this limitation.
 - If your print jobs use the outline fonts supplied in the AFP Font Collection, PSF/MVS can find equivalent raster fonts using its internal font mapping tables. Also PSF's user mapping tables can be set up to map to an equivalent raster font if its internal tables can't find one.
 - Another method is to substitute the outline font with a suitable existing raster font using PSF/MVS Exit 7 (APSUX07).
 - If necessary, you can create equivalent raster fonts using IBM Type Transformer (see section 3.2.3.1, "Font Utilities — Type Transformer and Others" on page 14).

- Outline fonts are not available in all font styles you may be using. Some IBM Compatibility Fonts and the font licensed products such as Sonoran Serif and Sonoran Sans Serif are not available as outline fonts.

Outline fonts equivalent or similar to many raster fonts are available either in the IBM AFP Font Collection or as Type 1 outlines from font vendors such as Adobe and Monotype. IBM Type Transformer can be used to create AFP outline fonts from Type 1 outlines (see section 3.2.3.1, "Font Utilities — Type Transformer and Others" on page 14).

- Specifying outline fonts is a little different. Host applications such as PPFA, OGL, and DCF provide easy-to-use parameters to allow you to specify outline fonts. However if you are using a custom-designed application, you may need to change the way you specify your fonts. This will be discussed later in chapter 4.0, "How Do You Specify Outline Fonts?" on page 15.
- The CHARS parameter in the PSF/MVS start-up procedure or print job JCL does not allow you to specify height, width or the font prefix that allows you to use an outline font. Some techniques to use an outline font with the CHARS parameter, are discussed later in section 5.3, "Coding an Exit to Substitute Raster and Outline Fonts" on page 26.

2.4 Inside Outline Font Technology

In the simplest sense, outline fonts are text characters expressed as mathematical curves. The boundaries of the character shapes are precisely defined by the properties of various mathematical equations and the areas defined within these boundaries are filled with pixels when the character is rasterized for printing. Many equations can be used for describing character outlines and some of the more common are: line and arc, conics, and Bezier functions.

Just as mathematical functions define shape, input values to these functions define size. This is the biggest plus that outline fonts provide. Font size can be changed simply by changing the input values. Also, by changing these values anamorphically (x-direction and y-directions are not in a 1:1 ratio) the actual shapes can be changed to produce expanded, condensed, stretched, and shortened characters.

If you visualize a hollow character where the inner and outer extents are represented by continuous lines you have the concept of an outline character. Now, imagine filling this character with pels and you have the general idea of what happens when a character is rasterized. Everything works out nicely until you get to the edges and find out that pels hang out the sides. Something has to be done to determine how best to determine which pels are turned on and which are left off. This is the concept of "hinting" where rules are established for solving such dilemmas. "Hints" tell the printer's rasterizer which shape characteristics are to be enforced when the pels are placed. Some of these characteristics are: stroke weight (horizontal and vertical), bowl shapes, serifs, and x-height. These hints can be extremely complex and have given rise to the term "intelligent outlines". Adobe Type 1 fonts used throughout IBM are "hinted".

2.5 What Outline Font Styles are Available

The IBM AFP Font Collection (Program Number 5648-113) contains AFP outline fonts that can be installed at the host for use with PSF/MVS and AFP printers. In this product, many fonts, both typographic and uniformly spaced, are available in outline font format. The font families included are:

Courier APL2

Courier Latin1, Latin235, Latin4, Cyrillic Greek, Arabic, Hebrew, and Symbols

Helvetica Latin1, Latin235, Latin4, Cyrillic Greek, Arabic, Hebrew, and Symbols

Times New Roman Latin1, Latin235, Latin4, Cyrillic Greek, Arabic, Hebrew, and Symbols

Gothic Katakana

Boldface Latin1

BookMaster Latin1 and BookMaster Latin1 Reverse

BookMaster Latin1 Specials and BookMaster Latin1 Specials Reverse

Gothic Text Latin1

Letter Gothic Latin1

Prestige Latin1

OCR-A and OCR-B

In addition, there are thousands of outline fonts available from vendors such as Adobe and Monotype. You can use IBM Type Transformer to produce AFP outline fonts from any font available in the Adobe Type 1 format. You can even customize Type 1 outlines using outline font editors such as Fontlab.

3.0 What Do You Need to Print with Outline Fonts?

To print with outline fonts you will need printers that use outline fonts and the appropriate host products.

3.1 Hardware

The following IBM AFP printers use outline fonts. All of these printers have printer resident outline fonts that correspond to equivalent host fonts.

- 3130 Advanced Function Printer
- 3160 Advanced Function Printer
- 3900 Advanced Function Printer (all models except Model 1)
- 3935 Advanced Function Printer

Additional outline font printers are currently under development. Most new non-impact printers developed by the IBM Printing Systems Company will have outline font support.

If you are using intermediate devices with your printers, you should determine if they support outline fonts. For example, outline font printers attached with PSF Direct can use outline fonts. Printers attached with RPM or DPF cannot use outline fonts.

3.2 Host Products

3.2.1 PSF/MVS

PSF/MVS 2.2 with APAR OW08340 provides new function to manage and load outline fonts. This includes:

- Using extensions in the MO:DCA Map Coded Font structured field (MCF2) for size and scaling of fonts.
- Using FOCA structured fields in AFP coded fonts and character sets to determine if the fonts are outline or raster technology.
- Using code pages (that already exist today in host font libraries) interchangeably with raster fonts and outline fonts.
- Issuing the appropriate IPDS commands to download or activate character sets and code pages for outline fonts and to activate the complete font at the specified size and ratio of width-to-height.

