Print Services Facility for z/OS
4.7

Customization

IBM

Note

Before using this information and the product it supports, read the information in "Notices" on page 495.

Contents

Figures

Tables

About this publication

This publication provides information about changing and maintaining Print Services Facility (PSF) 4.7.0 for z/OS licensed program (Program Number 5655-M32), hereafter referred to as PSF.

This publication assumes that you have experience in system programming and are familiar with Advanced Function Presentation (AFP) printers. In this publication, the word *printing* refers to presentation on paper, foils, labels, or microfilm. (*Microfilm* can mean either microfiche or 16 mm film.)

Who should read this publication

This publication contains information for use by system programmers who:

- Install PSF by using System Modification Program/Extended (SMP/E).
- Change the system environment.
- Create a communication network.
- Maintain data security.
- React to abnormal events.
- Code exit routines.

You must have system-level programming knowledge, including a thorough understanding of:

- Job control language (JCL)
- Job Entry Subsystem (JES)
- TCP/IP
- Advanced communication functions for non-channel-attached printers:
  - Virtual Telecommunications Access Method (VTAM®)
  - Network Control Program (NCP)

You must also be familiar with PSF on the level described in *PSF for z/OS: User's Guide*.

Why should I read this publication

The purpose of this publication is to help you with the system programmer's primary task of changing and maintaining PSF.

If you need to install PSF, first read *Program Directory for Print Services Facility for z/OS*. The program directory contains most of the instructions for installation. Information about modifying the host system for PSF is presented in this publication.

If you need a description of the job control language (JCL) used to print jobs on AFP printers, a description of resources that are used in AFP printing (data streams, form definitions, page definitions, fonts, page segments, overlays, and object containers), or examples of some common printing tasks, read *PSF for z/OS: User's Guide*. The user's guide contains instructions for using PSF.

How this publication is organized

This publication is organized so that you can read it sequentially or refer directly to individual tasks.

Chapter 1, "Understanding PSF for z/OS," on page 1 helps you understand PSF by describing what programs are included with PSF, what you can do with PSF, and what applications and printers PSF supports.

Chapter 2, "Configuring the system for PSF for z/OS," on page 5 through Chapter 18, "Understanding PSF data protection and security," on page 219 are more specific. They describe:

- How to change the system environment
- How to define channel-attached printers to z/OS
- How to create network nodes for SNA-attached printers
- How to connect and operate a TCP/IP printer
- How to install library data sets
- How to use Infoprint Server Printer Inventory
- How to use PSF parameters
- How to use deferred-printing mode
- How to use direct-printing mode
- How to manage resources and select special processing parameters
- How to refine PSF to affect system and printer performance
- How to use PSF display functions
- How to finish all pages in a job as a group
- How to take advantage of the cut-sheet emulation capability of some AFP printers
- How to use the PSF Side and Edge Sensitive Paper Handling function
- How to handle job streams that are formatted at different resolutions
- How PSF protects data

Chapter 19, "Supporting resident fonts," on page 221 through Chapter 29, "Supporting AFP output on microfilm," on page 341, which are specialized, describe:

- How to use and change the APSRMARK utility to support resident fonts
- How to mark fonts for font capture
- How to use FOCA outline fonts
- How to use TrueType and OpenType fonts
- How to use the APSRMARK utility to mark resources for DPF
- How to use color management resources (CMRs)
- How to use installation exits
- How to recover from print errors
- How to use the Line-Mode Migration function to print line-mode jobs
- How select fonts and mark forms for a 3800 printer and migrate from a 3800
- How to enable users to print to microfilm

Appendixes provide the following reference information:

- How to migrate PSF parameters to the Printer Inventory
- Examples of printer information in hardcopy and softcopy reports and in the system log. Also, all possible record formats of the softcopy report.
- Lists of sample APSRMARK jobs that are supplied by PSF
- Description of an APSRMARK long report format
- A copy of the APSRFTBL module code that lists code pages and character sets for some printers
- Field descriptions for Exit 7
- Information about APTRCONV, the Virtual Storage Extended (VSE) resource conversion utility program
- Information about reasonable resource loaded values (RRLV)
- Information about APSRCF30, the font conversion utility program (to convert 240-pel fonts to 300-pel fonts)
- An example of output that is obtained after you run the installation verification program (IVP)
- Lists of PSF macros for programming interfaces

- Copies of the optional user tables that map raster fonts to outline fonts and outline fonts to raster fonts
- An explanation of how to install and configure an i-data 7913 Intelligent Printer Data Stream (IPDS) Printer LAN Attachment
- How to use the point counting tool, APSGPPCT
- A description of the accessibility features available in z/OS

A notices section, glossary, bibliography, and index are included. The bibliography lists the publications that contain additional information about the system, hardware, or software used by PSF, or about installation and maintenance.

Understanding the syntax notation used in this publication

The following rules apply to syntax and coding illustrations throughout this publication:

- Bold highlighting identifies commands, keywords, files, directories, and other items whose names are predefined by the system, or items that must be entered as is, such as **DUPLEX** and **BLOCK**.
- Variable data is printed in italics. Enter specific data to replace the characters in italics; for example, for PRT*nnnn* you might enter PRT0002. Italics also identify the names of publications.
- Monospacing identifies examples of specific data values, examples of text similar to what you might see displayed, examples of portions of program code similar to what you might write as a programmer, messages from the system, or information you must actually type.
- Do not enter the following symbols as part of a parameter or option:

  Vertical Bar |
  Underscore ___
  Brackets [ ]
  Braces { }
  Ellipsis ...

- A vertical bar between two values means that you select only one of the values.
- An underscored value means that if an option is not specified, the underscored value, called the default, is used.
- Brackets around a value mean that you do not have to select the value; the value is optional.
- Braces around a value mean that you must select one of the mutually exclusive values. For example, `{ THIS | THAT }`
- An ellipsis that follows a command or set of commands indicates that the command or set of commands can be repeated.

Related information

Publications that are referred to in this document or that contain more information about AFP and related products are listed in the . For information about all z/OS product publications, see *z/OS Information Roadmap*.

For more information about z/OS and PSF for z/OS, see these web pages:

- z/OS home page (www.ibm.com/systems/z/os/zos)
- IBM Documentation (www.ibm.com/docs/en)

To obtain the latest documentation updates for PSF for z/OS, see the appropriate SYS1.SAMPLIB members in .

Table 1. SYS1.SAMPLIB members for PSF documentation updates	
Member	**Publication**
APSGADP7	*PSF for z/OS: AFP Download Plus*

Table 1. SYS1.SAMPLIB members for PSF documentation updates (continued)	
Member	**Publication**
APSGCUS7	*PSF for z/OS: Customization*
APSGDGN7	*PSF for z/OS: Diagnosis*
APSGDLG7	*PSF for z/OS: Download for z/OS*
APSGMAC7	*PSF for z/OS: Messages and Codes*
APSGSEC7	*PSF for z/OS: Security Guide*
APSGUSR7	*PSF for z/OS: User's Guide*

How to send your comments to IBM

We invite you to submit comments about the z/OS product documentation. Your valuable feedback helps to ensure accurate and high-quality information.

Important: If your comment regards a technical question or problem, see instead "If you have a technical problem" on page xxix.

Submit your feedback by using the appropriate method for your type of comment or question:

Feedback on z/OS function
> If your comment or question is about z/OS itself, submit a request through the IBM RFE Community (http://www.ibm.developerworks/ref) .

Feedback on IBM Documentation function
> If your comment or question is about the IBM Documentation functionality, for example search capabilities or how to arrange the browser view, send a detailed email to IBM Documentation Support at ibmdoc@us.ibm.com.

Feedback on the z/OS product documentation and content
> If your comment is about the information that is provided in the z/OS product documentation library, send a detailed email to mhvrcfs@us.ibm.com. We welcome any feedback that you have, including comments on the clarity, accuracy, or completeness of the information.
>
> To help us better process your submission, include the following information:
>
> - Your name, company/university/institution name, and email address
> - The following deliverable title and order number: *PSF for z/OS: Customization*, S550-0427
> - The section title of the specific information to which your comment relates
> - The text of your comment.

When you send comments to IBM, you grant IBM a nonexclusive authority to use or distribute the comments in any way appropriate without incurring any obligation to you.

IBM or any other organizations use the personal information that you supply to contact you only about the issues that you submit.

If you have a technical problem

If you have a technical problem or question, do not use the feedback methods that are provided for sending documentation comments. Instead, take one or more of the following actions:

- Go to the IBM Support Portal (support.ibm.com).
- Contact your IBM service representative.
- Call IBM technical support.

Summary of changes

PSF for z/OS Version 4 Release 7

This content includes terminology, maintenance, and editorial changes to information previously presented in *PSF for z/OS: Customization*, S550-0427-05. Technical additions or changes to the text and illustrations are indicated by a vertical line to the left of the change.

General changes (December 2019)

- References to PSF for z/OS 4.5 and z/OS 2.1 are removed because those releases are no longer in service.

New information (December 2019)

- Inline resource storage above the bar
  - "Selecting MEMLIMIT sizes" on page 195
  - APAR OA54593
    - "Updating MAXPROCSYS and MAXPROCUSER" on page 7

Changed information (June 2021)

- TCP/IP 64K buffer support (APAR OA59773)
  - "Understanding performance issues " on page 48
  - "FSA parameters" on page 62
  - "Parameters for the AFPPARMS control statement" on page 112
- Blank truncation (APAR OA60960)
  - "FSS parameters" on page 58
  - "Defining PSF FSSs and FSAs for printers" on page 123
  - "PARM parameters on the EXEC statement" on page 144

Changed information (December 2019)

- Inline resource storage above the bar
  - "Common message log" on page 55
  - Chapter 8, "PSF parameters," on page 57
  - "FSA parameters" on page 62 (formerly part of "Specifying Printer Inventory parameters" on page 57
  - "Specifying AFPPARMS control statement parameters" on page 111
  - "Selection hierarchy for AFPPARMS parameters" on page 112
  - "Parameters for the AFPPARMS control statement" on page 112
  - Chapter 10, "Using direct-printing mode," on page 167
  - "Inline resources" on page 188
  - "Glossary" on page 499
- APAR OA54593
  - "Updating MAXPROCSYS and MAXPROCUSER" on page 7

- "Specifying Printer Inventory parameters" on page 57 has been split into multiple topics.

PSF for z/OS Version 4 Release 6

This content includes terminology, maintenance, and editorial changes to information previously presented in *PSF for z/OS: Customization*, S550-0427-04.

General changes

- References to PSF 4.3.0 and 4.4.0 for z/OS are removed because they are no longer in service.
- References to z/OS 1.13 are removed because that release is no longer in service.
- References to AIX® 6.1.0 are removed because it is no longer in service.
- "System z" is changed to "IBM Z", and "S/390" is removed.
- References to Ricoh products, including "InfoPrint Manager" and "Ricoh ProcessDirector" are removed.
- "Distributed Print Function (DPF)", "PSF Direct", and "Workstation Print Manager (WPM)" are changed to "distributed print function (DPF)", "PSF direct", and "workstation print manager (WPM)".
- "InfoPrint AFP Resource Installer" is removed or changed to "AFP resource installer".

New information

- The MO:DCA AFP/Archive (AFP/A), AFP/A, IS/3, and Graphics Arts Function Set are new. See Chapter 1, "Understanding PSF for z/OS," on page 1 and "Setting up MO:DCA interchange set auxiliary pages" on page 116.
- End-data-set call (EDSC) is added to Chapter 7, "Using the Infoprint Server Printer Inventory," on page 53, "Specifying format resolution" on page 217, "Format resolution order" on page 217, Table 30 on page 263, "Control flow with Exit 7" on page 274, Table 32 on page 296, "Initialization (INIT) call" on page 301, "End-data-set call (EDSC)" on page 313, "XTP7 parameter area format" on page 421, "XTP7 fields" on page 429, and Table 42 on page 448.
- The **Resource deletion ratio** parameter is added to the Printer Inventory in Figure 32 on page 63.
- These new Printer Inventory parameters are added to Table 7 on page 66.
  - **Resource deletion ratio**
  - **SNMP enabled**
  - **Trace user data**
- These new AFPPARMS parameters are added to Table 10 on page 113:
  - **resource-deletion-ratio**
  - **snmp-enabled**
  - **trace-user-data**
- "Expediting resource deletion" on page 189 describes the new resource deletion function. See also "Using RDC to control resources" on page 312.

Changed information

- The web page of PSF-supported AFP/IPDS printers is removed from "Printers supported by PSF" on page 2.
- A reference for printing to channel-attached printers is added in "Changing the system environment" on page 5.
- A new paragraph about specifying CBLOC VIRTUAL31(IHAASVT) in SYS1.PARMLIB is added to "Special considerations for using channel-attached printers" on page 9.

- "Understanding performance issues " on page 48 is updated with recommended values for TCPSENDBFRSIZE.
- Information about MAXPROCSYS and MAXPROCUSER parameters is moved from "Common message log" on page 55 to a new section called "Updating MAXPROCSYS and MAXPROCUSER" on page 7.
- The Printer Inventory **Auxiliary files MO:DCA level** parameter in Table 7 on page 66 is updated with the **BPF match** attribute.
- The AFPPARMS **auxiliary-files-modca-level** parameter in Table 10 on page 113 is updated with the **bfp-match** attribute.
- "Setting up MO:DCA interchange set auxiliary pages" on page 116 is updated with information for setting the AFPPARMS **auxiliary-files-modca-level** parameter.
- The font library FONTOLN is changed to SFNTILIB in "Sample PSF startup procedure for four printers" on page 137 and "Resource libraries segmented for multiple resolution" on page 215.
- The INV parameter description in the EXEC statement is updated. See "PARM parameters on the EXEC statement" on page 144.
- "Structure of SMF type 6 records" on page 161 is updated.
- In the description of the ECEJOBPP field in Figure 92 on page 271, "(includes header, message, and trailer sheets)" is added.
- In the descriptions of the ECELPCNT and ECEPPCNT fields, " for the data set" is added. See Figure 92 on page 271, "ECELPCNT" on page 273, and "ECEPPCNT" on page 273.
- Figure 82 on page 259 is updated.
- Updates are made to Appendix N, "APSGPPCT point counting tool," on page 487:
  - "PSF Tier Level Report" on page 488 is updated with an IBM-supplied form definition, F1PPCT01, instead of a default form definition for generating a printed report.
  - "APSGPSPD printer speed table" on page 489 is updated with a reference to Figure 82 on page 259 for compiling and link-editing an assembler module on z/OS.
  - The AMODE and RMODE values in Figure 190 on page 490 are updated.
- "Glossary" on page 499 is updated with the definition for MO:DCA Graphic Arts Function Set.

Deleted information

- The reference to "Compatibility Fonts" in "Optional programs" on page 2 is removed because IBM Compatibility Fonts is included in z/OS Font Collection V2.3 and is no longer a feature of PSF 4.6.
- Simple Network Management Protocol (SNMP) support is removed from Chapter 7, "Using the Infoprint Server Printer Inventory," on page 53 because Infoprint Server no longer supports the SNMP subagent. The SNMP reporting parameter is also removed from Figure 32 on page 63 and Table 7 on page 66.

Chapter 1. Understanding PSF for z/OS

Print Services Facility (PSF) for z/OS (referred to as PSF) is a licensed printer-driver program that manages and controls the input and output data streams required by supported Advanced Function Presentation (AFP) printers or microfilm devices. PSF combines print data with other resources and printing controls to produce AFP data that can be printed.

PSF supports Mixed Object Document Content Architecture (MO:DCA) Presentation Interchange Set (IS) data streams, including:

MO:DCA AFP/Archive (AFP/A)
MO:DCA AFP/A is an AFP document architecture interchange set that is used for long-term preservation and retrieval. This subset ensures page independence and eliminates images without clearly specified resolution, device default fonts, and external resources.

MO:DCA Interchange Set 3 (IS/3)
MO:DCA IS/3 is the first interchange set to achieve industry consensus through a rigorous open standards process. It improves existing functions and introduces new functions, such as Begin Print File (BPF) and End Print File (EPF) structured fields, and multiple image TIFF object support.

MO:DCA AFP/A, IS/3
MO:DCA AFP/A, IS/3 is an AFP document architecture interchange set that complies with the rules and restrictions of both the AFP/Archive and IS/3 interchange sets.

MO:DCA Graphic Arts Function Set (GA)
MO:DCA GA is an extension of MO:DCA IS/3 that adds PDF presentation object support.

See *Mixed Object Document Content Architecture Reference* for more information.

PSF can be used in either of two ways:

- In *deferred-printing mode*, PSF serves as an output writer under Job Entry Subsystem (JES). It processes the spooled output from JES and sends a data stream to a page printer.

- In *direct-printing mode*, PSF serves as the access method to process an output data set. Printers that are attached by using Systems Network Architecture (SNA) or Transmission Control Protocol/Internet Protocol (TCP/IP) cannot be used in direct-printing mode.

For more information about:

- Using PSF to print, see *PSF for z/OS: User's Guide*.

- AFP and AFP products, see *Guide to Advanced Function Presentation*.

- Printing to microfilm, see Chapter 29, "Supporting AFP output on microfilm," on page 341.

- Submitting jobs to print to microfilm, see *PSF for z/OS: User's Guide*.

PSF's capabilities

You can use PSF to do these:

- Receive print jobs, access resources for print jobs, and send print commands to a printer. PSF supports both deferred printing, under JES; and direct printing, under the control of an application program.

- Manage resources required for the print job, such as form definitions, page definitions, fonts, page segments, object containers, and overlays.

- Handle print jobs that are formatted at different resolutions and select the resource libraries with the correct resolution to print the data.

- Provide operator control of printers in deferred-printing mode.

- Provide problem diagnosis and error recovery.

- Restart printing from the point of an error or from the last checkpoint.

- Write accounting records.
- Write separator pages between print jobs or copies of print jobs.
- Let installations manage resources; modify output records, separator pages, and accounting records; and inspect messages.
- Perform finishing operations, such as stapling or binding.

For more information about PSF's capabilities and new enhancements, see *PSF for z/OS: Introduction*.

Printers supported by PSF

The information in this document applies to IBM printers that are supported by PSF; however, not all models of those printers are supported. You need to be familiar with the specific printers installed on your system.

For information about a supported printer, see "Displaying printer information" on page 201 or the documentation provided with the printer.

If you are installing a 300-pel printer, and you do not have a 300-pel font library, you can use the z/OS Font Collection, which is a base feature of z/OS.Higher resolution single-byte and double-byte fonts are available with single-byte and double-byte outline fonts in these collections.

Note: The microfilm device can operate as either a 240-pel device or a 300-pel device. For more information, see Chapter 29, "Supporting AFP output on microfilm," on page 341.

If you plan to use the PSF print-labeling function, see *PSF for z/OS: Security Guide*. The information that you are now reading does not contain specific information about the print-labeling function.

PSF support for intermediate devices

PSF supports printing through the following programs that run on intermediate devices, such as personal computers:

- Distributed print function, called DPF
- PSF direct
- Workstation print manager, called WPM

The microfilm device is not supported by intermediate devices for printing to microfilm. To see if a printer supports an intermediate device, see "Displaying printer information" on page 201 or the documentation provided with the printer.

With DPF and WPM, you can store PSF resources in the fixed-disk storage of the personal computer; the stored resources can then be used by a printer that is attached to the intermediate device. PSF tells DPF or WPM not to store resources that are found in user libraries or inline resources. The Printer Inventory or Exit 7 can specify that inline resources are captured. See Capture inline resources for more information. The PSF direct program does not store resources in the intermediate device.

What is included with PSF

PSF includes optional programs, including separately priced and ordered features, and several tools and samples.

Optional programs

The following optional programs are either supplied with PSF or are separately priced and ordered features. These optional programs are on their own tapes, with their own installation instructions in separate program directories, and with their own separate publications:

- AFP Conversion and Indexing Facility (ACIF) — priced feature
- AFP Download Plus — priced feature

- AFP Upload — priced feature
- Download for z/OS — priced feature

These programs can assist you in:

- Converting and indexing data
- Transmitting data to and from your z/OS system

For details, see the publications for these programs.

Tools and samples

Several tools and samples are supplied with PSF, including:

AFP Reblocking Program (AFRREBLK)
AFRREBLK restructures AFP file data that is transferred from a workstation to a z/OS host so that the data records can be printed with PSF. For information about using the AFP Reblocking Program, see *PSF for z/OS: User's Guide*.

Color Mapping Tool
The Color Mapping Tool generates color mapping tables. A color mapping table translates certain Mixed Object Document Content Architecture (MO:DCA) structured fields to color structured fields used by more recent printers. For information about the Color Mapping Tool and color mapping tables, see *PSF for z/OS: User's Guide*.

Font Conversion Program (APSRCF30)
APSRCF30 is a font conversion program that uses your currently installed 240-pel fonts as input, converts them to 300-pel fonts, and places the converted output into a partitioned data set. For more information about this tool, see Appendix I, "Font conversion program, 240-pel to 300-pel (APSRCF30)," on page 457.

Installation Exits
You can use installation exits to modify PSF functions, such as separator pages, output records, System Management Facilities (SMF) type 6 records, PSF messages, and resources. For more information, see Chapter 25, "Using installation exits," on page 257.

Point Counting Tool (APSGPPCT)
APSGPPCT is a batch program that determines the correct software license for PSF, based on the printer points assigned to each printer in your installation. For more information, see Appendix N, "APSGPPCT point counting tool," on page 487.

Resident Font Table (APSRFTBL)
APSRFTBL identifies the symbol sets that are resident in printers, such as 4224, 4230, 4234, 4247, 6408, and 6412. APSRFTBL contains tables that associate the font character set and code page for a particular host font with the identifier of the resident symbol set. For more information about this tool, see "PSF resident font table (APSRFTBL)" on page 221.

Resident Font Utility (APSRMARK)
APSRMARK is a batch program that identifies host-font resources, such as code pages and character sets, as potentially printer-resident. For more information about this tool, see "PSF resident font utility—APSRMARK" on page 223.

Resource Conversion Utility for VSE (APTRCONV)
APTRCONV is a utility program that converts resources so that VSE can use them. You run APTRCONV with JCL to convert an AFP resource in a partitioned data set to a VSE job stream. For more information about this tool, see Appendix G, "APTRCONV resource conversion utility for VSE," on page 453.

Resources
Form definition and page definition resources are supplied with PSF. You can customize these resources for your organization. For more information about the form definitions and page definitions that are supplied with PSF, see *PSF for z/OS: User's Guide*.

Chapter 2. Configuring the system for PSF for z/OS

After you install PSF for z/OS, you must ensure that PSF is enabled. Then, before you can use PSF, you must change the system environment—system libraries, JES, and VTAM. The changes that you make depend on whether PSF is already installed, which printers are being used, and whether printers are controlled with JES. You can also optionally change the internal default program properties table (PPT) entry for PSF and update the MAXPROCSYS and MAXPROCUSER BPXPRMxx parameters in the SYS1.PARMLIB member.

Enabling PSF in the SYS1.PARMLIB member

After you install PSF, ensure that SYS1.PARMLIB contains member IFAPRD*xx* and that PSF is enabled in the member. Figure 1 on page 5 shows the PRODUCT entry that IFAPRD*xx* must contain for PSF. Note that the STATE value is set to ENABLED.

```
PRODUCT OWNER('IBM CORP')
        NAME('PSF for z/OS')
        ID(5655-M32)
        VERSION(*)
        RELEASE(*)
        MOD(*)
        FEATURENAME('PSF for z/OS')
        STATE(ENABLED)
```

Figure 1. PRODUCT entry for PSF in your IFAPRDxx member of SYS1.PARMLIB

Changing the system environment

To change the system environment for PSF for z/OS:

1. Set up the printer for either deferred-printing mode or direct-printing mode:

   - To set up the printer for deferred-printing mode with JES, you must change the system to enable PSF to act as the interface between JES and the printer. The following modifications are required:

     a. Define the PSF printer to JES. The initialization parameters for JES2 or JES3 must be changed for a PSF functional subsystem (FSS). For more information about JES, see Chapter 9, "Using deferred-printing mode," on page 123.

        You must understand the hierarchy by which resources are selected when the printer is used with JES. The JES forms control buffer (FCB) and universal character set (UCS) parameters and work-selection criteria are described in Chapter 11, "Managing resources," on page 173.

     b. Update the provided startup procedure—APSWPROA, APSWPROB, APSWPROC, APSWPROM, or APSWPROT, whichever is used for the installation verification program (IVP). For an example of IVP output from a 240-pel printer, see Appendix J, "Installation verification program example," on page 463. These procedures are placed in SYS1.PROCLIB when PSF is installed.

   - To set up the printer for use in direct-printing mode (host-connected, channel-attached non-SNA printers only), see Chapter 10, "Using direct-printing mode," on page 167. You cannot use microfilm devices in direct-printing mode.

   Note: You cannot set up a printer to use both direct-printing mode and deferred-printing mode at the same time.

2. Establish security for PSF startup procedures and resource libraries:

   - Define the startup procedure that you are using to the RACF® STARTED class. Each startup procedure requires a data segment (STDATA) entry with the user ID, group name, or both that you assigned to the printer.

Note: The RACF user ID or group name that is associated with the PSF startup procedure must be defined with an OMVS segment and a home directory.

The data segment also requires these characteristics: TRUSTED=NO, PRIVILEGED=NO, TRACE=NO (see Figure 2 on page 6 for an example). If you want to use the RACF Started Procedures Table (ICHRIN03) instead, see *z/OS Security Server RACF Security Administrator's Guide* for more information.

- Ensure that PSF has access to RACF-protected resource libraries (such as FONTLIBs, PDEFLIBs, FDEFLIBs). Use the RACF PERMIT command to authorize the user or group you specified in the RDEFINE command to the resource libraries used by PSF.

```
RDEFINE STARTED APSWPROT.*
                    STDATA(USER(userid)
               GROUP(groupname)
               TRUSTED(NO)
               PRIVILEGED(NO)
               TRACE(NO))
```

Figure 2. Defining your startup procedure in the STARTED profile (sample)

In Figure 2 on page 6, APSWPROT.* represents the name of your PSF startup procedure.

You might also choose to bypass security checking and run PSF as TRUSTED or PRIVILEGED. For more information about the TRUSTED or PRIVILEGED attribute, see *RACF System Programmer's Guide* for your operating system.

3. Set up for SNA-attached printers:

   An SNA-attached printer is a networked device that uses Communication Server SNA Services (VTAM) alone or with Advanced Communication Function for Network Control Program (NCP) to communicate with your system. Before you can use the printer, you must update the VTAM network definition or do an NCP generation.

   All printer configurations that contain a 37X5 communication controller require NCP and VTAM. All other SNA configurations require only VTAM.

   For more information, see Chapter 4, "Establishing network nodes for SNA-attached printers," on page 17.

For more information about host-connected printers that are channel-attached, see "Defining channel-attached printers" on page 11. If you are using PSF to print to channel-attached printers, see the restriction in "Special considerations for using channel-attached printers" on page 9. For information about printers that are attached with TCP/IP, see Chapter 5, "Connecting a TCP/IP-attached IPDS printer," on page 41.

For information about setting up PSF to use TrueType and OpenType fonts, see Chapter 22, "Using TrueType and OpenType Fonts," on page 243.

Default PPT entry

z/OS supplies an internal default program properties table (PPT) entry for PSF. The internal default values are:

```
PPT     PGMNAME(APSPPIEP)
        KEY(1)
        NOSWAP
        NOPRIV
        CANCEL
        SYST
        NODSI
        PASS
        AFF(NONE)
        NOPREF
```

These are the recommended values and no changes are required. However, you can change them by specifying a PPT entry in the SCHEDxx member in the system PARMLIB with overriding values for

APSPPIEP. For more details, see *z/OS MVS Initialization and Tuning Reference*. Modification of any values other than NODSI might cause a JES abend, or other unexpected results to occur.

By specifying NODSI (nonexclusive use of data sets) in the PPT, other programs, such as OGL and PPFA, can change the resources while PSF is running. NODSI also allows Data Management functions to occur, such as compression and library migration. However, these functions can cause problems for PSF if they run while PSF is active. Therefore, you must manually exclude PSF resource libraries from the Data Management functions. If you do not want to manually exclude PSF resource libraries, you need to use DSI in the PPT entry.

Specifying DSI and running PSF with DISP=SHR causes other programs to wait until PSF no longer has the PSF resource libraries open. Therefore, DSI prevents programs from updating or changing the resource libraries while PSF is active. Typically, PSF has the resource libraries open while any FSA is running. If the **Close libraries when idle** parameter in the Printer Inventory is YES or XTP7CLOS in Exit 7 is set to ON, PSF has the resource libraries open only while any FSA is actively printing (see Close libraries when idle for more information).

Updating MAXPROCSYS and MAXPROCUSER

The operator command D OMVS,O displays the current values for the MAXPROCSYS and MAXPROCUSER BPXPRMxx parameters in the SYS1.PARMLIB member. The operator command D OMVS,L displays the CURRENT USAGE, HIGHWATER USAGE, and SYSTEM LIMIT for the MAXPROCSYS parameter.

To update the MAXPROCSYS and MAXPROCUSER parameters, increase the value by 2 for each FSS and by 1 for each FSA.

See *z/OS MVS Initialization and Tuning Reference* for information about setting the MAXPROCUSER and MAXPROCSYS parameters.

Chapter 3. Using channel-attached printers with PSF

This information describes channel-attached printers and the types of channels supported by PSF. It discusses special considerations for using channel-attached printers. It also discusses sharing channel-attached printers with PSF and defining AFP channel-attached printers.

To define a channel-attached printer as a TCP/IP-attached printer, see .

Overview

IBM printers can be channel-attached:

- Directly to the host
- Through switches, directors, or channel-extenders to the host
- To an intermediate device, such as a personal computer through a distributed print function (DPF).

PSF supports these types of channel-attached printers: 3800, 3820, AFP1 and microfilm devices. However, IBM no longer provides service for the 3800 and 3820 printers.

PSF supports these types of channel attachments:

- Parallel channel
- ESCON channel
- FICON® Bridge channel
- FICON channel

To determine which type of channel your printer supports, see your printer's documentation.

Although channel-attached printers can be connected to an intermediate device, such as DPF, PSF communicates to these printers by using the SNA protocol. The intermediate device then communicates to the printer by using the channel. When you are setting up communications between PSF and an intermediate device, see and the publications for the intermediate device.

Special considerations for using channel-attached printers

You can specify CBLOC VIRTUAL31(IHAASVT) in the DIAG*xx* member of SYS1.PARMLIB with the IHAASVT control block above the line. However, PSF requires the IHAASVT control block to be below the line when you are printing to channel-attached printers. Therefore, if you are printing to channel-attached printers, ensure that IHAASVT is NOT specified with the CBLOC VIRTUAL31 statement in the DIAG*xx* member.

A PSF-supported, channel-attached printer can be attached to a VM/ESA host with any of the channel-attachments PSF supports, depending on the printer. You can then run PSF for z/OS on a z/OS guest and use the PSF-supported, channel-attached printer. To do this, the printer must be defined to the VM/ESA and z/OS I/O configurations. For more information about defining the VM/ESA I/O configuration, see *VM/ESA CP Planning and Administration*.

You can use the FICON Bridge channel on the 9032–005 ESCON director to attach a PSF-supported, ESCON channel-attached printer to a FICON channel. For more information about the FICON Bridge channel, see *Planning for the 9032 Model 5 Director*.

VM/ESA 2.2.0 or later is required to use a FICON channel-attached printer attached to a VM/ESA host with PSF for z/OS running on a z/OS guest.

A 3820 printer can be channel-attached to the host; therefore, it must be defined to the I/O configuration. Because PSF communicates with the printer by using the SNA protocol, the printer must also be defined

to VTAM. See Chapter 4, "Establishing network nodes for SNA-attached printers," on page 17 for information about defining a 3820 printer (LU 6.2 device type) to VTAM.

To use a 3800 printer with PSF, DFSMSdfp, which contains the device support code for the 3800, must be installed on your system. PSF supports only the 3800 Model 3, 6, and 8 in Page Mode. For more information about the 3800 Model 3 and 6, see the *IBM 3800 Printing Subsystem Model 3 and 6 Introduction*. For more information about the 3800 Model 8, see the *IBM 3800 Printing Subsystem Model 8 Introduction*.

For more information about using PSF and channel-attached microfilm devices, see Chapter 29, "Supporting AFP output on microfilm," on page 341. For performance considerations when you are using AFP channel-attached printers, see Chapter 12, "Understanding AFP performance," on page 185.

Sharing channel-attached printers

Channel-attached printers can be shared among multiple PSFs on the same or different hosts. However, no matter what the I/O configuration of the printer is (for example, multiple paths or switches), the printer can be in use by only one PSF at a time. See "Defining channel-attached printers" on page 11 for examples of how to share printers.

The operator determines which PSF uses the printer by stopping PSF and varying the printer offline on one host and then varying the printer online and starting PSF on another host. Having the operator determine which PSF uses the printer can be error prone, particularly if the hosts are in physically separate locations.

If multiple PSFs drive the printer concurrently, I/O errors, job output problems, or both can occur. I/O errors, such as `Start pending (IOS071I)` or `No path available (IOS551I)` can occur or the job output from each PSF can be interleaved.

The multi-host feature on ESCON or FICON channel-attached printers can be used to ensure that only one PSF is using the printer at a time. If a printer is shared by PSFs on the same host, PSF's use of dynamic allocation ensures that only one PSF can use the printer at a time.

"Parallel channel: two-channel switch feature" on page 10 and "ESCON and FICON channel" on page 10 describe sharing a printer among PSFs on different hosts.

Parallel channel: two-channel switch feature

IBM parallel-channel printers, such as the InfoPrint 4000, provide an optional two-channel switch feature, which consists of a second parallel-channel adapter in the printer. This two-channel switch feature can be used to share the printer between two different PSFs, each on different hosts. See your printer's documentation to determine whether it supports the two-channel switch feature.

By defining one of the channel attachments to one host, defining the other channel attachment to the other host, and configuring the switch in manual mode, the printer can be shared between two hosts. Because only one of the attachments can be online at a time, the printer can be used by only one PSF at a time. The operator must manually operate the switch to control which host has access to the printer.

ESCON and FICON channel

An IBM ESCON or FICON channel-attached printer can be shared among multiple PSFs on the same or different hosts, but it can be in use by only one PSF at a time. The operator manually controls this sharing by stopping PSF and varying the printer offline on one host and then varying the printer online and starting PSF on another host.

IBM printers with a FICON channel adapter, or an ESCON PCI channel adapter with microcode level 9.6 or later, have a multi-host feature that supports multiple concurrent paths to the same printer. When the multi-host feature is enabled, the printer allows only one PSF to use the printer at a time. Multi-host support is also available if an ESCON channel-attached printer is attached to a FICON Bridge channel. For more information about the FICON Bridge feature, see *Planning for 9032 Model 5 Director*.

Note: You enable the multi-host feature only if systems connected to the printer have z/OS installed (systems can be stand-alone systems or guests on VM/ESA). If the multi-host feature is enabled but one of the hosts is not installed with the correct software, PSF encounters printer errors and is not able to use the printer. For more information about the multi-host feature, see the *Introduction and Planning Guide* for your printer.