- Additions to PSF internal mapping tables to include IBM AFP Font Collection outline fonts and enable mapping of raster-to-outline fonts and outline-to-raster fonts for migration purposes.
- Addition of new user-modifiable mapping tables to add mappings for outline and raster fonts that are not shipped with the AFP Font Collection. If a customer has both outline and equivalent raster fonts, then the mapping tables can be used to print with custom fonts using either technology.
- Addition of new fields in PSF/MVS Exit 7 (APSUX07) to allow substitution and reload of outline fonts. Also PSF/MVS passes information to Exit 7 that tells the exit what font technology the printer supports. The exit can be coded to force substitution to use an outline font if available.
- Using AFP Font Collection fonts to activate printer resident fonts without running the APSRMARK utility.

If an application uses the new capabilities to specify an outline font (see section 3.2.3, “Applications” on page 13), then PSF MVS 2.2 with APAR OW08340 will use an outline font in the host library or the equivalent printer-resident outline font it activates. This assumes that the printer supports outline fonts and the outline fonts are available in PSF host libraries.

If an application uses the old methods of specifying fonts, it is specifying a raster font. If a raster font is specified, PSF will select an equivalent *printer-resident* outline font if the raster font in the host library contains the information to identify and activate a printer resident font. PSF will use a *host-library* outline font instead of the raster font only if Exit 7 specifies that it do this. See section 5.3, “Coding an Exit to Substitute Raster and Outline Fonts” on page 26.

PSF/MVS 2.1 can also activate printer-resident outline fonts in the same way it can activate printer-resident raster fonts in printers such as the 4028 and 3916. If the host-library raster font selected by the print job is marked public and identifies an outline font resident on the printer, the resident font will be used. The information in the host font can be created by the APSRMARK utility.

3.2.2 Font Products

To use a printer-resident outline font, you must have a font in your PSF host library that contains information for identifying and activating the printer-resident font. To use a downloaded outline font, you must have a font in the AFP FOCA outline format. The IBM AFP Font Collection (program number 5648-113) provides AFP FOCA outline fonts and 240 and 300 pel raster fonts that contain information for activating resident outline fonts. The outline fonts can be used by

PSF/MVS 2.2 with APAR OW08340. The raster fonts can be used with either PSF/MVS 2.1 or PSF/MVS 2.2.

Font Collection outline fonts are available in CDROM (for use with Optical Media Attach/2 hardware), 3480 cartridge, or 4mm tape (available 3/96). The Font Collection is installed using SMP/E. The AFP Font Collection also provides tools for creating AFP FOCA outline fonts from Adobe Type 1 format outlines, and for managing and modifying outline fonts. See section 3.2.3.1, "Font Utilities — Type Transformer and Others" on page 14.

3.2.3 Applications

Applications developed using raster fonts can be printed with outline fonts if the raster fonts contain information to activate printer resident fonts, or if PSF tables and Exit 7 are set up to substitute outline fonts for raster fonts. However to take advantage of the new capabilities of outline fonts, you must change the method used to specify fonts for your application. The key host products used in the MVS environment have been updated to provide parameters to reference and scale outline fonts.

- PPFA/370

APAR PN70854 adds HEIGHT and RATIO subcommands to the FONT Command. The FONT command has been expanded to allow referencing of code page/character set pairs explicitly. Also you now can reference the GRID (the font's hexadecimal Global Resource IDentifier - a combination of the code page and character set identifiers, plus the font width).

Also APAR PN79369 is recommended. This APAR improves PPFA to allow you to use the same coded font or code page/character set pair names in more than one FONT command with different HEIGHT and RATIO subcommands.

- OGL/370

APAR number PN71601 adds HEIGHT and SCALE subcommands to the FONT Command when specifying outline fonts with a coded font name.

- DCF

APAR PN69168 adds support for outline fonts.

- SCRIPT/VS has been modified to use outline font technology.
- The Font Library Index Program (FLIP) recognizes outline fonts and includes them in the DCFINDEX and output listing file.
- Outline fonts can be used with the Define Font command.

If you are coding your own application that generates MO:DCA or line data, then you should use the MO:DCA MCF Format 2 structured field, referencing either code page/character set pairs or outline coded fonts, and include the size information in the Font Descriptor Specification Triplet. You can also use the MCF Format 1 or MCF Format 2 with outline coded fonts. Outline coded fonts provided with the AFP Font Collection already contain the size information and reference the code page and outline character set. If you want to change the ratio of width to height you need to add a Font Horizontal Scale Factor Triplet to your MCF2 reference.

MCF font referencing is discussed in section 4.5, “Custom applications and formatting” on page 18.

3.2.3.1 Font Utilities — Type Transformer and Others

Type Transformer is an OS/2 program that is available as a feature of the AFP Font Collection. It provides functions for creating and managing fonts. It can be installed and used on a workstation with OS/2 and Communications Manager, a CD-ROM drive, a hard disk, and at least 2MB available random access memory (RAM).

You can use Type Transformer and its associated tools to do the following functions:

- Create AFP outline fonts from Adobe Type 1 outline fonts. AFP outline fonts are used by the host applications and PSF.
- Create AFP raster fonts in 240 or 300 pel at integer point sizes from Adobe Type 1 outline fonts.
- Edit and create custom coded fonts using the code pages, character sets, and the size information you specify.
- Edit code pages to change mappings of code points to characters.
- Upload fonts into host font libraries.

4.0 How Do You Specify Outline Fonts?

Outline font printers can use both raster and outline fonts. Knowing how you intend to use your fonts, both outline and raster, will help simplify specifying the fonts in your applications.

If you use outline font printers, and the raster fonts you need are at the same resolution as the printer and you do not need custom sizes, you can still specify raster fonts in your applications.

If you intend to use the fonts shipped with the AFP Font Collection in the standard sizes and resolutions provided, you probably do not need to be concerned whether outline fonts or raster fonts are used in your applications. If you want to provide new font sizes in your printing applications or use the same font libraries with outline font printers in different resolutions, host utilities such as PPFa and OGL can be used to exploit the benefits of outline fonts. These programs provide simple-to-use parameters to specify the font with height and font width-to-height ratio. Examples are described later in this chapter.