Exclusive access to a printer can also be obtained by enabling the multi-host feature and configuring the printer to come online at IPL. If multiple systems, which share a printer with the multi-host feature enabled and defined to come online at IPL, are restarted concurrently, the first system to successfully issue the printer ASSIGN gets access to the printer. When you are defining the printer by using a hardware configuration definition (HCD), the OFFLINE parameter can be optionally specified to configure the printer to come offline or online during an IPL. The default is to have the printer come online at IPL.

When the printer is configured in multi-host mode, it is important to follow proper operational procedures, such as draining PSF and varying the printer offline before re-IMLing the printer. Otherwise, the printer ASSIGN might be lost, the printer might be boxed, and PSF might issue an ABEND024 RC617. For more information about reconfiguration and recovery procedures for a printer ASSIGN, see the *z/OS MVS System Commands*.

Defining channel-attached printers

This section shows examples of how to define printers to your I/O configuration with basic definitions or definitions for sharing printers. The examples in this section are described by using MVSCP/IOCP statements; however, you can use HCD to define your I/O configuration.

Note: These examples are not meant to include all the steps that are required to define a printer. For more details about the steps that are required to define a printer with HCD, see *z/OS HCD User's Guide*.

For more flexibility when you are adding AFP1 printers to the I/O configuration, define the printer as AFP1, which lets any AFP1 channel-attached printer use that device address. When you are defining a 3800 printer for performance reasons, use a block-multiplex channel and use the data streaming protocol. For high-speed, APF1 parallel channel-attached printers, such as the InfoPrint 4000, specify a data transfer protocol of 4.5 MB data streaming (S4).

Basic definitions

Figure 3 on page 11 shows the parameters required for defining a 3800 Model 3 printer.

```
CHPID PATH=03,Type=BL
CNTLUNIT CUNUMBR=100,PATH=03,PROTOCL=S,UNIT=3800,UNITADD=80
IODEVICE CUNUMBR=100,UNIT=3800,MODEL=3,ADDRESS=380
```

Figure 3. Defining a 3800 Model 3 printer

Figure 4 on page 11 shows the parameters required for defining a host-connected, channel-attached 3820 printer.

```
CHPID PATH=04,TYPE=BY
CNTLUNIT CUNUMBR=200,PATH=04,PROTOCL=D,UNIT=3820,UNITADD=A0
IODEVICE CUNUMBR=200,UNIT=3820,ADDRESS=4A0
```

Figure 4. Defining a 3820 printer

Figure 5 on page 11 shows the parameters required for defining an AFP1 ESCON channel-attached printer.

```
CHPID PATH=05,TYPE=CNC
CNTLUNIT CUNUMBR=250,PATH=05,UNITADD=50,UNIT=AFP1
IODEVICE CUNUMBR=250,UNIT=AFP1,ADDRESS=550
```

Figure 5. Defining an AFP1 printer

Sharing parallel-channel printers with a two-channel switch

Figure 6 on page 12 shows an example of using a two-channel switch on a parallel channel-attached printer to share a printer between two hosts.
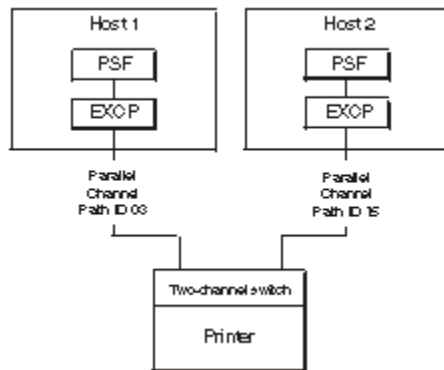
Figure 6. Example of a parallel-channel printer with two-channel switch configuration

Access to the printer is controlled by:

1. Doing these on the host that has access to the printer:

    a. Draining PSF

    b. Varying the printer offline

    c. Configuring the path offline

2. Doing these on the other host:

    a. Setting the two-channel switch

    b. Configuring the path online

    c. Varying the printer online

    d. Starting PSF

Figure 7 on page 12 shows the parameters required for defining a shared parallel channel-attached printer with two-channel switch configuration.

```
Host 1
    CHPID PATH=03,TYPE=BL
    CNTLUNIT CUNUMBR=300,PATH=03,PROTOCL=S4,UNIT=AFP1,UNITADD=80
    IODEVICE CUNUMBR=300,UNIT=AFP1,ADDRESS=380

Host 2
    CHPID PATH=15,TYPE=BL
    CNTLUNIT CUNUMBR=350,PATH=15,PROTOCL=S4,UNIT=AFP1,UNITADD=80
    IODEVICE CUNUMBR=350,UNIT=AFP1,ADDRESS=1580
```

Figure 7. Defining a parallel channel-attached printer with a two-channel switch

Sharing printers with an ESCON or FICON channel

Figure 8 on page 13 shows an example of using an ESCON or a FICON channel to share a printer between LPARs.
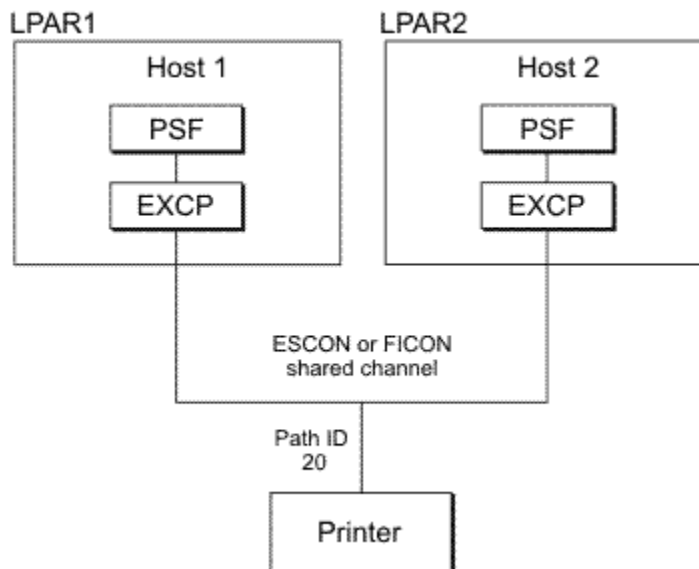
Figure 8. Example of a shared ESCON or FICON channel-attached printer

If the processor complex has ESCON Multiple Image Facility (EMIF) and is in LPAR mode, a path from each LPAR can be configured online concurrently to the printer. For those printers that support the multi-host feature, access to the printer is controlled by:

1. Doing these on the host that has access to the printer:

    a. Draining PSF

    b. Varying the printer offline

2. Doing these on the other host:

    a. Varying the printer online

    b. Starting PSF

For more information about LPAR mode and EMIF, see the *Processor Resource/Systems Manager Planning Guide* for your processor.

shows the parameters required for defining a shared ESCON or FICON channel-attached printer with multi-host support.

```
CHPID PATH=20,TYPE=CNC or FC,SHARED,PART=(LPAR1,LPAR2)
CNTLUNIT CUNUMBR=400,PATH=20,UNITADD=10,UNIT=AFP1
IODEVICE CUNUMBR=400,UNIT=AFP1,ADDRESS=2010
```

Figure 9. Defining an ESCON or FICON printer with multi-host support

For those printers that do not support the multi-host feature, only one path from each LPAR can be configured online to the printer. Access to the printer is controlled by:

1. Doing these on the host that has access to the printer:

    a. Draining PSF

    b. Varying the printer offline

    c. Configuring the path offline from the LPAR

2. Doing these on the other host:

    a. Configuring the path online to the LPAR

    b. Varying the printer online

    c. Starting PSF

All channel paths must be ESCON or FICON; they cannot be intermixed.

Figure 10 on page 14 shows the parameters required for defining a shared ESCON or FICON channel-attached printer without multi-host support.

```
CHPID PATH=20,TYPE=CNC or FC,PART=[(0),(LPAR1,LPAR2),REC]
CNTLUNIT CUNUMBR=400,PATH=20,UNITADD=10,UNIT=AFP1
IODEVICE CUNUMBR=400,UNIT=AFP1,ADDRESS=2010
```

Figure 10. Defining an ESCON or FICON printer without multi-host support

Sharing printers through a director

Figure 11 on page 14 shows an example of using a director between two systems to share a printer.
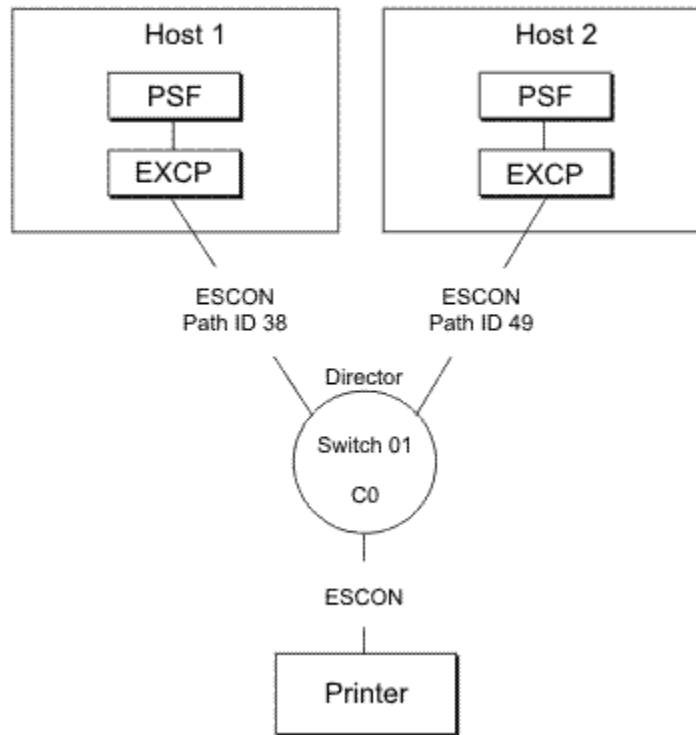
Figure 11. Example of a shared ESCON or FICON channel-attached printer that uses a director

For those printers that support the multi-host feature, a path from each host system can be configured online concurrently to the printer. Access to the printer is controlled by:

1. Doing these on the host that has access to the printer:

    a. Draining PSF

    b. Varying the printer offline

2. Doing these on the other host:

    a. Varying the printer online

    b. Starting PSF

All channel paths must be ESCON or FICON; they cannot be intermixed.

Figure 12 on page 15 shows the parameters that are required for defining an ESCON or FICON channel-attached printer with the multi-host support going though a director.

```
Host 1
    CHPID PATH=38,TYPE=CNC or FC,SWITCH=01
    CNTLUNIT CUNUMBR=500,PATH=38,LINK=C0,UNITADD=30,UNIT=AFP1
    IODEVICE CUNUMBR=500,UNIT=AFP1,ADDRESS=3830

Host 2
    CHPID PATH=49,TYPE=CNC or FC,SWITCH=01
    CNTLUNIT CUNUMBR=550,PATH=49,LINK=C0,UNITADD=30,UNIT=AFP1
    IODEVICE CUNUMBR=550,UNIT=AFP1,ADDRESS=4930
```

Figure 12. Defining an ESCON or FICON printer through a director with multi-host support

For those printers that do not support the multi-host feature, only one path from each host system can be configured online to the printer. Access to the printer is controlled by:

1. Doing these on the host that has access to the printer:

    a. Draining PSF

    b. Varying the printer offline

    c. Configuring the path offline

2. Doing these on the other host:

    a. Configuring the path online

    b. Varying the printer online

    c. Starting PSF

Access can also be controlled by blocking access to the printer at the director by blocking the port to the nonprinting system.

Chapter 4. Establishing network nodes for SNA-attached printers

This information describes sample network configurations for SNA-attached printers and provides specific parameters for setting up a printer network. Microfilm devices are not described in this information because they cannot be configured as SNA-attached printers.

For information about network configuration for Advanced Function Common Control Unit (AFCCU) printers (for example: 3130, InfoPrint 60, and InfoPrint 62), see *Attachment Configuration Handbook for Printers that Use the Advanced Function Common Control Unit*.

Do not use this information when configuring printers supported through distributed print function (DPF) or PSF direct programs. Instead, see the publications for those programs. The graphics show DPF and PSF direct together because they work the same way in concept even though they differ in details of configuration.

Overview of a network

To establish network nodes, you must define the communications control unit for the z/OS host. Then, you need to define the SNA-attached printers (or the devices to which they are attached) to Communication Server SNA Services, also referred to as Virtual Telecommunications Access Method (VTAM), as either local-attached or communication-attached.

Defining the communications control unit to z/OS

If you have not already done so, use a hardware configuration definition (HCD) to define the communications control unit for the z/OS host, such as the 3172 or the 3745 control unit, to z/OS.

For more information about using HCD, see one of the following publications:

- *z/OS HCD Planning*
- *z/OS HCD User's Guide*

Local-attached devices

Figure 13 on page 18 shows three sample configurations for local-attached devices. PSF sends data through VTAM to a printer, either directly or by way of a control unit. The control unit then sends the data to a printer, either directly or by way of a print server.
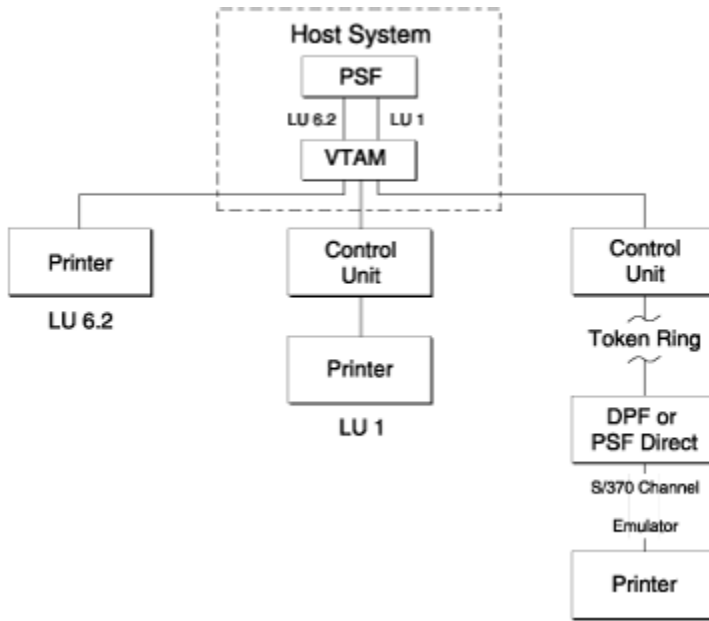
Figure 13. Sample PSF configurations for local-attached devices

Communication-attached devices

shows three sample configurations for communication-attached devices. VTAM sends data from PSF through a communication controller that runs the Network Control Program (NCP). Under NCP control, data is then sent to the printer directly, by way of a control unit or a print server.
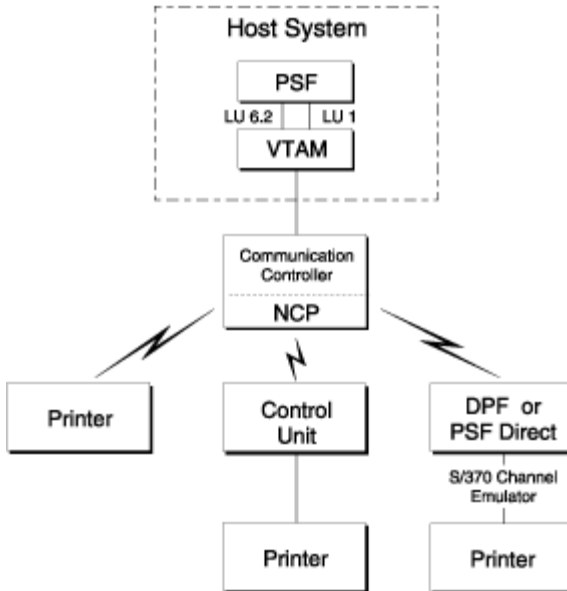
Figure 14. Sample PSF configurations for communication-attached devices

Communication can be established between the controller and the remote station by using either a switched line or a nonswitched line. (With a switched line, the connection is established by dialing; with a nonswitched line, no dialing is needed.)

shows four possible communication attachments with DPF and PSF direct:

- Synchronous Data Link Control (SDLC), attached through a communication controller
- Token-ring, attached through a communication controller token-ring subsystem
- Token-ring, attached through a remote communication controller token-ring subsystem

- Token-ring, attached through a remote control unit token-ring gateway
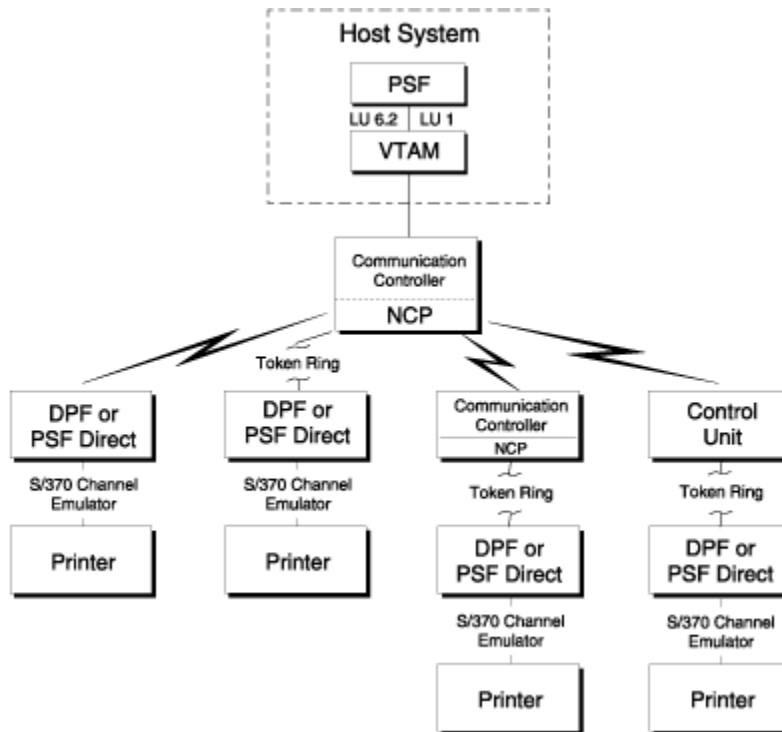
Figure 15. Sample PSF configurations for communication attachment to DPF or PSF direct

VTAM definitions for PSF network nodes

Before you use an SNA-attached printer, identify these:

- PSF application program nodes
- Logon-mode table entries for each printer model
- Printer-device nodes
- Communication lines for NCP, for communication-attached devices

For communication-attached devices, you must perform an NCP generation, which includes a definition of the communication lines to be connected to each printer. For more information, see "NCP generation" on page 30.