The outline fonts provided in the AFP Font Collection use a naming convention with an "XZ" prefix for coded fonts and a "CZ" prefix for the character sets. The same code pages (prefixed with "T1") can be used with either outline or raster fonts.

If you use height parameters to specify the point sizes or font width-to-height ratio parameters then you need to be sure that the required outline fonts are in your font libraries and they are concatenated in the PSF start-up procedure.

Included in this chapter are some of the most popular ways to specify fonts in the MVS environment when printing with PSF/MVS.

4.1 JCL

When you specify a font using the CHARS parameter in the print job JCL (including referencing CHARS fonts in the TRCs in line data) or in the PSF/MVS startup procedure, the coded font name is specified without the "Xn" prefix. Also there are no JCL parameters to identify the size of the font. The CHARS parameter does not give you the capability to specify an outline font. If you use the CHARS parameter, you are specifying a raster font. You can instruct PSF to use an outline font instead of the raster font specified, as described in section 5.3, "Coding an Exit to Substitute Raster and Outline Fonts" on page 26.

4.2 PPFA

To use outline fonts in page definitions, two new subcommands have been introduced to the FONT command:

- **HEIGHT**
This subcommand specifies the font height in points and must be an integer. A point is 1/72 of an inch. For example, for 1 inch tall characters, specify HEIGHT 72. The maximum HEIGHT that can be specified is 999.
- **RATIO**
This subcommand specifies the scaling ratio to be applied in the horizontal direction, expressed as an integer "percent". If not specified, normal scaling occurs. If RATIO 50 is specified, the horizontal direction is scaled half as much as the vertical, resulting in condensed (narrow) characters. If RATIO 200 is specified, the horizontal direction is scaled twice as much as the vertical, resulting in expanded (wide) characters. RATIO 100 is "normal" scaling. This means that the printed characters will have their "normal" aspect ratio, and does not mean that the output characters will have the same width as height. They will be scaled by the same amount for both width and height. For RATIO to be specified, HEIGHT must be present.

To specify outline fonts in the FONT command, the following may be used.

- The best method for specifying an outline font is to use the character set and code page name explicitly in the FONT command and include HEIGHT and (optionally) RATIO. For example, to specify a Times New Roman font at 10 points with the width 1.5 times the height, you could use the following:

```
FONT MYFONT CS N200 CP V10500 HEIGHT 10 RATIO 150;
```

The character set name is not prefixed with a "C0" or "CZ" and the code page name is not prefixed with "T1". PPFA will append the prefixes to the font when it generates the page definition. Since HEIGHT is specified, PPFA will append a "CZ" prefix to the name N200 to make the complete character set name CZN200.

- You can specify an outline font using a coded font name. You still need to use the HEIGHT subcommand to generate an outline font name. For example:

```
FONT MYFONT N2100C HEIGHT 16 RATIO 150;
```


The coded font name XZN2100C generated by PPFA already has a size of 10 points, but the height and ratio in the command above will override the size in the coded font.

You can use the same font name with different sizes and ratios and the fonts will print in the sizes specified in the FONT command. For example:

```
FONT MYFONT1 N2100C HEIGHT 10;  
FONT MYFONT2 N2100C HEIGHT 16;  
FONT MYFONT3 N2100C HEIGHT 14 RATIO 125;
```

MYFONT1 will print Times New Roman at 10 points, MYFONT2 will print Times New Roman at 16 points, and MYFONT3 will print Times New Roman at 14 points with a width-to-height ratio of 125 percent.

4.3 OGL

For creating overlays, two parameters used when specifying outline fonts are HEIGHT and SCALE.

HEIGHT represents the vertical point size and is specified in the FONT command as a whole or real number from 1 to 999.9.

The SCALE parameter represents a ratio of the desired font width to the normal font width and is used for printing expanded or condensed characters. It is similar to the RATIO parameter in PPFA.

With AFP outline fonts, you can use the HEIGHT parameter of the FONT command to define the font's vertical point size and the SCALE parameter to define the the font's ratio of character width to height. For example, AFP outline coded font XZN2100C pairs Times New Roman Latin1 character set CZN200 and code page T1V10500. The vertical font size in XZN2100C is 10 points. An OGL user could define fonts as follows:

```
FONT FONT1 n2100c ddname FONTOLN;  
FONT FONT2 n2100c ddname FONTOLN HEIGHT 16;  
FONT FONT3 n2100c ddname FONTOLN HEIGHT 16 SCALE 125;
```

As shown in this example FONT1 would be 10 points high, the vertical font size from the coded font. FONT2 would be 16 points high as specified in the FONT command. FONT3 would be 16 point but the widths of the characters would be expanded in the horizontal direction to 125 percent of the 16 point character widths.

SETUNITS values do not affect outline font scaling.

4.4 DCF

To use outline fonts in DCF, you should first run the Font Library Index Program (FLIP) with the installed outline font library. Then you can use the outline fonts either explicitly or implicitly.

- You can use the .DF FONT command to name the outline font explicitly. It is the same as using the command with raster fonts, only the font is prefixed with an "XZ". For example:

```
.DF MYFONT FONT XZN210BC
```

- You can use the .DF TYPE command to name the outline font implicitly. For example:

```
.DF MYFONT type ('Times New Roman Latin1' 12)
```

When invoking SCRIPT/VS to format the document, use formatting parameters in your JCL to indicate you are using outline fonts. For example:

```
DEV(AFPA) FONTLIB('SYS1.FONTOLN') CHARS(XZN210BC)
```

Note: DCF does not support scaling the font width-to-height ratio or using fractional point sizes for outline fonts.

4.5 Custom applications and formatting

If you are coding your own applications that generate AFP data streams, you can specify both outline and raster fonts using the MO:DCA Map Coded Font (MCF2) structured field. Use your MO:DCA reference (SC31-6802) to build the structured fields for MCF2 (structured Field ID X'D3AB8A'). You will need a Resource Local ID triplet (X'24'), a Character Rotation triplet (X'26'), plus the Fully Qualified Name triplet (X'02') with either a code page/character set pair or a coded font name. For size information you will also need to add a Font Descriptor triplet (X'1F'). If you wish to change the font width-to-height ratio, you will need the Font Horizontal Scale Factor triplet (X'5D'). Some examples of coded MO:DCA MCF2 structured fields are in section 4.6, "Sample MCF2 Structured Fields" on page 21. The following tables describe how to code these fields:

- Fully Qualified Name Triplet with Code page/character set pair
Insert in your MCF structured field a X'02' Fully Qualified Name triplet, one for each code page and character set, as follows:

# Bytes	Description
1	Triplet Length
1	Triplet ID = X'02'
1	Name type = X'85' (Code Page)
1	GID format = X'00'
4-8	Code Page Name such as T1V10500
1	Triplet Length
1	Triplet ID = X'02'
1	Name type = X'86' (Character Set)
1	GID format = X'00'
4-8	Outline Character Set Name such as CZN200

- Fully Qualified Name Triplet with Coded Font
Use this triplet if you want to use a coded font name instead of a code page/character pair.
Insert in your MCF structured field a X'02' Fully Qualified Name triplet as follows:

# Bytes	Description
1	Triplet Length
1	Triplet ID = X'02'
1	Name type = X'8E' (Coded Font Name)
1	GID format = X'00'
4-8	Coded Font Name such as XZN2100C

- Font Descriptor Triplet
The font descriptor triplet is required when using a code page and outline font character set pair. It is optional when you use an outline coded font that already contains size information. If size is included with an outline coded font, the information in this triplet will override the size information in the coded font. If both Font Height and Font Width are coded, Font Height will be used by PSF.

# Bytes	Description
1	Triplet Length = X'09'- X'14'

# Bytes	Description
1	Triplet ID = X'1F'
1	Character stroke thickness (Reserved) X'00'
1	Character width-to-height ratio (Reserved) X'00'
2	Font Height in 1440ths of an inch (point size X 20) (recommended size parameter for outline fonts)
2	Font Width in 1440ths of an inch (alternate size parameter for outline fonts)
1	Qualifier Flags (Reserved) X'00'
0 - 10	(Reserved) X'00'

- Font Horizontal Scale Factor Triplet

This triplet scales the font width-to-height ratio and should be used only if you want to expand or compress the characters. If the Horizontal Scale Factor value is the same as the Font Height in the Font Descriptor triplet or in the coded font then the font is uniformly scaled to its normal size. If it is greater than the Font Height, then the widths of the printed characters are expanded or wider than normal. If it is less than the Font Height, then the widths of the printed characters are compressed or narrower than normal.

# Bytes	Description
1	Triplet Length = X'04'
1	Triplet ID = X'5D'
2	Horizontal Scale Factor in 1440ths of an inch

4.6 Sample MCF2 Structured Fields

```

Header          Repeating Groups:
-----
!.FLz«...      .... .XZN2100C....
50ADA8000      010080EEDFFFC0200
A083BA005  1)  02C2E0795210034451
-----

                -----
                .... .XZN2100C.....0.....
                020080EEDFFFC11000F000000000000000200
2)  06C2E0795210034F00000000000000000004453
                -----

                -----
                ....e.T1V10500..f.CZN200 .....0.....
                030080EFEFFFFF0080CEDFFF4411000F000000000000000200
3)  02C25031510500C260395200004F0000000000000000004453
                -----

                -----
                ....e.T1V10500..f.CZN200 .....).°
                030080EFEFFFFF0080CEDFFF44110004000000000000002000509
4)  06C25031510500C260395200004F001000000000000000044554D10
                -----

```

Figure 3. ISPF Browse (Hex On) of MCF-2 Structured Field

The figure above shows an MCF-2 Structured field containing outline font definitions. The format is from an ISPF Browse screen with hexadecimal displayed ("HEX ON"). For readability, in this figure the MCF-2 structured field record is displayed in sections: the header followed by 4 repeating fields, one for each font defined:

1. Repeating group 1 defines coded font XZN2100C. This is the outline coded font for Times New Roman medium 10-point, supplied with the AFP Font Collection. No size parameters are specified in this repeating group, so the size information in the coded font (which is 10 points) will be used.
2. Repeating group 2 also defines coded font XZN2100C but includes triplet X'1F' to specify a font height of 12 points. The value specified, X'F0', is 240 1440ths-inch, or 12 72nds (12 points). This height definition will override the size information in the outline coded font.
3. Repeating group 3 uses character set and code page names. Triplet X'1F' must be used to specify size. Again height is specified as X'F0' or 12 points.

4. Repeating group 4 also uses character set and code page names and triplet X'1F' to specify height. Here height is X'140' which is 320 1440ths-inch or 16 72nds (16 points). In addition, this definition includes triplet X'5D' to specify horizontal scale factor. Here the factor is X'190' or decimal 400, which is 125% of the height value 320. The characters will be expanded 25% in the horizontal direction.

4.7 Coded fonts

If you create outline coded fonts, it is recommended that the size information should be included, even if you use the MCF structured field to override the size.