Note: The definition samples and explanations described here are not intended to be a complete description of the statements and operands. For more information, see your publications for VTAM.

SNA devices, attachment mode, and requirements

Table 2 on page 19 gives an overall picture of SNA devices, the mode of attachment, and the details about the requirements. SNA devices are attached by one of the modes in the table, each of which requires definitions for NCP.

Table 2. Requirements for SNA devices									
	Requirements								
						Define NCP major node			
Attachment Modes for SNA Devices	Define a local attached device	Define a PSF application node	Define the logon mode table entry	Define local major nodes	Define switched major nodes	"Switched line" on page 31	"Nonswitched line" on page 32	"Multipoint nonswitched line" on page 34	"Token-ring subsystem connection" on page 37
Local-Attached	X	X	X	X					

Table 2. Requirements for SNA devices (continued)									
	Requirements								
						Define NCP major node			
Attachment Modes for SNA Devices	**Define a local attached device**	**Define a PSF application node**	**Define the logon mode table entry**	**Define local major nodes**	**Define switched major nodes**	**"Switched line" on page 31**	**"Nonswitched line" on page 32**	**"Multipoint nonswitched line" on page 34**	**"Token-ring subsystem connection" on page 37**
Communication- Attached: Token Ring through Communication Controller Token-Ring Subsystem		X	X		X				X
Communication- Attached: Token Ring through Remote Communication Controller Token-Ring Subsystem		X	X		X		X		X
Communication- Attached: Token Ring through Remote Control Unit Token-Ring Gateway		X	X					X	
Communication- Attached: Switched Line		X	X		X	X			
Communication- Attached: Nonswitched Line		X	X				X		

With PSF, SNA devices are attached in one of the following ways, each of which requires network definitions for NCP:

Local-attached devices

- PSF application node. See "Defining a PSF application node" on page 22.
- Logon-mode table entry. See "Defining the logon-mode table entry" on page 22.
- Physical unit (PU) and logical unit (LU) statements in a local major node. See "Defining local major nodes" on page 25.

Make sure that the I/O device is included in the HCD. (The I/O device can be either a printer or a control unit.) If it is a 3820 channel-attached to the host, the 3820 must also be defined to the I/O configuration. See Figure 4 on page 11 and "Defining local major nodes" on page 25.

Communication-attached devices (switched line)

- PSF application node. See "Defining a PSF application node" on page 22.
- Logon-mode table entry. See "Defining the logon-mode table entry" on page 22.
- PU, PATH, and LU statements in a switched major node. See "Defining switched major nodes" on page 27.
- LINE and PU statements in a switched line GROUP under an NCP major node. An NCP generation is required. See "Switched line" on page 31.

Communication-attached devices (nonswitched line)

- PSF application node; see "Defining a PSF application node" on page 22.
- Logon-mode table entry; see "Defining the logon-mode table entry" on page 22.
- LINE, SERVICE (optional), PU, and LU statements in a nonswitched line GROUP under an NCP major node. An NCP generation is required. See "Nonswitched line" on page 32.

Communication-attached devices (token-ring)

- Through a communication controller token-ring subsystem:
  - PSF application node. See "Defining a PSF application node" on page 22.
  - Logon-mode table entry. See "Defining the logon-mode table entry" on page 22.
  - PU, PATH, and LU statements in a switched major node. See "Defining switched major nodes" on page 27.

- LINE, PU, and LU statements in a token-ring GROUP under an NCP major node. An NCP generation is required. See "Token-ring subsystem connection" on page 37.

- Through a remote communication controller token-ring subsystem:

  - PSF application node. See "Defining a PSF application node" on page 22.
  - Logon-mode table entry. See "Defining the logon-mode table entry" on page 22.
  - PU, PATH, and LU statements in a switched major node. See "Defining switched major nodes" on page 27.
  - LINE and PU statements in a nonswitched line GROUP under two NCP major nodes. An NCP generation is required. See "Nonswitched line" on page 32.
  - LINE, PU, and LU statements in a token-ring GROUP under an NCP major node. An NCP generation is required. See "Token-ring subsystem connection" on page 37.

- Through a remote control unit token-ring gateway:

  - PSF application node. See "Defining a PSF application node" on page 22.
  - Logon-mode table entry. See "Defining the logon-mode table entry" on page 22.
  - LINE, SERVICE, PU, and LU statements in a multipoint nonswitched line GROUP under an NCP major node. An NCP generation is required. See "Multipoint nonswitched line" on page 34.

VTAM syntax rules

VTAM definition statements and NCP macroinstructions must follow these syntax rules:

- The name field symbolically identifies the macroinstruction, definition statement, or minor node.
- If a symbolic name is specified in the field, the name must contain 1–8 characters in the following format:

  - Character 1: A–Z, or the characters @, #, or $
  - Characters 2–8: A–Z, 0–9, or the characters @, #, or $

- The name must begin in the first position of either the definition statement or the macroinstruction, and must be followed by one or more blanks.
- If the name is optional, it needs to be coded because VTAM uses it in operator messages that refer to the defined resource.
- A statement that does not fit on one line can be continued on the next line by using a continuation character (any nonblank character in column 72) and continuing the statement on column 16 of the next line. If you do not add a continuation character to a VTAM statement that is continued to a second line, you do not receive an error message, but processing stops at the end of the first line and is unable to continue with the second line of the statement. Therefore, the printer is incorrectly initialized.

These conventions apply to the definition-statement name fields described in this information.

VTAM statements and NCP macro instructions follow a convention called sift-down. By this convention, an operand can be coded on a higher-level node (for example, PU is a higher-level node than LU), so that the operand does not need to be recorded on each of the lower-level nodes for which the same value applies. Many operands do not affect the higher-level nodes, but are placed there to take advantage of sifting. A sifted value can be overridden by coding the same operand, with the new value, on a lower-level node. For more information about the sift-down effect, see *z/OS Communications Server: SNA Resource Definition Reference*.

The network configuration examples in this information do not follow the sift-down convention.

Note: Some statement parameters and their values differ, depending on which LU protocol is used between VTAM and the device. For example, for a 3820 printer, the parameters and values that are associated with LU type 6.2 are used; for a 3812 printer, the parameters and values that are associated with LU type 1 are used.

Defining a PSF application node

An APPL statement defines a PSF application program node, called a logical unit (LU), within the network domain of a VTAM supervisor services control point. This APPL statement also defines some of the communication capabilities of the application program. You need to also define an APPL statement for each PSF functional subsystem application (FSA) for each printer, and you must define at least one FSA per printer. The format in Figure 16 on page 22 is used.

```
name            VBUILD TYPE=APPL
PSFFSA1         APPL   AUTH=ACQ,EAS=1,SONSCIP=YES
```

Figure 16. Sample APPL statement

Note: Do not specify APPC=YES on the APPL statement.

APPL

Defines a PSF application program node and some of the communications capabilities of the application program.

Note: The name of the application program node is given (PSFFSA1 in this example) in the name field, unless the ACBNAME parameter is specified. If the ACBNAME parameter is specified, it defines the name of the application program node. For more information, see your publications for VTAM.

The application program node must also be specified in the **Applid** parameter in the Printer Inventory (see Applid) or the APPLID parameter in the PRINTDEV statement of the PSF startup procedure (see APPLID). The name of the APPL statement must be unique within the network.

AUTH

Allows PSF to use the SIMLOGON macroinstruction to initiate a session with the printer.

Do not specify AUTH=PPO on the APPL statement.

EAS

Specifies the number of concurrent sessions a PSF FSA can have with other logical units (LU–LU sessions). Specify EAS=1.

SONSCIP

Specifies that PSF is to receive UNBIND request/response units (RU) in its SCIP exit routine when PSF is acting as the primary end of a failing session.

Defining the logon-mode table entry

The logon-mode table must contain an entry (MODEENT statement) for each attachment protocol.

The logon-mode table can be one supplied by IBM, or it can be locally defined. If the table is locally defined, it must be identified by the MODETAB operand in the LU statement, and that table is searched first for the specified entry.

Local-attached entries (excluding InfoPrint Manager)

```
For an LU type 6.2 device

IBM3820C MODEENT LOGMODE=IBM3820C,FMPROF=X'13',TSPROF=X'07',
                 PRIPROT=X'B0',SECPROT=X'B0',
                 COMPROT=X'D0B1',RUSIZES=X'85C7',
                 PSERVIC=X'0602000000000000000002000',
                 PSNDPAC=X'03',SRCVPAC=X'03',
                 SSNDPAC=X'00'

For an LU type 1 device

IBM3812C MODEENT LOGMODE=IBM3812C,FMPROF=X'03',TSPROF=X'03',
                 PRIPROT=X'B1',SECPROT=X'B0',
                 COMPROT=X'7080',RUSIZES=X'85C7',
                 PSERVIC=X'0140000100000000001000000',
                 PSNDPAC=X'02',SRCVPAC=X'02',
                 SSNDPAC=X'00'
```

Figure 17. Sample logon-mode table entry for local-attached devices

The LU type 1 example can be used for all the LU type 1 printers that PSF supports.

Communication-attached entries (excluding token ring)

```
For an LU type 6.2 device

IBM3820  MODEENT LOGMODE=IBM3820,FMPROF=X'13',TSPROF=X'07',
                 PRIPROT=X'B0',SECPROT=X'B0',
                 COMPROT=X'D0B1',RUSIZES=X'8585',
                 PSERVIC=X'0602000000000000000002000',
                 PSNDPAC=X'03',SRCVPAC=X'03',
                 SSNDPAC=X'00'

For an LU type 1 device

IBM3812  MODEENT LOGMODE=IBM3812,FMPROF=X'03',TSPROF=X'03',
                 PRIPROT=X'B1',SECPROT=X'B0',
                 COMPROT=X'7080',RUSIZES=X'8585',
                 PSERVIC=X'0140000100000000001000000',
                 PSNDPAC=X'03',SRCVPAC=X'03',
                 SSNDPAC=X'00'
```

Figure 18. Sample logon-mode table entry for communication-attached devices

For VTAM definitions for DPF, see the program publications.

```
For a token-ring device

IBM3820R  MODEENT LOGMODE=IBM3820R,FMPROF=X'13',TSPROF=X'07',
                  PRIPROT=X'B0',SECPROT=X'B0',
                  COMPROT=X'D0B1',RUSIZES=X'mnmn',
                  PSERVIC=X'0602000000000000000002000',
                  PSNDPAC=X'xx',SRCVPAC=X'yy',
                  SSNDPAC=X'00'
```

Figure 19. Sample logon-mode table entry for token-ring devices

Values for RUSIZES, PSNDPAC, and SRCVPAC depend upon the configuration. Recommended values are listed in the following descriptions of these parameters.

The parameters for the sample MODEENT statements follow. (For information not provided in the following descriptions, see your publications for VTAM.)

MODEENT
Creates an entry in the logon-mode table. The statement name is optional and has no function in the specification of the logon-mode table.

COMPROT=*hex value*
The value for an LU type 6.2 device is X'D0B1'. The value for an LU type 1 device is X'7080'.

FMPROF=*hex value*
The value for an LU type 6.2 device is X'13'. The value for an LU type 1 device is X'03'.

LOGMODE
Specifies the logon-mode name to be used as a key for the session operands in this table entry. If the same name appears twice in the table, the first occurrence is used. Alternatively, the name identifies this entry in the logon-mode table, with the DLOGMOD operand of the LU statement for a printer, with the **Logmode** parameter in the Printer Inventory (see Logmode) or the LOGMODE parameter in the PRINTDEV statement of the PSF startup procedure (see LOGMODE). If LOGMODE is omitted, eight blanks are used.

PRIPROT=*hex value*
The value for an LU type 6.2 device is X'B0'. The value for an LU type 1 device is X'B1'.

PSERVIC=*hex value*
The value for an LU type 6.2 device is X'0602 0000 0000 0000 0000 2000'. The value for an LU type 1 device is X'0140 0001 0000 0000 0100 0000'.

PSNDPAC=*hex value*
Suggested values are:

- For an LU type 6.2 device—X'03'
- For an LU type 1.0 device—X'nn'

Note: Pacing is an important performance parameter, for it determines how many RUs the host can send before a pacing response is required. This value is device-dependent. For more information about setting the pacing value, see the publications for your device and for VTAM.

RUSIZES=X'mnmn'
Specifies the maximum length (request units, RUs) in bytes that can be sent. The two leftmost hexadecimal digits apply to data sent by the secondary LU (printer). The two rightmost hexadecimal digits apply to data sent by the primary LU (PSF). In each pair, the first digit is the mantissa (*m*), and the second digit is the exponent (*n*) in the formula:

```
LENGTH = m x 2ⁿ
```

Thus, RUSIZES=X'8585' specifies $length = 256$ (8×2^5) for both logical units.

Recommended values are:

- For local-attached devices: X'85C7'
- For communication-attached devices: X'8585'
- For communication-attached devices: 3935
  - Communication-attached through a 3745 with SDLC modem: X'86F8'
  - Communication-attached through a 3745 with token-ring subsystem: X'87F8'
- For 3935 printer:
  - Communication-attached through 3745 with SDLC modem: X'86F8'
  - Communication-attached through 3745 with token-ring subsystem: X'87F8'

SECPROT=*hex value*
The value is X'B0'.

SRCVPAC=*hex value*
Suggested values are:

- For an LU type 6.2 device: X'03'
- For an LU type 1 device: X'*nn*'
  - Local-attached through a control unit token-ring Gateway: X'10'
  - Communication-attached through a remote control unit token-ring Gateway: X'10'

- Communication-attached through a local or remote communication controller token-ring Subsystem: X'10'
- Communication-attached through an SDLC link: X'03'

Note: 3820 printers default to a pacing of X'03' if any larger number is specified.

SSNDPAC=*hex value*
The value for all devices is X'00'.

TSPROF=*hex value*
The value for an LU type 6.2 device is X'07'. The value for an LU type 1 device is X'03'.

Defining local major nodes

A local-attached configuration is defined to VTAM as using a local major node. For example, for a printer that is connected to an LU type 1 control unit, the connection between VTAM and the control unit is defined with a PU statement. Each printer that is attached to the control unit is defined with an LU statement. Figure 20 on page 25 shows an example.
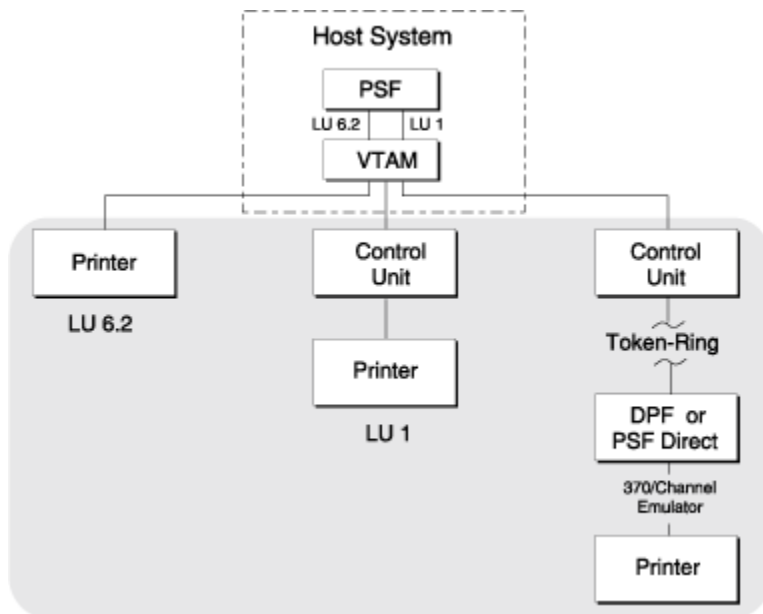
Figure 20. Example showing three local major nodes

Figure 21 on page 25 shows an example of VTAM definitions for local major nodes.

```
For an LU type 6.2 device

CHAN3TST  VBUILD  TYPE=LOCAL
PUYALE8   PU      CUADDR=nnn,MAXBFRU=13
LUYALE8   LU      LOCADDR=1,DLOGMOD=IBM3820C,MODETAB=MODETAB1

For an LU type 1 device

LOC3812   VBUILD  TYPE=LOCAL
PU13812   PU      CUADDR=nnn,MAXBFRU=2
LU13812   LU      LOCADDR=8,MODETAB=MODETAB1,DLOGMOD=IBM3812C,ISTATUS=ACTIVE
```

Figure 21. Sample local major node definitions for different LU protocols

VBUILD
Describes the node.

TYPE=LOCAL
Specifies a local major node.

PU
Describes the physical unit. For LU type 1 printers, the PU is the control unit.

CUADDR
Specifies the hexadecimal channel-device name to be used when the PU is activated. The value must match the channel-device name specified in the device configuration for the control unit or printer connected to the host.