If you want to create a complete coded font using an outline font character set, code page, and size information, use IBM Type Transformer to create the FOCA coded font. If you don't have Type Transformer, you can edit the Coded Font Index structured field (structured field ID = X'D38C8A') repeating group. An example of the structured fields in a coded font is in section 4.8, "Sample Coded Font Structured Fields" on page 23. The following table describes how to code the Coded Font Index structured field data for outline fonts:

# Bytes	Description
8	Outline character set name such as CZN200 (Pad with blanks after name if necessary)
8	Code page name such as T1V10500 (Pad with blanks after name if necessary)
2	Font Height in 1440ths of an inch (point size X 20)
2	Horizontal Scale Factor in 1440ths of an inch
4	Not used X'0000 0000'
2	Section number = X'00'

4.8 Sample Coded Font Structured Fields

```
Begin Coded Font :
-----
!..Ly«...
500DA8000
A0838A001
-----

Coded Font Control:
-----
!..Lx«... ..
500DA8000 10
A0A37A002 91
-----

Coded Font Index:
-----
!..L%«... CZN200 T1V10500.H.....
502D88000 CEDFFF44EFEFFFFFF0C0000000
A013CA003 3952000031510500080000000
-----

End Coded Font :
-----
!..Lz«...
500DA8000
A0839A004
-----
```

Figure 4. ISPF Browse (Hex On) of Outline Coded Font XZN2100C

The figure above shows the FOCA coded font structured fields as seen on an ISPF Browse screen with hexadecimal displayed ("HEX ON"). For the Coded Font Control and the Coded Font Index the structured field headers and data are separated for readability.

The structured field which has the outline font size parameters is the Coded Font Index. The Coded Font Index in this example defines an outline Times New Roman medium character set, a code page, and a font height of X'00C8' or 200 1440ths-inch (equivalent to 10 points).

5.0 Migrating to Outline Fonts

5.1 Considerations for Mapping Between Outline and Raster fonts

For printing custom font sizes on outline font printers, it is recommended that outline fonts be used for document preparation. However, if you also need to print these documents on raster-font-only printers, you can do so by creating custom raster fonts for the required sizes and resolutions and installing them in the appropriate host libraries used with your printers.

You may need to use the user-modifiable font mapping tables in PSF/MVS 2.2. Sample tables are provided with APAR OW08340. Using these tables will allow you to reference an outline font in your document and print with the equivalent custom raster font when using a raster-font-only printer. Chapter 13 in the PSF/MVS System Programming Guide (S544-3672) explains in detail how to set up and enable the mapping of outline-to-raster technology.

You can use PSF/MVS Exit 7 (APSUX07) to perform substitution of a suitable raster font for your outline font. An example of how to code an exit to do outline-to-raster substitution is described in section 5.3, "Coding an Exit to Substitute Raster and Outline Fonts" on page 26.

IBM Type Transformer does not use the width-to-height ratio to create custom raster fonts. Also the PSF/MVS mapping tables do not use this ratio. Therefore expanded or condensed fonts can only be used on outline font printers.

Type Transformer creates raster fonts in integer point sizes only. If you are using applications or utilities such as OGL to specify height using 10ths of a point (for example, 8.7 pt), you may not be able to use an equivalent raster font. If you need to print jobs with outline fonts with a raster font printer, you may be able to use Exit 7 to substitute a raster font that is close to the size specified with the outline font.

5.2 IBM-Supplied Compatibility Font and Expanded Core Outlines

IBM-supplied Compatibility fonts are available as 240 and 300 pel raster fonts but are not available as AFP outline fonts. However, the

AFP Font Collection Expanded Core fonts do provide many of the most popular Compatibility font typefaces in outline form. Included are:

- Courier
- Boldface
- Gothic Text
- Letter Gothic
- Prestige
- APL
- OCR-A
- OCR-B

Refer to "AFP Fonts: Font Summary" (G544-3810) for information about these typefaces.

5.3 Coding an Exit to Substitute Raster and Outline Fonts

Exit 7 (APSUX07) provides some methods to assist migrating your applications to outline fonts.

If your print job specifies raster fonts and you wish to use host-library outline fonts on your outline printer, you must use Exit 7 to tell PSF/MVS to locate an outline font instead of the raster font. Enabling XTP7MTOF tells PSF to search the libraries for outline fonts instead of the raster fonts specified in your JCL, resources, or document. When XTP7MTOF=ON you should also set XTP7RFNF=ON to ensure that if the outline font is not found it will use the raster font name. The additional library searches may impact performance. Chapter 13 in the PSF/MVS System Programming Guide (S544-3672) has explicit instructions describing how to set up and use the mapping of raster and outline fonts.

For some applications, you may wish to use Exit 7 to substitute a specific outline font for a specific raster font. For example, you can substitute XZ6980 for Gothic Katakana X0KN12. You can even change the size or width-to-height ratio.

If you need to print a job that specifies outline fonts on a raster-font-only printer, you can use Exit 7 to substitute suitable raster fonts for the outline fonts specified. You will not need to do this for all outline fonts. If the fonts, at the sizes used, are in the PSF mapping tables, PSF will use the equivalent raster font. The PSF-supplied tables map standard sizes of all the AFP Font Collection outline fonts to equivalent raster fonts. You may also customize PSF mapping tables to add other fonts. Appendix A, "Sample Font Substitution Exit" on page 29 illustrates possible substitution methods based on the printer's font technology — either raster-font-only or outline.

6.0 Summary

To take maximum advantage of outline font technology, when installing outline font printers, consider the following recommendations:

- If you are developing new applications, create your AFP resources and applications using outline fonts. You will have the freedom to use outline font printers with different resolutions using the same font libraries, and with virtually unlimited sizes. This will ease future migration to printers with different resolutions.

For existing applications, consider changing your raster fonts to equivalent or similar outline fonts in your applications and AFP resources. If you intend to use specific raster fonts and your outline font printer is at the same resolution as your raster fonts, you can use your raster fonts also.

- If you must also use raster-font-only printers along with your outline font printers, you should determine what raster fonts need to be provided in the resolution of your printers.