MAXBFRU
Specifies, as a decimal integer, the number of buffer units (elements of the IOBUF buffer pool) that are used for data by VTAM. The number of buffers depends on the maximum RU size and the size of the IOBUF elements. For information about customizing this value, see your publications for VTAM.

LU
Describes the logical unit (LU). The name that is specified on the LU statement must also be specified on the **LU name** parameter in the Printer Inventory (see LU name) or the LUNAME parameter in the PRINTDEV statement of the PSF startup procedure (see LUNAME).

DLOGMOD
Specifies the name of the VTAM logon-mode table entry if one is not otherwise provided. The name must match the LOGMODE operand of the MODEENT statement for the entry. For more information, see "Defining the logon-mode table entry" on page 22.

This parameter is required when **Management mode=Dial in** is specified in the Printer Inventory (see Management mode) or MGMTMODE=DIALIN is specified in the PRINTDEV statement of the PSF startup procedure (see MGMTMODE).

If **Logmode** is specified in the Printer Inventory (see Logmode) or LOGMODE is specified in the PRINTDEV statement (see LOGMODE), it overrides the name that is specified here when either **Management mode=Immediate** or **Management mode=Output available** is specified in the Printer Inventory, or MGMTMODE=IMMED or MGMTMODE=OUTAVAIL is specified in the PRINTDEV statement.

ISTATUS={ACTIVE|INACTIVE}
Specifies whether this logical unit is to be activated when the major node is first activated.

LOCADDR
Specifies the LU address at the PU. For 3820 or a 3935 printer, LOCADDR=1. For an LU1 printer, use the address of the printer line that is coming from the control unit. For example, for a 3174 control unit that is using LU1, this value equals the port number of the printer plus 2.

A LOCADDR of X'00' defines the LU as an independent LU, and all sessions use the same X'00' LU address.

A LOCADDR of X'01' through X'FF' defines the LU as a dependent LU, and each session must have a unique LU address.

LOGAPPL
Specifies the name of the application program and must match the name that is specified in the **Applid** parameter in the Printer Inventory (see Applid) or the APPLID parameter in the PRINTDEV statement (see APPLID). This parameter is required only when **Management mode=Dial in** is specified in the Printer Inventory (see Management mode) or MGMTMODE=DIALIN is specified in the PRINTDEV statement (see MGMTMODE). For more information, see the note in "APPL" on page 22.

Note: When LOGAPPL is specified, the host application program node must be active before the device LU node is activated. Also, if **Management mode=Dial in** is specified in the Printer Inventory or MGMTMODE=DIALIN is specified in the PRINTDEV statement, printers cannot be shared.

MODETAB
Specifies the name of the logon-mode table if it is defined locally. If it is not specified, the logon-mode table that is supplied by IBM is used.

PACING

Determines how VTAM paces the flow of data from the boundary node, which does pacing for the physical unit to the secondary logical unit (SLU). Zero means no pacing.

VPACING

Determines how VTAM paces the flow of data from the host system to the boundary node, which does pacing for the physical unit. Zero means no pacing.

Defining switched major nodes

Figure 22 on page 27 shows major nodes that are switched to link VTAM with a communication controller.
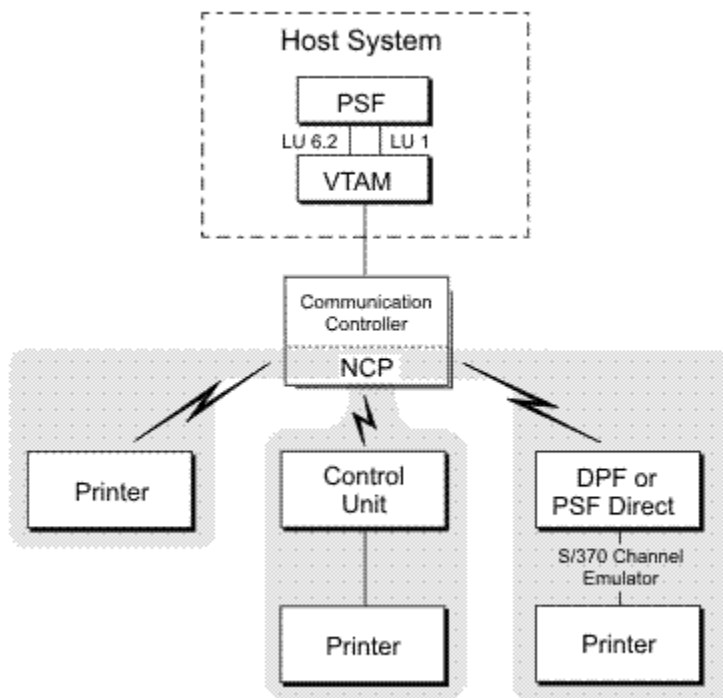
Figure 22. Sample configuration of switched major nodes

Figure 23 on page 27 shows samples that define switched major nodes for several different protocols.

```
For an LU type 6.2 device

SWTAUTIO  VBUILD  TYPE=SWNET,MAXGRP=5,MAXNO=4
PATINOUT  PU      ADDR=C1,IDBLK=nnn,IDNUM=0F0F0,MAXPATH=1,
                  DISCNT=YES,PUTYPE=2
          PATH    DIALNO=95556928,GID=4,GRPNM=SNAS5
LUPRT1    LU      LOCADDR=1,LOGAPPL=PSFFSA1,DLOGMOD=zzzzzzzz,
                  MODETAB=MODETAB1

For an LU type 1 device

SWT3812   VBUILD  TYPE=SWNET,MAXGRP=1,MAXNO=2
PU2810    PU      ADDR=C1,IDBLK=017,IDNUM=0E0E0,MAXPATH=1,PUTYPE=2
          PATH    DIALNO=9245566,GID=1,GRPNM=G28SW10,DISCNT=NO
LU381210  LU      LOCADDR=8,DLOGMOD=IBM3812,MODETAB=MODETAB1
```

Figure 23. Sample definitions of switched major nodes

An NCP generation is required. To do the NCP generation, see "Defining network control program (NCP) major nodes" on page 30.

The samples consist of the following statements:

VBUILD
Defines the node.

> **MAXGRP**
> Specifies the number of unique path groups (GROUP names) that are specified in the GRPNM operand of each PATH statement in the switched major node.

> **MAXNO**
> Specifies how many unique telephone numbers are specified in the DIALNO operand of each PATH statement in the switched major node.

> **TYPE**
> Specifies that the VBUILD statement defines a switched major node.

PU
Defines the physical unit:

> **ADDR**
> Specifies the 8-bit SDLC station address for the physical unit.
>
> **Note:** This value must match the station address specified for the 3820 microcode diskette configuration. For an LU type 1 printer, this address must match the address of the control unit.

> **DISCNT**
> Specifies whether VTAM physically disconnects the physical unit when the logical unit ends its session with PSF. DISCNT=YES prevents excess line charges; with DISCNT=NO, the line remains connected after the session is complete. For more information, see your publications for VTAM.

> **IDBLK**
> Specifies the device identification number. For a 3935 printer, this value must be X'071'. For most other LU type 6.2 devices, this value is X'042'.

> **IDNUM**
> Specifies the 20-bit station ID number. This number must match the configuration value of the PU.
>
> **Note:** VTAM uses the value of IDBLK and the value of IDNUM when it builds a 48-bit station ID that is used in the XID exchange during the dial procedure. This station ID must be unique for each station in the network. Therefore, each station must have a unique IDNUM value. These are the definitions for bits in the station ID:

> > **Bits 0–3:**
> > Reserved

> > **Bits 4–7:**
> > Contain the value of PUTYPE

> > **Bits 8–15:**
> > Contain X'00'

> > **Bits 16–27:**
> > Contain the value of IDBLK

> > **Bits 28–47:**
> > Contain the value of IDNUM

> **MAXDATA**
> Specifies the maximum number of data bytes that the physical unit can receive in one data transfer. MAXDATA equals the maximum RUSIZE for the PU plus the header length, which is 9 bytes. The maximum RU size is specified by the logon-mode table RUSIZES parameter. If RUSIZES=X'8787', for example, MAXDATA=1033.
>
> For non-3820 LU 6.2 devices and for the 3935 printer, MAXDATA=3849 is recommended.

> **MAXPATH**
> Specifies the number of dial-out paths to the physical unit.

> **PUTYPE**
> Specifies the physical-unit type of the station represented by this PU statement. PUTYPE=2 must be specified and is the default.

PATH

Defines a dial-out path from a communication controller to a physical unit in a switched major node:

DIALNO

Specifies the telephone number that connects with the physical unit—for example, the telephone number of a 3274 control unit.

GID

Identifies a group of paths across all physical units in the switched major node.

GRPNM

Specifies the symbolic name of a GROUP statement in an NCP major node. That GROUP statement defines a logical group of SDLC switched lines or token-ring switched lines.

PID

Identifies the path that is being defined.

LU

Defines the logical unit. The name that is specified in the LU statement must match the name that is specified for the **LU name** parameter in the Printer Inventory (see LU name) or the LUNAME parameter in the PRINTDEV statement of the PSF startup procedure (see LUNAME).

DLOGMOD

Specifies the name of the VTAM logon-mode table entry if one is not otherwise provided. The name must match the LOGMODE operand of the MODEENT statement for the entry. For more information, see "Defining the logon-mode table entry" on page 22.

This parameter is required when **Management mode=Dial in** is specified in the Printer Inventory (see Management mode) or MGMTMODE=DIALIN is specified in the PRINTDEV statement of the PSF startup procedure (see MGMTMODE).

If **Logmode** is specified in the Printer Inventory (see Logmode) or LOGMODE is specified in the PRINTDEV statement (see LOGMODE), it overrides the name that is specified here when either **Management mode=Immediate** or **Management mode=Output available** is specified in the Printer Inventory, or MGMTMODE=IMMED or MGMTMODE=OUTAVAIL is specified in the PRINTDEV statement.

Note: For most LU type 6.2 devices, DLOGMOD=IBM3820.

LOCADDR

Specifies the LU address at the PU. For an LU 6.2 device, LOCADDR=1.

A LOCADDR of X'01' through X'FF' defines the LU as a dependent LU, and each session must have a unique LU address.

LOGAPPL

Specifies the name of the application program and must match the name that is specified in the **Applid** parameter in the Printer Inventory (see Applid) or the APPLID parameter in the PRINTDEV statement (see APPLID). This parameter is required only when **Management mode=Dial in** is specified in the Printer Inventory (see Management mode) or MGMTMODE=DIALIN is specified in the PRINTDEV statement (see MGMTMODE). For more information, see the note in "APPL" on page 22.

Note: When LOGAPPL is specified, the host application program node must be active before the device LU node is activated. Also, if **Management mode=Dial in** is specified in the Printer Inventory or MGMTMODE=DIALIN is specified in the PRINTDEV statement, printers cannot be shared.

MODETAB

Specifies the name of the logon-mode table if it is defined locally. If it is not specified, the logon-mode table that is supplied by IBM is used.

SSCPFM

Determines whether the logical unit can support field-formatted functional subsystem (FSS) RUs or character-coded RUs over its system services control point (SSCP)/LU session.

VPACING

   Specifies how VTAM paces the flow of data from the host system to the boundary node to which the logical unit is connected. A value of 0 indicates no pacing.

Defining network control program (NCP) major nodes

An NCP major node defines an NCP, the set of lines attached to it, and the physical units and logical units associated with the lines. Lines of the same type are grouped into sets by the GROUP macroinstruction. Each line in a set is defined by a LINE macroinstruction, followed by PU macroinstructions that define the physical units that are attached to the NCP by this line and LU macroinstructions that define the logical units associated with the physical units. Each of these macroinstructions, including the GROUP macroinstruction, defines a minor node that represents a network resource.

All the macroinstructions that are listed in "NCP generation" on page 30 must also be supplied to VTAM as part of the NCP major node definition. For a diagram of a sample definition, see Figure 24 on page 30. For an overview of the VTAM and NCP definitions that PSF requires see Table 2 on page 19.
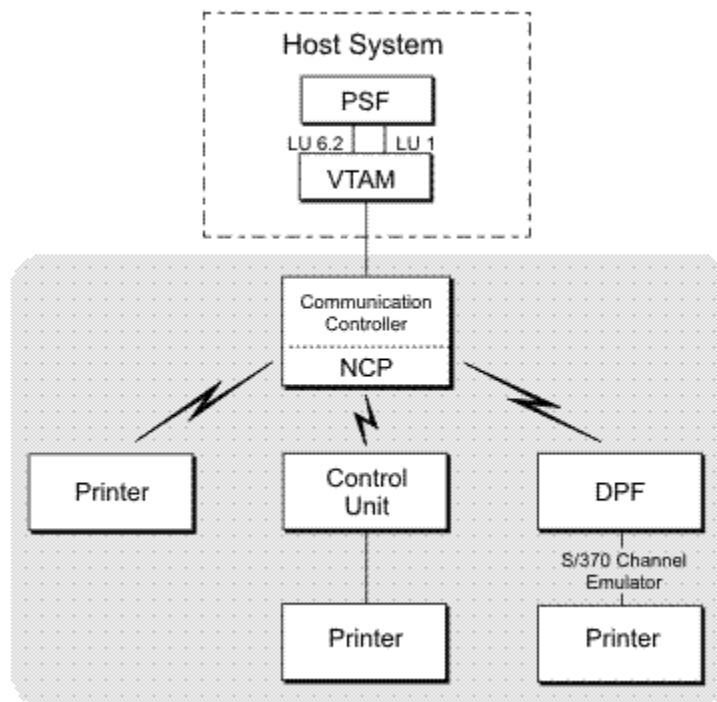
Figure 24. Sample NCP major node

NCP generation

The macroinstructions that compile the NCP have two purposes:

- They generate an NCP that can be loaded into a communication controller and describe to that NCP the lines, line groups, link stations, and terminals it is to control.
- They describe to VTAM the same NCP major node and the resources that the NCP is to control.

Most of the operands in NCP-generation macroinstructions are used only for generating modules of the NCP. However, some of the operands are directed either to both NCP and VTAM, or only to VTAM. The NCP-generation macroinstructions are statements for VTAM network definition, and true macroinstructions for NCP generation.

The NCP-only macroinstructions and the operands that are used by both NCP and VTAM are verified during NCP generation but not during VTAM network definition. Therefore, the NCP needs to be generated before the generation deck is filed in the VTAM definition library as a VTAM major node.

The macroinstructions to be added to the NCP generation for the printer depend on the communication lines. VTAM and NCP support many network configurations; these are representative of communication lines used with PSF:

Switched line
    A telecommunication line in which the connection is established by dialing.

Nonswitched line
    A telecommunication line in which the connection is permanently established and does not require dialing.

Multipoint line (nonswitched)
    A line that connects more than two devices.

Token-ring subsystem connection
    Cabling that links devices in an IBM token-ring local area network (LAN).

Note: The samples and explanations described here are not intended to be a complete description of the macroinstructions and operands. For more information, see *NCP: Resource Definition Guide* and *NCP: Resource Definition Reference*.

Switched line

Figure 25 on page 31 shows an example of NCP macroinstructions for switched lines.

```
For an LU type 6.2 device

SNAS1     GROUP   DIAL=YES,LNCTL=SDLC,TYPE=NCP,ISTATUS=ACTIVE,ANSWER=ON,REPLYTO=1
SNAL022   LINE    CALL=OUT,ADDRESS=(035),SPEED=4800,CLOCKNG=EXT,AUTO=03F
PATINOUT  PU      PUTYPE=2,MAXLU=1


For an LU type 1 device

G28SW10   GROUP   DIAL=YES,LNCTL=SDLC,REPLYTO=1,ANSWER=ON,TYPE=NCP
LN2810    LINE    ADDRESS=(010,HALF),CLOCKNG=EXT,SPEED=4800,CALL=INOUT
PU2810    PU      PUTYPE=(2),MAXLU=10
```

Figure 25. Sample NCP macroinstructions for switched lines

These samples contain the following instructions:

GROUP
    Describes a group of communication lines:

    ANSWER
        Specifies whether the physical units can dial into VTAM when the line is active.

    DIAL
        Specifies whether the lines in the group require switched-line control procedures.

    ISTATUS
        Specifies whether the lines in the group are to be activated after the first activation of the major node.

    LNCTL
        Specifies the line control for the lines in the group.

    REPLYTO
        Specifies the reply timeout value in seconds for the lines in the group.

    TYPE
        Specifies, for the lines in the group, that all lines operate in network control mode.

LINE
    Describes one communication line:

    ADDRESS
        Specifies the relative line-number-interface address for the controller.

AUTO
> Specifies that the auto-call facility is installed and provides the line number interface address for the controller.

CALL
> When CALL=OUT or CALL=INOUT is specified, VTAM prompts the VTAM operator to dial the telephone at the correct time. When CALL=IN and MGMTMODE=DIALIN are specified, PSF waits for the printer operator to dial.

CLOCKNG
> Specifies whether the modem or the communication controller for the line is to provide the clocking.

SPEED
> Specifies the data rate for this line in bits per second.

PU
Defines the physical unit:

MAXLU
> Specifies the maximum number of logical units that can be associated with this physical unit.

PUTYPE
> Specifies the physical-unit type of the station represented by this PU macroinstruction. PUTYPE=2 must be specified.

Nonswitched line

Figure 26 on page 32 shows sample NCP macro instructions for nonswitched lines.

```
For an LU type 6.2 device

SNAL1     GROUP   DIAL=NO,LNCTL=SDLC,TYPE=NCP,REPLYTO=1,ISTATUS=ACTIVE
SNAL022   LINE    ADDRESS=(022),SPEED=9600,CLOCKNG=EXT,RETRIES=(5,1,2)
          SERVICE ORDER=(CTL01)

CTL01     PU      ADDR=C1,ANS=STOP,MAXDATA=nnnn,
                  MAXOUT=7,PUTYPE=2
LUPRT1    LU      LOCADDR=1,LOGAPPL=PSFFSA1,DLOGMOD=zzzzzzzz,MODETAB=MODETAB1

For an LU type 1 device

G28PTPT   GROUP   DIAL=NO,LNCTL=SDLC,TYPE=NCP,ISTATUS=ACTIVE,REPLYTO=1
LN2804    LINE    ADDRESS=(004,HALF),CLOCKNG=EXT,RETRIES=(5,1,2),SPEED=9600
          SERVICE ORDER=PU2804

PU2804    PU      ADDR=C1,ANS=STOP,MAXDATA=265,MAXOUT=7,
                  PASSLIM=8,MAXLU=10,PUTYPE=2
LU38124   LU      DLOGMOD=IBM3812,LOCADDR=8,MODETAB=MODETAB1
```

Figure 26. Sample NCP macro instructions for nonswitched lines

This example consists of the following instructions:

GROUP
Describes a group of communication lines:

ACTIVTO
> Specifies the timeout value that the secondary NCP waits for communication from the primary NCP before it initiates shutdown.

DIAL
> Specifies whether the lines in the group require switched-line control procedures.

ISTATUS
> Specifies whether the group is to be active after the first activation of the major node.

LNCTL
> Specifies the line control for the lines in this group.

MODE
>Specifies the mode of operation of a line group associated with a selection table entry defined by an SDLCST operand in a LINE definition statement.

REPLYTO
>Specifies the reply timeout value in seconds. One second is the default for SDLC lines.

TYPE
>Specifies, for the line group, that all lines are to operate in network control mode.

LINE
>Describes one communication line:

>**ADDRESS**
>>Specifies the relative line number of the line interface address for the controller.

>**CLOCKNG**
>>Specifies whether the modem or the communication controller is to provide clocking.

>**DUPLEX**
>>Specifies whether the communication line and modem have a half-duplex or full-duplex capability.

>**IPL**
>>Specifies whether the NCP being generated can be loaded and dumped over this link.

>**PAUSE**
>>Specifies a cycle delay according to the link.

>**RETRIES**
>>Specifies the number of attempts to recover from errors that occurred during transmission over the link.

>**SDLCST**
>>Specifies in order which SDLC selection-table entry the NCP uses when it is operating in a primary state and which entry the NCP uses when it is operating in a secondary state.

>**SPEED**
>>Specifies the data rate for this line in bits per second.

SERVICE
>Generates a service-order table for a nonswitched SDLC link:

>**ORDER**
>>Specifies the order in which the stations on the line are serviced.

PU
>Defines the physical unit:

>**ADDR**
>>Specifies the 8-bit SDLC station address for the physical unit.

>>**Note:** This value must match the station address specified for the device configuration. For an LU type 1 printer, this address must match the address for the control unit.

>**ANS**
>>Specifies whether the operation of the station is to stop or continue if the NCP enters automatic network shutdown.

>**MAXDATA**
>>Specifies the maximum number of data bytes that the physical unit can receive in one data transfer. MAXDATA equals the maximum RU size for the PU plus the header length, which is 9 bytes. The maximum RU size is specified by the logon-mode table RUSIZES parameter. For example, when RUSIZES=X'8585', MAXDATA=265. For most LU type 6.2 and LU type 1 devices, MAXDATA=265.

>>For non-3820 LU6.2 devices and the 3935 printer, MAXDATA=3849 is recommended.

>**MAXOUT**
>>Specifies the maximum number of path-information units that the NCP sends to the physical unit before it requests a response. The recommended range for this value is 3 - 7.

PASSLIM
Specifies the maximum number of consecutive path information units (PIUs) or PIU segments the NCP sends at one time to the physical unit.

PUTYPE
Specifies the physical-unit type of the SDLC station represented by this PU macroinstruction.

TGN
Specifies the transmission group number of the subarea link associated with this physical unit.

LU

Defines the logical unit. The name that is specified in the LU statement must match the name that is specified for the **LU name** parameter in the Printer Inventory (see LU name) or the LUNAME parameter for the printer FSA in the PRINTDEV statement of the PSF startup procedure (see LUNAME).

DLOGMOD
Specifies the name of the VTAM logon-mode table entry if one is not otherwise provided. The name must match the LOGMODE operand of the MODEENT statement for the entry. For more information, see "Defining the logon-mode table entry" on page 22.

This parameter is required when **Management mode=Dial in** is specified in the Printer Inventory (see Management mode) or MGMTMODE=DIALIN is specified in the PRINTDEV statement of the PSF startup procedure (see MGMTMODE).

If **Logmode** is specified in the Printer Inventory (see Logmode) or LOGMODE is specified in the PRINTDEV statement (see LOGMODE), it overrides the name that is specified here when either **Management mode=Immediate** or **Management mode=Output available** is specified in the Printer Inventory, or MGMTMODE=IMMED or MGMTMODE=OUTAVAIL is specified in the PRINTDEV statement.

Note: For most LU type 6.2 devices, DLOGMOD=IBM3820.

LOCADDR
Specifies the LU address at the PU. For LU 6.2 devices, LOCADDR=1.

A LOCADDR of X'01' through X'FF' defines the LU as a dependent LU, and each session must have a unique LU address.

LOGAPPL
Specifies the name of the application program and must match the name that is specified in the **Applid** parameter in the Printer Inventory (see Applid) or the APPLID parameter in the PRINTDEV statement (see APPLID). This parameter is required only when **Management mode=Dial in** is specified in the Printer Inventory (see Management mode) or MGMTMODE=DIALIN is specified in the PRINTDEV statement (see MGMTMODE). For more information, see the note in "APPL" on page 22.

Note: When LOGAPPL is specified, the host application program node must be active before the device LU node is activated. Also, if **Management mode=Dial in** is specified in the Printer Inventory or MGMTMODE=DIALIN is specified in the PRINTDEV statement, printers cannot be shared.

MODETAB
Specifies the name of the logon-mode table if it is defined locally. If MODETAB is not specified, the logon-mode table that is supplied by IBM is used.

Multipoint nonswitched line

Figure 27 on page 35 shows an example of macroinstructions for generating NCP for multipoint lines.

```
For an LU type 6.2 device

PRTRPPM    LINE     ADDRESS=026,SPEED=9600,CLOCKNG=EXT,RETRIES=(5,1,2)
           SERVICE  ORDER=(PRTRPU3,PRTRPU4)

PRTRPU3    PU       ADDR=C3,ANS=STOP,MAXDATA=nnnn,MAXLU=1,MAXOUT=7,PUTYPE=2
PRTRLU3    LU       LOCADDR=1,LOGAPPL=PSFFSA3,DLOGMOD=zzzzzzzz,MODETAB=MODETAB1
PRTRPU4    PU       ADDR=C4,ANS=STOP,MAXDATA=nnnn,MAXLU=1,MAXOUT=7,PUTYPE=2
PRTRLU4    LU       LOCADDR=1,LOGAPPL=PSFFSA4,DLOGMOD=zzzzzzzz,MODETAB=MODETAB1

For an LU type 1.0 device

LN2804     LINE     ADDRESS=(004,HALF),CLOCKNG=EXT,RETRIES=(5,1,2),PACING=3,
                    VPACING=3,SPEED=9600
           SERVICE  ORDER=(PU2804A,PU2804B)


PU2804A    PU       ADDR=C1,ANS=STOP,IRETRY=YES,MAXDATA=265,MAXOUT=7,
                    PASSLIM=8,MAXLU=10,PUTYPE=2
LU32704A   LU       LOCADDR=2,MODETAB=MODETAB1,USSTAB=USSTAB1,
                    DLOGMOD=S32782,ISTATUS=ACTIVE
LU32704B   LU       LOCADDR=3,MODETAB=MODETAB1,USSTAB=USSTAB1,
                    DLOGMOD=S32782,ISTATUS=ACTIVE
LU38124A   LU       LOCADDR=8,MODETAB=MODETAB1,DLOGMOD=IBM3812
PU2804B    PU       ADDR=C2,ANS=STOP,IRETRY=NO,MAXDATA=265,MAXOUT=3,
                    MAXLU=10,PUTYPE=2
LU38204B   LU       LOCADDR=01,MODETAB=MODETAB1,DLOGMOD=IBM3820,
                    LOGAPPL=PSFFSA1
```

Figure 27. Sample NCP macroinstructions for multipoint lines

This example consists of the following instructions:

LINE
Describes one communication line:

ADDRESS
Specifies the relative line number of the line interface address for the controller.

CLOCKNG
Specifies whether the modem or the communication controller is to provide clocking.

DUPLEX
Specifies whether the communication line and the modem have a half-duplex or full-duplex capability.

RETRIES
Specifies how many times PSF is to attempt to recover from errors that occurred during transmission over the link.

SPEED
Specifies the data rate for this line in bits per second.

SERVICE
Generates a service-order table for a nonswitched SDLC link by using the following operand:

ORDER
Specifies the order in which the stations on the line are serviced.

PU
Defines the physical unit:

ADDR
Specifies the 8-bit SDLC station address for the physical unit.

Note: This value and the station address that is specified for the device must match. For an LU type 1 printer, this address must match the address for the control unit.

ANS
Specifies whether the operation of the station is to stop or continue if the NCP enters automatic network shutdown.

IRETRY

Specifies whether the NCP again polls a secondary station before it advances to the next station in the service-order table.

MAXDATA

Specifies the maximum number of data bytes that the physical unit can receive in one data transfer. MAXDATA equals the maximum RU size for the PU plus the header length, which is 9 bytes. The maximum RU size is specified by the RUSIZES parameter in the logon-mode table. When RUSIZES=X'8585', for example, MAXDATA=265.

For non-3820 LU 6.2 devices, MAXDATA=3849 is recommended.

MAXLU

Specifies the maximum number of logical units that can be associated with this physical unit.

MAXOUT

Specifies the maximum number of path information units that the NCP is to send to the physical unit before it requests a response. The recommended range for this value is 3 - 7.

PASSLIM

Specifies the maximum number of consecutive path units (PIUs) or PIU segments the NCP is to send to the physical unit at one time.

PUTYPE

Specifies the physical unit type of the SDLC station represented by this PU macroinstruction. PUTYPE=2 must be specified.

LU

Defines the logical unit. The name that is specified in the LU statement must match the name that is specified for the **LU name** parameter in the Printer Inventory (see LU name) or the LUNAME parameter for the printer FSA in the PRINTDEV statement of the PSF startup procedure (see LUNAME). LU defines the logical unit:

DLOGMOD

Specifies the name of the VTAM logon-mode table entry if one is not otherwise provided. The name must match the LOGMODE operand of the MODEENT statement for the entry. For more information, see "Defining the logon-mode table entry" on page 22.

If **Logmode** is specified in the Printer Inventory (see Logmode) or LOGMODE is specified in the PRINTDEV statement (see LOGMODE), it overrides the name that is specified here when either **Management mode=Immediate** or **Management mode=Output available** is specified in the Printer Inventory, or MGMTMODE=IMMED or MGMTMODE=OUTAVAIL is specified in the PRINTDEV statement.

Note: For most LU type 6.2 devices, DLOGMOD=IBM3820.

ISTATUS

Specifies whether the logical unit is to be active after the first activation of the major node.

LOCADDR

Specifies the LU address at the PU. For LU 6.2 devices, LOCADDR=1. For an LU type 1 printer, use the address of the printer line that comes from the control unit.

A LOCADDR of X'01' through X'FF' defines the LU as a dependent LU, and each session must have a unique LU address.

LOGAPPL

Specifies the name of the application program and must match the name that is specified in the **Applid** parameter in the Printer Inventory (see Applid) or the APPLID parameter in the PRINTDEV statement (see APPLID). This parameter is required only when **Management mode=Dial in** is specified in the Printer Inventory (see Management mode) or MGMTMODE=DIALIN is specified in the PRINTDEV statement (see MGMTMODE). For more information, see the note in "APPL" on page 22.

Note: When LOGAPPL is specified, the host application program node must be active before the device LU node is activated. Also, if **Management mode=Dial in** is specified in the Printer

Inventory or MGMTMODE=DIALIN is specified in the PRINTDEV statement, printers cannot be shared.

MODETAB
Specifies the name of the logon-mode table if it is defined locally. If MODETAB is not specified, the IBM-supplied logon-mode table is used.

USSTAB
Specifies the name of an Unformatted System Services (USS) definition table. If USSTAB is not specified, a USS definition table that is supplied by IBM is searched when VTAM receives character-coded input from the logical unit.

Note: The USSTAB operand might be specified on LU statements for terminals, but is typically not specified on LU statements for printer devices. If USSTAB is specified for printer LU statements, USSMSG10 must not be included in the USSTAB defined.

Token-ring subsystem connection

GROUP
Describes a group of communication lines:

AUTOGEN
Specifies how many LINE and PU definition-statement pairs must be added to the GROUP statement.

CALL
Determines whether VTAM is able to establish connections over this group of lines.

ECLTYPE
Specifies whether this GROUP statement defines a physical or logical connection to the token ring.

ISTATUS
Specifies whether the group is to be active or inactive after the major node is first activated.

MAXLU
Specifies the number of logical units that can be associated with a physical unit.

PHYPORT
Specifies the physical port address with which the lines in this logical group communicate. The value must correspond to a PORTADD value in a physical group LINE definition statement.

LINE
Describes one communication line:

ADDRESS
Specifies the relative line number of this line and whether data is transferred in half-duplex or full-duplex mode.

ISTATUS
Specifies whether the line is active or inactive after the major node is first activated.

LOCADD
Specifies the address of the communication controller on the token ring.

MAXTSL
Specifies the maximum number of data bytes that the NCP token-ring interface (NTRI) can transmit. MAXTSL equals the maximum RU size for any PU plus the header length (9 bytes). The maximum RU size is specified by the RUSIZES parameter in the logon-mode table. When RUSIZES=X'8787', for example, MAXTSL=w033; when RUSIZES=X'86F8', MAXTSL=3849. For a formula for calculating RUSIZES, see .

PORTADD
Specifies the physical port address.

RCVBUFC
Specifies the NCP token-ring interface (NTRI) buffer capacity for receiving data from a data link during one data transfer.

General network considerations

When you are establishing a network, consider these:

- Logon-mode table
- VTAM definition names
- LOGAPPL statement and the LU statement

Logon-mode table

The logon-mode table entry must be specified in one of three places:

- The **Logmode** parameter in the Printer Inventory (see Logmode).
- The LOGMODE parameter in the PRINTDEV statement (see LOGMODE).
- The DLOGMOD parameter of the LU statement for the printer. In the cross-domain case, when DLOGMOD is specified, the logon-mode table entry must be defined in the VTAM that owns the printer LU.

If the **Logmode** parameter is specified in the Printer Inventory or the LOGMODE parameter is specified in the PRINTDEV statement, VTAM uses it. If that parameter is not specified, VTAM uses the DLOGMOD parameter. If none of the three, **Logmode**, LOGMODE, or DLOGMOD, is specified, the default DLOGMOD entry for VTAM is used.

If the value specified for the **Logmode** or LOGMODE parameter is not valid, an error message is issued even if the DLOGMOD parameter is valid. Conversely, if the value specified for the DLOGMOD parameter is not valid but the **Logmode** or LOGMODE parameter is valid, no message is issued.

VTAM definition names and PSF

The following values must match:

- The name of the APPL statement in the VTAM definition for the PSF application node
- Any LOGAPPL name of the VTAM definition for generating the NCP for the printer

  Note: LOGAPPL is required only when the **Management mode=Dial in** parameter is specified in the Printer Inventory (see Management mode) or the MGMTMODE=DIALIN parameter is specified in the PRINTDEV statement (see MGMTMODE).

- The **Applid** name that is specified in the Printer Inventory (see Applid) or the APPLID name that is specified in the PRINTDEV statement (see APPLID)

LOGAPPL and the LU statement

When **Management mode=Dial in** is specified in the Printer Inventory (see Management mode) or MGMTMODE=DIALIN is specified in the PRINTDEV statement (see MGMTMODE), LOGAPPL is required only on the LU statement. If LOGAPPL is specified, the DLOGMOD must also be specified in the LU statement for the printer.

Considerations for switched lines

The way PSF establishes a printing session for communication-attached devices on switched lines depends on the management mode parameter specified in either the Printer Inventory or the PRINTDEV statement:

Table 3. Recommended management mode parameter values for switched lines		
Printer Inventory	**PRINTDEV Statement**	**Description**
Management mode=Dial in	MGMTMODE=DIALIN	Recommended for printers where a switched line goes directly to a single printer (for example, the 3820 printer) and is dropped when the LU-LU session ends. PSF then uses a VTAM OPNDST command to accept the CINIT generated by LOGAPPL and DLOGMODE. You also need to specify DISCNT=YES. For more information, see the DISCNT statement and Figure 23 on page 27.
Management mode=Output available	MGMTMODE=OUTAVAIL	Recommended for printers in which the line goes to a control unit (for example, the 3812 printer) and cannot be dropped when the LU_LU session ends. PSF then uses a VTAM SIMLOGON command to request a session with the printer.
Management mode=Immediate	MGMTMODE=IMMED	Recommended for printers in which the line goes to a control unit (for example, the 3812 printer) and cannot be dropped when the LU_LU session ends. PSF then uses a VTAM SIMLOGON command to request a session with the printer.

For more information about the management mode parameter, see Management mode.

AFCCU printer configuration

Specific information and examples for various attachments of AFCCU printers are presented in *Attachment Configuration Handbook for Printers that Use the Advanced Function Common Control Unit*.

DPF and PSF direct considerations

When you are setting up communication between a DPF or PSF direct program and a printer, use the publications for that program.