One helpful tip: standard sizes of the Expanded Core Fonts in the AFP Font Collection are available as 240 and 300 pel raster fonts and as outline fonts. The AFP Font Summary (G544-3810) has a list of these fonts. PSF/MVS has internal mapping tables to find equivalent Expanded Core raster fonts if standard sizes are used and a raster-font-only printer is attached. To minimize creating custom raster fonts for your raster-font-only printers, use only standard point sizes in applications that must also print on raster-font-only printers.

Appendix A. Sample Font Substitution Exit

The following sample exit illustrates how you can substitute raster and outline fonts based on the printer's font technology. For more information on the fields used in this exit, refer to the PSF/MVS System Programming Guide Version 2, Release 2, S544-3672.

```
**** START OF SPECIFICATIONS ****
*
*01* MODULE NAME = APSUX07
* $MOD (APSUX07 ) COMP(APS) PROD(PSF) : RELEASE 2.2.0
*
*01* DESCRIPTIVE NAME = RESOURCE EXIT - FOR CODED FONT SUBSTITUTION
*
*01* STATUS = VERSION 2, RELEASE 2, LEVEL 0
*
*01* FUNCTION = THIS EXIT SUBSTITUTES AN OUTLINE CODED FONT FOR A
*                RASTER FONT AND CHANGES THE SIZE OF THE FONT
*                IF THE PRINTER IS AN OUTLINE FONT PRINTER.
*                IT ALSO SUBSTITUTES A RASTER CODED FONT FOR AN
*                OUTLINE FONT IF THE PRINTER IS A RASTER FONT.
*                NOTE: WTO STATEMENTS HAVE BEEN ADDED TO ASSIST IN
*                DEBUGGING.
*
*01* NOTES = SEE BELOW:
*         DEPENDENCIES = NONE
*         RESTRICTIONS = NONE
*
*01* MODULE TYPE = PROCEDURE
*         PROCESSOR = ASSEMBLER
*         ATTRIBUTES = REENTRANT, SUPERVISOR STATE, CALLER'S KEY,
*                   PAGEABLE
*
*01* ENTRY POINT = APSUX07
*
*01* REGISTER ON ENTRY :
*         R1 = ADDRESS OF A 4 BYTE FIELD WHICH CONTAINS THE
*             ADDRESS OF APSGEXTP
*         R13 = SAVE AREA ADDRESS
*         R14 = RETURN ADDRESS
*         R15 = ENTRY POINT ADDRESS
*
*01* INPUT = PARAMETER AREA (APSGEXTP)
*
*01* OUTPUT = N/A
*         EXIT-NORMAL = RETURN TO CALLER
*         EXIT-ERROR = NONE
*
*01* EXTERNAL REFERENCES = SEE BELOW:
*         ROUTINES =
*         DATA AREAS = NONE
*         CONTROL BLOCKS = (SEE MACROS LISTED BELOW)
*
*01* TABLES = NONE
*
*01* MACROS = MACROS USED FROM MACRO LIBRARY:
*         APSGEXTP - PSF INSTALLATION EXIT PARAMETER AREA
*         APSUECA - PSF EXIT COMMUNICATIONS AREA
*         APSURLST - PSF EXIT RESOURCE LIST AREA
```

```

**** END OF SPECIFICATIONS ****
APSUX07 START 0
        TITLE 'DSECT - GEXTP'
        APSGEXTP LIST=YES
        TITLE 'DSECT - UECA'
        APSUECA LIST=YES
        TITLE 'DSECT - RLST'
        APSURLST LIST=YES
        TITLE 'APSUX07 - INSTALLATION EXIT 7 - RESOURCE EXIT'
APSUX07 CSECT
APSUX07 AMODE 31
APSUX07 RMODE ANY
        USING *,R15
        B START
        DC AL1(16)                LENGTH OF FOLLOWING FIELDS
        DC CL8'APSUX07 '          NAME OF THIS ROUTINE
        DC CL8'&SYSDATE'         DATE OF THIS ASSEMBLY
        DROP R15
START DS 0H
        STM R14,R12,R12(R13)     SAVE CALLERS REGISTERS
        USING APSUX07,BASEREG    NEW ADDRESSABILITY
        LR BASEREG,R15          SWITCH BASE REGISTER
*****
* GET BASING FOR CONTROL BLOCKS *
*****
        USING APSGEXTP,GEXTPTR   SET ADDRESSABILITY TO GEXTP
        USING XTP7,XTP7PTR       SET ADDRESSABILITY TO XTP7
        USING APSUECA,ECAPTR     SET ADDRESSABILITY TO APSUECA
        USING APSURLST,RLSTPTR   SET ADDRESSABILITY TO APSURLST
        USING RLSTFEXT,RLSTEPTR  SET ADDRESSABILITY TO EXTENSION
        L GEXTPTR,0(R1)          LOAD ADDRESS OF GEXTP
        L ECAPTR,XTPECAP         LOAD ADDRESS OF APSUECA
        LR R2,R13                LOAD ADDRESS OF CALLERS SAVE
        LA R13,ECARSCSV          ADDR OF APSUX07 SAVE AREA
        ST R2,4(R13)             SAVE CALLERS SAVE AREA ADDRESS
        ST R13,8(R2)             SAVE APSUX07 SAVE AREA ADDR
        L XTP7PTR,XTPRECP        LOAD ADDRESS OF XTP7
        L RLSTPTR,XTP7LSTP       LOAD ADDRESS OF APSURLST
        L RLSTEPTR,RLSTEXTP      LOAD ADDRESS OF EXTENSION
*****
* DETERMINE WHAT TYPE OF CALL IS BEING MADE TO THE EXIT *
*****
CALLTYPE EQU *
        TM XTP7ETYP,INITCALL     INITIALIZATION CALL?
        BO INITRTN              YES, GOTO INIT ROUTINE
        TM XTP7ETYP,ACCCALL      RESOURCE ACCESS TIME CALL?
        BO ACCRTN               YES, GOTO ACCESS ROUTINE
        B GETOUT                 THIS SHOULD NEVER HAPPEN
*****
* INITIALIZATION ROUTINE: *
* FUNCTION: *
* - REQUEST CONTROL AT ACCESS TIME FOR FORMDEFS AND PAGEDEFS *
* - AND FONTS AND OVERLAYS AND PAGE SEGMENTS *
* - REQUEST CONTROL AT DSE TIME FOR FORMDEFS AND PAGEDEFS *
* - AND FONTS AND OVERLAYS AND PAGE SEGMENTS *
* - SET/RESET XTP7MTOF AND XTP7RFNF *
* - SET RRLVS FOR ALL RESOURCES *
*****
INITRTN EQU *
        MVI XTP7NACC,XTP7AFD+XTP7APD+XTP7ACF+XTP7AM0+XTP7APS
        B GETOUT                 EXIT APSUX07
*
*****
* ACCESS ROUTINE: *
* FUNCTION: *