Not all PSF functions are supported when printing through DPF. See the DPF publication for the IPDS functions that are not supported when you print with a DPF.

Attachment to a microfilm device is not supported through PSF direct.

Chapter 5. Connecting a TCP/IP-attached IPDS printer

This information describes how to configure z/OS, TCP/IP, and PSF to print on TCP/IP-attached Intelligent Printer Data Stream (IPDS) printers, including IPDS printers attached by use of the i-data 7913 IPDS Printer LAN Attachment.

This information also describes how to migrate channel-attached printers to TCP/IP-attached printers.

Overview

PSF views TCP/IP-attached printers in the same way it views SNA-attached printers. PSF does error recovery, handles intervention-required situations, and displays messages exactly as it does for SNA-attached printers.

TCP/IP software provides universal communication services (interfaces) between physical networks and applications. The communication services reside at the network layer and are independent of the topology of the underlying physical network, as Figure 28 on page 41 shows.

Figure 28. Appearance of the TCP/IP connection between a z/OS host and a printer

For example, in communications between a z/OS host and a printer, as in Figure 28 on page 41, the IP network operates like a single network. In fact, the network is physically like that shown in Figure 29 on page 41.
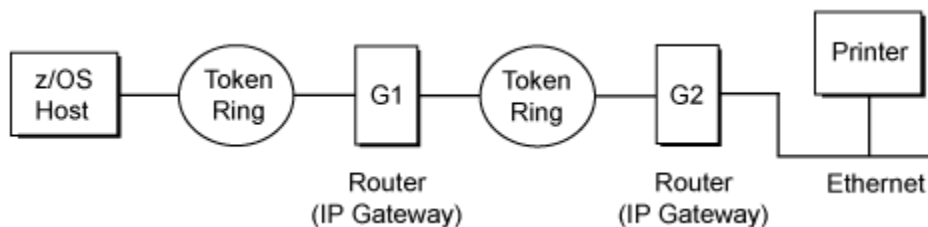
Figure 29. Actual physical TCP/IP connection between a z/OS host and a printer

The routing of information is determined by the IP address, and is performed by IP gateways. After the network is configured correctly, the z/OS host appears to be communicating directly with the TCP/IP-attached printer.

For a more detailed description of Internet Protocol networks, see *TCP/IP Tutorial and Technical Overview*, GG24-3376.

Requirements for using a TCP/IP-attached IPDS printer

To print on TCP/IP-attached IPDS printers through PSF for z/OS, you need Communications Server IP Services that are installed and configured on the z/OS system you are running. Communications Server supports IPv4 and IPv6.

Note: The **TCP/IP job name** parameter in the Printer Inventory or the PARM parameter on the EXEC statement can be used to specify the name of the TCP/IP startup procedure, if different from the default of TCPIP. PSF uses the name of the TCP/IP startup procedure to identify the TCP/IP address space (the TCP/IP started task name that is in the TCP/IP profile data set with the TCPIPJOBNAME keyword), so that PSF can establish communications to a TCP/IP-attached printer.

To specify a TCP/IP startup procedure other than the default, do one of these:

- Specify the name of the TCP/IP startup procedure with the **TCP/IP job name** parameter in the Printer Inventory (see "FSS parameters" on page 58).
- Code this EXEC statement in the appropriate PSF writer procedure:

```
//STEP01    EXEC PGM=APSPPIEP,REGION=4M,PARM=(,,,,tcpip_name)
```

where *tcpip_name* is the name of the TCP/IP startup procedure. You can use the first four parameters in the PARM parameter to specify PSF tracing values. For a description of these parameters, see "PARM parameters on the EXEC statement" on page 144.

To determine whether a printer can be TCP/IP-attached, see the documentation provided with the printer.

Migrating channel-attached printers to TCP/IP

This information describes how you can migrate a printer from channel-attached to TCP/IP-attached by defining initialization statements for the printer with parameters specific for TCP/IP. It also explains how to display intervention and communication failure messages on the operator console and effectively use JES commands to backspace and advance printer output.

To define a channel-attached printer as a TCP/IP-attached printer:

1. For JES2, remove the UNIT parameter from the PRT*nnnn* statement. See "Sample initialization statements for JES2" on page 124.
2. For JES3, remove the device address from the JUNIT parameter on the DEVICE statement. See "Sample initialization statements for JES3" on page 129.
3. Define initialization parameters for a TCP/IP-attached printer in the Printer Inventory or the PRINTDEV statement in the PSF startup procedure. Be sure to remove any parameters that are used for channel-attached printers only, such as the **Channel buffer count** or BUFNO parameter. Table 18 on page 136 shows sample startup procedures for channel-attached and TCP/IP-attached printers.

   Table 4 on page 42 lists the parameters that you might need to set for a TCP/IP-attached printer, depending on whether you are using the Printer Inventory or the PRINTDEV statement.

Table 4. PSF parameters for TCP/IP-attached printers

Printer Inventory	PRINTDEV Statement
Acquire interval	ACQINTV
Connect interval	CONNINTV
Disconnect action	TIMEOUT
Disconnect interval	DISCINTV
Failure action	FAILURE
Issue setup messages	SETUP
Management mode	MGMTMODE
No response action No response action: Notify	NORESP
Port number	PORTNO
Printer IP address	IPADDR
Release interval	RELINTV
Release mode	RELMODE
Response timeout	RESPTIME

See "FSA parameters" on page 62 and "Specifying PRINTDEV parameters" on page 102 for more information about the parameters to use for TCP/IP printers. Also, see "Selecting PSF parameters for shared printers" on page 118 to determine the parameter values to specify.

4. Complete the migration by following the procedure in "Connecting TCP/IP-attached printers" on page 43.

Displaying intervention messages on the console

When a printer condition requires intervention, such as a paper jam, an intervention message is displayed and PSF does not resume processing until the operator takes corrective action. For channel-attached printers, IOS issues the message, IOS003A 0E11,INTERVENTION REQUIRED, to the operator console. For TCP/IP-attached printers, INT-REQ is displayed in the System Display and Search Facility (SDSF).

If you want PSF to send intervention messages to the operator console when the printer is TCP/IP-attached (in addition to SDSF), do one of these:

- Specify the **Issue intervention messages** parameter in the Printer Inventory. See Issue intervention messages.
- Specify the INTRVMSG parameter on the PRINTDEV statement. See INTRVMSG.
- Set the Exit 7 XTP7IMSG field. See "XTP7IMSG" on page 438.

Displaying communications failure messages

To highlight message APS6501A on the operator console when a communications failure occurs with TCP/IP, do one of these:

- Specify the **Highlight communications failure** parameter in the Printer Inventory. See Highlight communications failure message.
- Set the Exit 7 XTP7HLCF field. See "XTP7HLCF" on page 438.

The communications failure message remains highlighted on the operator console until you delete it. To delete the message, enter K E , nn, where nn is the message line number on the console.

If a communications failure occurs with TCP/IP and you did not do one of the actions to highlight the message, message APS6501I is displayed on the operator console. This message is not highlighted and you do not have to delete it. However, because the message is not highlighted, it might scroll off the operator console before the operator sees it.

Using commands to backspace and advance printer output

The operator can use the JES2 backspace (**$B PRTnnnn**) or forward space (**$F PRTnnnn**) commands to backspace or advance printer output. For channel-attached printers, PSF issues commands that immediately stop the printer processing and then do the backspace or forward space command. However, for TCP/IP-attached printers, operator commands are embedded in the data stream. The printer must receive and parse the data before it can recognize the commands. Therefore, extra pages might be printed before the backspace or forward space takes effect.

To improve how quickly the backspace and forward space commands take effect, you need to clear the IPDS buffers from the printer before you issue the JES2 command. See your printer documentation for details about this function. If your printer does not have such a function, specify a Print Adjust/Print One Page function to clear the IPDS buffers.

You can also see your printer documentation for information about synchronizing the printer and PSF for backspacing and forward spacing.

Connecting TCP/IP-attached printers

To connect a TCP/IP printer so that you can print from PSF, do these:

1. Ensure that the Communications Server element of z/OS is installed and operational.

2. Modify the TCP/IP profile on your z/OS system, if necessary.

3. Install the i-data 7913 IPDS Printer LAN Attachment, if necessary (see Appendix M, "Installing the i-data 7913 IPDS Printer LAN Attachment," on page 473).

4. Ping the printer; see "Pinging the printer" on page 45.

5. Define the printer as a writer-controlled printer to JES with MODE=FSS.

6. Define the printer to PSF, including the IP address.

   Note: If you add a printer with an IP address to an existing PSF startup procedure, make sure that you drain all printers in the FSS and cancel the PSF FSS before you attempt to start the new printer.

7. Define a RACF (or other security product) user ID, a group name, or both for the printer.

8. Define an OMVS segment in RACF (or other security product) for the user ID, group name, or both.

9. Define the startup procedure that you are using to the RACF STARTED class. Each startup procedure requires a data segment (STDATA) entry with the user ID, group name, or both that was assigned in Step "7" on page 44. See "Changing the system environment" on page 5.

This information does not provide all the data that you need to install and configure TCP/IP on your z/OS system. For more information about installing TCP/IP, see *z/OS Communications Server: IP Configuration Reference*.

Modifying the TCP/IP profile in z/OS

The TCP/IP profile contains system configuration statements that are used to initialize the TCP/IP address space. For complete information about profile statements, see:

- *z/OS Communications Server: IP Configuration Reference*
- *z/OS Security Server RACF Security Administrator's Guide*

Note: If you change any of the values in the TCP/IP profile, restart TCP/IP to pick up the changes.

These statements in the TCP/IP profile require special considerations when you are printing from PSF on TCP/IP-attached printers:

```
TCPCONFIG INTERVAL 10 SENDGARBAGE FALSE
GATEWAY
```

TCPCONFIG
> The TCPCONFIG statement updates the TCP layer of TCP/IP, including the operating parameters of the TCP keep-alive packets. PSF relies on TCP to detect when a connection with a TCP/IP-attached printer or an i-data 7913 IPDS Printer LAN Attachment is no longer usable. When no data is exchanged between PSF and its connection partner, TCP sends keep-alive packets to the connection partner periodically. TCP uses these periodic packets to discover when a connection is no longer usable even if the connection partner is abruptly powered off or is no longer accessible through the network.
>
> The parameters that are needed for transmitting keep-alive packets are:
>
> **INTERVAL** *minutes*
>> Specifies the number of minutes between keep-alive packet transmissions for a connection. The range is 0 - 35791; the default is 120. A value of 0 means that TCP does not send any keep-alive packet transmissions.
>>
>> For printing on TCP/IP-attached printers, IBM recommends that you specify a shorter interval than the default, such as 10 minutes, for the interval between keep-alive packet transmissions.
>
> **SENDGARBAGE** FALSE|TRUE
>> Specifies whether the keep-alive packets sent by TCP contain 1 byte of random data.
>>
>> **FALSE**
>>> The packet contains no data. This is the default.

TRUE

The packet contains 1 byte of random data and an incorrect sequence number, assuring that the data is not accepted by the remote TCP. Use this value if a target host requires that the keep-alive packet contain data.

GATEWAY

The GATEWAY statement defines static routes, including the maximum packet size for each route. The *Packet_size* parameter of the GATEWAY statement defines the maximum transmission unit (MTU) for the z/OS host. The MTU size must not exceed the maximum size that can be sent through the control unit; if it does, transmission problems occur.

For printing on TCP/IP-attached printers, IBM recommends these:

- An MTU size of 4,096 bytes and a dedicated LAN for high-speed printers on a token-ring network; otherwise, for printers on an Ethernet network, the MTU size must be below 1,500 bytes.
- The MTU size for the z/OS host needs to be the same as the MTU size for the printer.

Note: For values in the GATEWAY statement other than the packet size, specify the values that are correct for your installation.

Installing the i-data 7913 IPDS Printer LAN Attachment

Some IPDS printers must be attached to a LAN by use of the i-data 7913 IPDS Printer LAN Attachment. The i-data 7913 IPDS Printer LAN Attachment must be configured to TCP/IP on an AIX system that is running the **bootp** daemon program, which is required for initialization of the i-data 7913 IPDS Printer LAN Attachment.

If the i-data 7913 IPDS Printer LAN Attachment is not already installed and configured, see Appendix M, "Installing the i-data 7913 IPDS Printer LAN Attachment," on page 473, for information about installing and configuring it.

Pinging the printer

To verify that the z/OS system can establish a connection with the TCP/IP-attached printer, ping the printer from the z/OS system. To do this, issue the following command from a TSO session:

```
TSO PING ip-address
```

In JES2, enter the following command from the System Display and Search Facility (SDSF) command menu:

```
ping ip-address
```

ip-address

Specifies the Internet Protocol (IP) address or host name of the printer, or of the i-data 7913 IPDS Printer LAN Attachment.

This example shows a successful ping:

```
EZA0458I Ping V3R1: Pinging host 192.0.2.33
(Use ATTN to interrupt.)
EZA0463I PING: Ping #1 response took 0.084 seconds.
Successes so far = 1.
```

This example shows an unsuccessful ping:

```
EZA0458I Ping V3R1: Pinging host 192.0.2.33
(Use ATTN to interrupt.)
EZA0464I PING: Ping #1 timed out
```

Ping is not successful

If the ping is not successful, and the printer is attached by use of the i-data 7913 IPDS Printer LAN Attachment, see "Troubleshooting the i-data 7913 IPDS Printer LAN Attachment" on page 484; otherwise, verify these items:

- The printer is powered on.
- The z/OS address is unique in the Internet Protocol network.

  If the IP address of the z/OS system is not unique, see your network administrator to resolve the IP address problem.

- The maximum transmission unit (MTU) size of the IP packet for the z/OS system is equal to the MTU size of the printer. For information about the values that are recommended for the z/OS system, see "GATEWAY" on page 45.

  - To change the MTU size for the z/OS system, change the GATEWAY statement in the MVS™ TCP/IP profile, and restart TCP/IP to pick up the changes.
  - To change the MTU size for the printer, see the documentation for your printer.

If these items are all in order, consult your Internet Protocol network administrator about a possible network problem.

Ping is successful

A successful ping typically indicates that the z/OS system can communicate with the printer; however, you might receive a successful ping even though the IP address of the TCP/IP-attached printer is a duplicate of another IP address.

If PSF is unable to establish a network connection with the TCP/IP-attached printer, or if PSF output for this printer prints elsewhere, follow these steps to determine whether the IP address of the printer is unique:

1. Turn off the printer and the i-data 7913 IPDS Printer LAN Attachment, if any.
2. Wait at least 5 minutes for TCP/IP to clear the Address Resolution Protocol (ARP) tables.[1]
3. Enter the **ping** command again from the z/OS system.

If you receive a successful response to the ping command, a duplicate IP address exists. Consult your Internet Protocol network administrator.

Defining the printer to JES

To define the printer to JES, see "Sample initialization statements for JES2" on page 124 and "Sample initialization statements for JES3" on page 129.

Defining the printer to PSF

Each TCP/IP-attached printer must be defined to PSF. To define the printer, see "PSF startup procedures for AFP printers" on page 135.

Operating a TCP/IP-attached printer

To operate a TCP/IP-attached printer, including a printer that is attached by the i-data 7913 IPDS Printer LAN Attachment, you use JES operator commands, as if the printer were channel-attached or SNA-attached.

---

[1] If your installation specified a longer interval on the ARPAGE configuration statement in the TCP/IP profile, you might need to wait longer. For information about the ARPAGE statement, see *z/OS Communications Server: IP Configuration Reference*.

In order for JES operator commands to take effect immediately on TCP/IP-attached printers, you must first stop the printer and clear the printer buffers. See your printer documentation to see whether the printer has this capability and how to do it.

Attention: When you are issuing any JES operator command, use the CLEAR BUFFERS option at the printer before you issue the JES command; otherwise, pages might be lost. Pages might also be lost if you use the PRINT ADJUST/ONE PAGE PRINT function without using the CLEAR BUFFERS option as the last function at the printer.

For example, if the JES RESTART command was issued, the PSF message data set (if present) or the job trailer page might be lost. After the job is restarted, the message data set and job trailer page are printed when the job completes.

Overview of the TCP/IP and PSF interface

TCP/IP must be running on the z/OS system before any attempt is made to start PSF. PSF must establish a communication path with TCP/IP (by issuing the INITAPI macro from each printer FSA) before any communication can occur.

TCP/IP must keep state information for this interface. If a TCP/IP abend occurs, the state information is gone and TCP/IP has no knowledge of any interfaces established previously. This has several implications to PSF FSAs:

- If an FSA is actively using TCP/IP at the time of the abend, a PSF abend also occurs because communication is ended with the printer. The abend is not a restartable abend because a severe communication error exists that must be fixed.
- If an FSA is idle when a TCP/IP abend occurs or is stopped, the FSA remains active until a print job is acquired, at which time an FSA abend occurs if TCP/IP is still not active. If TCP/IP is recycled and is active again, PSF gets a bad return code from TCP/IP because PSF is still using the old interface established from the previous TCP/IP session. PSF attempts to reestablish the link with TCP/IP again by issuing an INITAPI. The print job is sent after the link is established.

No TCP/IP services are run until a TCP/IP printer is started. Then, hangs or abends in TCP/IP services might influence other FSAs (any attachment) under the same FSS as the TCP/IP FSA. For this reason, it is recommended that TCP/IP-attached printers are not driven by the same FSS as printers with any other attachment type (SNA, parallel channel, or ESCON).

Starting a TCP/IP-attached printer

To start a TCP/IP-attached printer, do these:

1. Start TCP/IP.
2. Power on the printer.
3. Power on the i-data 7913 IPDS Printer LAN Attachment, if any.
4. Start the printer FSA by the method described in "Starting PSF FSAs" on page 146.