```

```

*      - CHECK DATASET TYPE AND DON'T PROCESS HEADERS, TRAILERS, MSGDS*
*****
*
ACCRTN  EQU  *
        TM    XTP7DSAT,XTP7PDFT+XTP7PJHD+XTP7PJTR+XTP7PDSH+XTP7PMDS
*
        TEST FLAGS FOR NON-USER DATASET
        BNZ   GETOUT          EXIT IF NON-USER
        TM    XTP7RTYP,FONT    IS THIS A FONT?
        BO    ACCFONT         YES
        B     GETOUT          EXIT, WE'RE DONE
*****
* RESOURCE ACCESS ROUTINE: SUBSTITUTION FOR FONT CODED FONTS.
* CHECK PRINTER'S FONT TECHNOLOGY AND DETERMINE WHICH TYPE OF
* SUBSTITUTION TO DO.
*****
ACCFONT WTO  '***EXIT 07 ACCESS FOR FONT***',ROUTCDE=2,DESC=6
        CLC   RLSTMISC,OFNTPRT    CHECK IF OUTLINE FONT PRINTER
        BNE   SBFONT2            GO TO RASTER SUBSTITION
*****
* SUBSTITUTE AN OUTLINE CODED FONT FOR RASTER CODED FONT
* AND CHANGE THE SIZE OF THE OUTLINE CODED FONT
*****
SBFONT1 WTO  '**OUTLINE FONT PRT **',ROUTCDE=2,DESC=6
        CLC   RLSTNAME,SUBFONT1   CHECK TO SEE IF WE GOT A MATCH
        WTO   '**FONT 1 MATCHES **',ROUTCDE=2,DESC=6
        BNE   NEXTFONT           IS THERE ANOTHER FONT?
        MVC   RLSTNAME,NEWFONT1   SUBSTITUTE NEW FONT
        WTO   '**FONT 1 SUBSTITUTED**',ROUTCDE=2,DESC=6
        MVC   RLSTVFS(2),VFS      SET VERTICAL FONT SIZE
        MVC   RLSTHSCF(2),HSF     SET HORIZONTAL SCALE FACTOR
        OI    RLSTAFLG,RLSTMSG    ISSUE SUBSTITUTION MESSAGE
        B     NEXTFONT           IS THERE ANOTHER FONT IN LIST?
*****
* SUBSTITUTE A RASTER CODED FONT FOR AN OUTLINE CODED FONT
*****
SBFONT2 WTO  '**RASTER FONT PRT **',ROUTCDE=2,DESC=6
        CLC   RLSTNAME,SUBFONT2   CHECK TO SEE IF WE GOT A MATCH
        BNE   NEXTFONT           IS THERE ANOTHER FONT?
        WTO   '**FONT 2 MATCHES **',ROUTCDE=2,DESC=6
        MVC   RLSTNAME,NEWFONT2   SUBSTITUTE NEW FONT
        WTO   '**FONT 2 SUBSTITUTED**',ROUTCDE=2,DESC=6
        OI    RLSTAFLG,RLSTMSG    ISSUE SUBSTITUTION MESSAGE
        B     NEXTFONT           IS THERE ANOTHER FONT IN LIST?
*****
* GO TO THE NEXT FONT IN LIST
*****
NEXTFONT EQU  *
        CLC   RLSTNEXT,ZERO       IS THERE ANOTHER RLST?
        BE    GETOUT             EXIT, WE ARE DONE
        L     RLSTPTR,RLSTNEXT    YES, MOVE POINTER TO NEXT RLST
        L     RLSTEPTR,RLSTEXTP   LOAD ADDRESS OF EXTENSION
        B     ACCFONT            BRANCH TO CHECK THE NEXT FONT
*****
* EXIT LINKAGE
*****
*
GETOUT  EQU  *
        SLR   R15,R15             PSF EXPECTS ZERO RETURN CODE
        L     R13,4(R13)         RESTORE CALLERS SAVE AREA ADDR.
        L     R14,12(R13)        RESTORE CALLERS RETURN ADDRESS
        LM    R0,R12,20(R13)     RESTORE CALLERS REGISTERS
        BR    R14                RETURN TO CALLER
*
*****
* EQUATES TO VERIFY TYPE OF EXIT CALL, COMPARE WITH XTP7ETYP
*

```

```

*****
INITCALL EQU B'10000000' RESOURCE MANAGER INITIAL CALL
ACCCALL EQU B'00100000' RESOURCE ACCESS
*****
* EQUATES TO VERIFY TYPE OF RESOURCE CALL, COMPARE WITH XTP7RTYP *
*****
FONT EQU B'00100000' CODED FONT
*****
* REGISTER EQUATES *
*****
R0 EQU 0
R1 EQU 1
R2 EQU 2
R3 EQU 3
R4 EQU 4
R5 EQU 5
R6 EQU 6
R7 EQU 7
R8 EQU 8
R9 EQU 9
R10 EQU 10
R11 EQU 11
R12 EQU 12
R13 EQU 13
R14 EQU 14
R15 EQU 15
GEXTPTR EQU R5
XTP7PTR EQU R6
ECAPTR EQU R7
RLSTPTR EQU R8
RLSTEPTR EQU R9
CULPTR EQU R11
BASEREG EQU R12
*
*****
* RESOURCE NAME TO BE COMPARED FOR SUBSTITUTION *
*****
*
SUBFONT1 DC CL8'X0KN12 ' RASTER CODED FONT NAME
SUBFONT2 DC CL8'XZNEWFNT' OUTLINE CF NAME
ZERO DC F'0'
OFNTPRT DC X'80'
*
*****
* NEW RESOURCE NAME TO BE USED *
*****
*
NEWFONT1 DC CL8'XZ6980 ' OUTLINE CODED FONT NAME
NEWFONT2 DC CL8'X0KN12 ' RASTER CODED FONT NAME
*****
* CONSTANTS TO CHANGE SIZE AND EXPAND THE FONT WIDTH FOR OUTLINE *
* FONTS *
*****
*
VFS DC H'300' VERTICAL FONT SIZE (PT SIZE X 20)
HSF DC H'450' HORIZONTAL SCALE FACTOR (1.5 THE
* VERTICAL FONT SIZE) *
*
END

```


Table Definitions			
-------------------	--	--	--

<u>id</u>	<u>File</u>	<u>Page</u>	<u>References</u>
HEADER	OFSPCFY	18	19, 19, 19, 20, 22
AROW	OFSPCFY	18	18, 19, 19, 19, 19, 19, 20, 20, 22, 22

Figures			
---------	--	--	--

<u>id</u>	<u>File</u>	<u>Page</u>	<u>References</u>
SAMPLE	OFWHATIS	3	1 3
FLOW	OFSYS	6	2 5
MCF2	OFMCF	21	3
CFNT	OFCF	23	4

Headings			
----------	--	--	--

<u>id</u>	<u>File</u>	<u>Page</u>	<u>References</u>
SNOTICE	OFCOOKBK SCRIPT	iii	Notices
TRADE	OFCOOKBK SCRIPT	iii	Trademarks and Service Marks
PUBS	OFCOOKBK SCRIPT	v	Related Publications
TECH	OFTECH	9	2.4, Inside Outline Font Technology 4
STYLES	OFSTYLES	9	2.5, What Outline Font Styles are Available
PSFREQ	OFREQMT	11	3.2.1, PSF/MVS
APP	OFREQMT	13	3.2.3, Applications 12
FONTUTI	OFREQMT	14	3.2.3.1, Font Utilities — Type Transformer and Others 8, 8, 13
SPCFY	OFSPCFY	15	4.0, How Do You Specify Outline Fonts? 8
JCL	OFSPCFY	15	4.1, JCL
SPECUS	OFSPCFY	18	4.5, Custom applications and formatting 14
MCFXMP	OFMCF	21	4.6, Sample MCF2 Structured Fields 18
CFXMP	OFCF	23	4.8, Sample Coded Font Structured Fields 22
MIGRATE	OFMIGRAT	25	5.0, Migrating to Outline Fonts
EXIT	OFMIGRAT	26	5.3, Coding an Exit to Substitute Raster and Outline Fonts 8, 12, 15, 25
SUBEXIT	OFEXIT	29	Appendix A, Sample Font Substitution Exit 26

Processing Options

Runtime values:

Document fileid	OFCOOKBK SCRIPT
Document type	USERDOC
Document style	OFCOOKBK
Profile	EDFPRF40
Service Level	0030
SCRIPT/VS Release	4.0.0
Date	96.01.18
Time	13:06:43
Device	3820A
Number of Passes	4
Index	YES
SYSVAR G	INLINE
SYSVAR H	YES
SYSVAR M	YES
SYSVAR X	YES

Formatting values used:

Annotation	NO
Cross reference listing	YES
Cross reference head prefix only	NO
Dialog	LABEL
Duplex	YES
DVCF conditions file	(none)
DVCF value 1	(none)
DVCF value 2	(none)
DVCF value 3	(none)
DVCF value 4	(none)
DVCF value 5	(none)
DVCF value 6	(none)
DVCF value 7	(none)
DVCF value 8	(none)
DVCF value 9	(none)
Explode	NO
Figure list on new page	YES
Figure/table number separation	YES
Folio-by-chapter	NO
Head 0 body text	(none)
Head 1 body text	(none)
Head 1 appendix text	Appendix
Hyphenation	NO
Justification	NO
Language	ENGL
Keyboard	395
Layout	OFF
Leader dots	YES
Master index	YES
Partial TOC (maximum level)	4
Partial TOC (new page after)	INLINE
Print example id's	NO
Print cross reference page numbers	YES
Process value	(none)
Punctuation move characters	,
Read cross-reference file	(none)
Running heading/footing rule	NONE
Show index entries	NO
Table of Contents (maximum level)	3
Table list on new page	YES
Title page (draft) alignment	CENTER
Write cross-reference file	(none)

Imbed Trace

Page viii	OFINTRO
Page 1	OFWHATIS
Page 4	OFSYS
Page 6	OFADVAN
Page 7	OFLIMIT
Page 8	OFTECH
Page 9	OFSTYLES
Page 10	OFREQMT
Page 14	OFSPCFY
Page 20	OFMCF
Page 22	OFCE
Page 23	OFMIGRAT
Page 26	OFSUMM
Page 28	OFEXIT