Displaying TCP/IP status

If you need to know the status of a TCP/IP connection, you can do so without taking a PSF trace. By using the MVS MODIFY command as a PSF operator interface, you can dynamically display the TCP/IP status on the console. However, this interface can be used only in deferred-printing mode. For more information about using the MODIFY command to display TCP/IP status, see *PSF for z/OS: Diagnosis*.

Stopping a TCP/IP-attached printer

You can stop a TCP/IP-attached printer in several ways:

- The preferred method is to first stop the PSF FSA for the printer by entering the JES commands from the z/OS console. Then, you can turn off power to the printer and the i-data 7913 IPDS Printer LAN Attachment, if any.

See "Stopping PSF FSAs and FSSs" on page 147 for more information about stopping FSAs.

- You can turn off power to a printer that is not attached by use of the i-data 7913 IPDS Printer LAN Attachment.

  This procedure is not recommended because it might take PSF some time to recognize that the printer is powered off. The length of that time depends on the value specified for the keep-alive transmissions, described in "Modifying the TCP/IP profile in z/OS" on page 44.

  To end the PSF FSA for the printer, use the JES commands described in the first bullet. If you are unable to stop or cancel the printer by use of the JES commands, enter the following command:

  ```
  MODIFY fss_name,FORCE,fsa_name
  ```

- You can turn off power to a printer that is attached by use of the i-data 7913 IPDS Printer LAN Attachment. The 7913 notifies PSF that the printer is powered off, and the printer FSA eventually ends normally.

- You can turn off power to the i-data 7913 IPDS Printer LAN Attachment without turning off power to the printer.

  This procedure is not recommended because it might take PSF some time to recognize that the attachment is powered off. The length of that time depends on the value specified for the keep-alive transmissions, described in "Modifying the TCP/IP profile in z/OS" on page 44.

  To end the PSF FSA for the printer, use the JES commands described in the first bullet. If you are unable to stop or cancel the printer by use of the JES commands, enter the following command:

  ```
  MODIFY fss_name,FORCE,fsa_name
  ```

  Note: Before you power the i-data 7913 IPDS Printer LAN Attachment back on, be sure to power off the printer and then power on again.

Understanding performance issues

Performance testing shows that a TCPSENDBFRSIZE buffer size of 16384 severely restricts printer performance. Therefore, IBM recommends that you use a minimum TCPSENDBFRSIZE buffer size of 32768. Performance testing also shows that 262144 is the optimum buffer size for TCP/IP-attached high-speed duplex printers. Avoid printer performance issues by consulting with your system programmer to select the correct buffer size.

PSF runs with either 8K, 32K, or 64K send buffers. When attaching to the printer, PSF indicates that its maximum send buffer size is 32K or 64K, based on the setting of the **tcpip-64k-buffer-support** AFPPARMS parameter. The printer responds back indicating that it supports a maximum send buffer size of either 32K or 64K. That size is used. Ensure that your printer supports 64K send buffers before setting the **tcpip-64k-buffer-support** parameter.

Chapter 6. Working with libraries

This information provides general information about defining library data sets, formatting libraries, concatenating defined libraries to system libraries, and working with user libraries in deferred-printing mode.

Resource libraries contain the resources needed for printing. Because some of the libraries (font, object container, and page segment) can contain multiple resource types and because PSF does not enforce a prefix for most of the eight-character resource names, you need to define a naming convention that identifies each type of resource in a library. IBM recommends that you use a two-character prefix.

Table 5 on page 49 shows the PSF resource libraries, the types of resources stored in each library, and the required and IBM-recommended prefixes. For more information about resources, see *PSF for z/OS: User's Guide*.

Table 5. PSF system resource libraries, resource types, and resource prefixes

Library	Type of Resource	Description	Prefix
Font	FOCA font	Stored in PDS or PDSE font resource libraries:	
		240- and 300-pel character set	C0
		3800 character set	C1–CG
		Outline character set	CZ
		Code page; extended code page	T1
		240- and 300-pel coded font	X0 (required)
		3800 coded font	X1–XG (required)
		Outline coded font	XZ (required)
		Stored in UNIX file path resource libraries:	
		Extended code page	T1
	TrueType and OpenType font	Stored in UNIX file path resource libraries.	No prefix used
Form definition	Form definition	Stored in PDS or PDSE resource libraries.	F1 (required)

Library	Type of Resource	Description	Prefix
Object container	Color mapping table	Stored in PDS or PDSE resource libraries.	M1
	Encapsulated PostScript	Stored in PDS, PDSE, or UNIX file path resource libraries.	E1
	Image (IOCA)	Hard and soft IOCA resources. IOCA resources that are identified in the data stream as hard resources with the Map Data Resource (MDR) structured field must be placed in the object container system library, a user library, or inline with the print data; soft IOCA resources can be placed in the object container or page segment library. Stored in PDS, PDSE, or UNIX file path resource libraries.	I1
	IOCA tile	Stored in PDS, PDSE, or UNIX file path resource libraries.	IT
	Microfilm setup	Stored in PDS or PDSE resource libraries.	H1
	PDF resource	Stored in PDS, PDSE, or UNIX file path resource libraries.	PR
	PDF single-page	Stored in PDS, PDSE, or UNIX file path resource libraries.	PP
	Other data object resources	Stored in PDS, PDSE, or UNIX file path resource libraries. See the object type registry in *Mixed Object Document Content Architecture Reference* at AFP Consortium (www.afpcinc.org).	No prefix recommended
Overlay	Overlay	Stored in PDS or PDSE resource libraries.	O1
Page definition	Page definition	Stored in PDS or PDSE resource libraries.	P1 (required)
Page segment	Bar code (BCOCA)	Stored in PDS or PDSE resource libraries.	B1
	Graphic (GOCA)	Stored in PDS or PDSE resource libraries.	G1
	IOCA	IOCA resources that are not identified with the MDR structured field (soft resources); stored in PDS or PDSE resource libraries.	I1
	Text (PTOCA)	PTOCA object with OEG; stored in PDS or PDSE resource libraries.	No prefix recommended
	Page segment	Stored in PDS or PDSE resource libraries.	S1

Table 5. PSF system resource libraries, resource types, and resource prefixes (continued)

Note: When you are installing PSF, you do not define user libraries. They are set up by a user and are dynamically accessed by PSF when specified in the user's JCL.

Library format

Libraries can be:

- A concatenation of partitioned data sets (PDS or PDSE) that contain members for one or more resources. Typically, a resource is built by an application or by an AFP licensed program and is stored in a library for other print jobs. Partitioned data sets must have record formats of variable-blocked machine (VBM), variable-blocked ANSI [2] (VBA), variable machine (VM), variable ANSI (VA), variable-blocked (VB), or variable (V).

  Note: Generally, PSF resources that are created by IBM products, such as Page Printer Formatting Aid (PPFA), are built with a record format of VBM.

- A path or set of paths for system UNIX files (Hierarchical File System (HFS) or z/OS File System (zFS) files) that contain TrueType and OpenType font objects.

Concatenating defined libraries to system libraries

You can define libraries and concatenate them to the system libraries. For example, a single library can contain a combination of Font Object Content Architecture (FOCA) fonts, page segments, and overlays. Use unique member names within concatenated libraries so that PSF can access a particular member in a library. If a member in a concatenated library has the same name as a member in the system library, PSF uses the member that it finds first. When PSF is started, libraries are searched in the order in which they were concatenated. For information about concatenating libraries, see the JCL user's guide for your operating system.

With user library support, you might not want to concatenate all such libraries to the system libraries. Do not concatenate to the system libraries any libraries that have only a few users, or that contain sensitive resources. You can limit access to such libraries by using RACF control, and the users can specify these libraries in their JCL with the USERLIB keyword.

User libraries in deferred-printing mode

When PSF processes the print data set, it accesses user libraries for AFP resources. FOCA fonts, page segments, overlays, page definitions, form definitions, object containers, Bar Code Object Content Architecture (BCOCA), Graphics Object Content Architecture (GOCA), Image Object Content Architecture (IOCA), and Presentation Text Object Content Architecture (PTOCA) resources are found in user libraries that print-job submitters specify with the USERLIB keyword in their OUTPUT JCL statement; extended code pages that are stored in UNIX files, TrueType and OpenType fonts, data object resources, and any resource that is installed with a resource access table (RAT) are found in user path libraries specified with the USERPATH keyword. For example, the following example directs PSF to search for AFP resources in the specified user libraries. If unable to find the resources in the user libraries, PSF searches the system libraries defined in the startup procedure.

```
//OUT1    OUTPUT    USERLIB=(USER.IMAGES.AFP,USER.FONTS.AFP)
                    USERPATH=('/u/fonts/truetype/','/u/fonts/truetype/myfonts/')
```

For more information about user libraries, see *PSF for z/OS: User's Guide*.

Controlling access to USERLIB libraries

You can use RACF to control access to user libraries specified with the USERLIB keyword. PSF checks whether the print-job submitter has RACF authority to access a specified user library; if not, PSF does not print the job. When the data set is finished printing, PSF deletes the resource from the printer.

PSF sends a RACROUTE to RACF by using the user ID of the user who submitted the job. PSF checks to make sure that the user has READ authority to the data sets specified in the USERLIB keyword for the job. Make sure that your user ID has READ authority for any data sets specified in the USERLIB keyword.

---

[2] ANSI stands for American National Standards Institute.

Controlling access to USERPATH libraries

You use z/OS UNIX System Services to control access to user path libraries specified with the USERPATH keyword. The user or the system administrator can set permissions to access the directories and files in the USERPATH libraries. Read permissions must be set for each path directory and file that PSF is accessing. If the permissions for each directory or file are not set correctly, PSF cannot access the path library.

For more information about z/OS UNIX System Services, see *z/OS UNIX System Services Command Reference*.

Constraints to dynamic allocation

Keep in mind the following constraints to dynamic allocation:

- You can specify up to eight user libraries in your JCL.
- User library data sets must be accessible on all systems on which output is selected to print. This might necessitate reconfiguring shared volumes or storage classes, and allocating the resource libraries to such volumes or storage classes.
- PSF does not support the USERLIB or USERPATH keyword in direct-printing mode.
- Integrated catalog facility (ICF) catalogs are not allocated to the user's address space in PSF. To free the catalogs for maintenance, you can enter a catalog address space (CAS) restart or a MODIFY CATALOG, UNALLOCATE command.
- Dynamic allocation by any task in an address space serializes on SYSZTIOT, which might cause a performance problem when PSF supports large numbers of FSAs in an address space. Therefore, use user libraries only when necessary.
- The maximum number of catalogs that an address space can allocate is 1635, which can optionally be increased to 3274. If every user in the system has a separate catalog, be aware of the possibility of approaching the maximum number of catalogs.

Chapter 7. Using the Infoprint Server Printer Inventory

The Printer Inventory component of the Infoprint Server manages the inventory of printer information that is used by NetSpool, Print Interface, IP PrintWay, and PSF for z/OS. These applications obtain information from the inventory about handling jobs on the JES spool and starting and stopping their processes. You use the Printer Inventory to define and update printer definitions and define the functional subsystem (FSS), the functional subsystem application (FSA), and printer definition pool parameters.

In deferred-printing mode, PSF can use printer initialization, tracing, and execution option parameters that are defined in the Printer Inventory instead of those parameters that are defined in the PRINTDEV statement (see "Specifying PRINTDEV parameters" on page 102) or the EXEC statement (see "EXEC" on page 142) of the PSF startup procedure or in the Exit 7 initialization (INIT) call (see "Resource-management exit: Exit 7" on page 294).

With the Printer Inventory, you can define a printer in the PSF startup procedure and JES initialization statements before the printer is used. You can also define a range of printers in the JES initialization statements if you support numerous printers. When the new printer is added, you can assign variable parameters for it in the Printer Inventory. Using the Printer Inventory saves you time because you do not need to restart all the printers in a startup procedure when you add a printer or change parameters for an existing printer. The Printer Inventory also makes it much easier to define parameters than through the PSF startup procedure or Exit 7 (see "Specifying Printer Inventory parameters" on page 57 for more information).

Note: When you use the Printer Inventory, you must specify the parameters that you want to use in the Printer Inventory; if you specify them in the startup procedure or in the Exit 7 INIT call, the parameters are ignored. However, any parameters that are specified in the Exit 7 begin-data-set call (BDSC) or end-data-set call (EDSC) override those parameters that you specify in the Printer Inventory.

This information describes how to access the Printer Inventory and customize PSF to use the Printer Inventory, explains how the Printer Inventory works with PSF, and describes Infoprint Central and the Infoprint Server common message log.

Accessing the Printer Inventory

You can use the Printer Inventory component of Infoprint Server to define PSF printers without licensing the Infoprint Server feature of z/OS. The Printer Inventory without the Infoprint Server feature is referred to as the "Printer Inventory for PSF".

When you use the Printer Inventory for PSF, you can start and stop Printer Inventory daemons and create PSF FSS and FSA definitions in the Printer Inventory the same way as with the Infoprint Server feature. However, the ISPF panels display only those functions that you are allowed to use with PSF.

For more information about using and configuring the Printer Inventory, see *z/OS Infoprint Server Printer Inventory for PSF* or *z/OS Infoprint Server Customization*.

Customizing PSF to use the Printer Inventory

To use the Printer Inventory with PSF, do these tasks:

1. Define one or more PSF FSAs in the Printer Inventory (see "Specifying Printer Inventory parameters" on page 57 for more information).

   You can use a migration program to copy parameters from the PRINTDEV statement, the EXEC PARM statement, and the AFPPARMS control statement into the Printer Inventory (see Appendix A, "Migrating PSF parameters to the Printer Inventory ," on page 353). You must manually migrate parameters from Exit 7.

2. Change the printer startup procedure to specify INV=*piname* as the first parameter in the PARM field of the EXEC statement:

```
//  EXEC PGM=APSPPIEP,PARM=('INV=piname')
```

where *piname* is the four-character name of the Printer Inventory that is specified on the **inventory** attribute in the Infoprint Server configuration file, which by default is in `/etc/Printsrv/aopd.conf`. For example, this EXEC statement specifies the default name of the Printer Inventory:

```
// EXEC PGM=APSHPOSE,PARM=('INV=AOP1')
```

The INV=*piname* parameter indicates that PSF uses the Printer Inventory and obtains parameters from the specified Printer Inventory for each printer in the startup procedure. No other parameters in the PARM field are used when PSF uses the Printer Inventory. For more information, see "PARM parameters on the EXEC statement" on page 144.

See *z/OS Infoprint Server Customization* or *z/OS Infoprint Server Printer Inventory for PSF* for more information about the Infoprint Server configuration file and defining printers in the Printer Inventory.

How PSF uses the Printer Inventory

When PSF starts, it determines whether the Printer Inventory is used. If the PARM field of the EXEC statement contains INV=*piname* as the first parameter, PSF ignores any other PARM parameters that are specified and connects to the Printer Inventory specified. PSF then calls the Infoprint Server to get FSS parameters from the Printer Inventory.

When an FSA is started, PSF gets the PRINTDEV JCL parameters and then calls the Infoprint Server to get the FSA parameters from the Printer Inventory. Those parameters that are specified in the Printer Inventory override the equivalent parameters in the PRINTDEV. Any parameters available in the Printer Inventory, but not specified, are ignored if they are specified in the PRINTDEV.

Those parameters that are specified in the Printer Inventory override the equivalent parameters set during the Exit 7 INIT call. Any parameters available in the Printer Inventory, but not specified, are ignored if they are set in the Exit 7 INIT call.

Infoprint Central

Infoprint Central is a print management application that runs on the web. Operators use Infoprint Central to display information about and work with printers that you define in the Printer Inventory. Operators can:

- Display the properties and the status of PSF printers.
- Start, stop, space, interrupt, pause, and ping PSF printers.
- Change job-selection criteria for PSF printers.

If PSF is enabled to write messages to the Infoprint Server common message log, you can also:

- View FSA and print job messages that PSF writes to the common message log.
- Search for print jobs and view the properties for each job.
- Release held print jobs, delete jobs, change the priority of jobs, and move jobs (as long as PSF has not started processing the jobs).

See "Common message log" on page 55 for information about customizing PSF to use the common message log with the Printer Inventory.

To use Infoprint Central, you must have a license for the Infoprint Server feature.

For more information about using Infoprint Central, see *z/OS Infoprint Server Operation and Administration*.

Common message log

PSF can write FSS, FSA, and print job messages to the Infoprint Server common message log. The common message log links the messages to their respective objects with a log name so an operator can view the messages in Infoprint Central. With the common message log function, operators can also use Infoprint Central to:

- Search for print jobs and view the properties for each job.
- Release held print jobs, delete jobs, change the priority of jobs, and move jobs (if PSF is not processing the jobs).

PSF writes FSS, FSA, and Print Job message types to the common message log. PSF does not write Exit WTO, NST, Operator Interface, or Secured Console message types.

The common message log requires:

- A license for the Infoprint Server feature
- JES2 or JES3 at the supported z/OS level

To enable PSF to write messages to the common message log, you must customize PSF and select the **Log messages** parameter in the Printer Inventory:

1. Update these BPXPRMxx parameters in the SYS1.PARMLIB member:

   MAXSOCKETS
   This parameter is on the NETWORK statement. Increase the value by 3 for each FSS and by 1 for each FSA.

   MAXPROCSYS and MAXPROCUSER
   Increase the value by 1 for each FSS because the common message log adds a process for each FSS.

   See to make sure that the MAXPROCSYS and MAXPROCUSER parameters are set correctly.

2. Define a valid OMVS segment in RACF for the user ID, group ID, or both so PSF can write messages to the common message log.

3. Select the **Log messages** parameter in the Printer Inventory to use the common message log. You do not need to restart PSF when you set or change this parameter. The update takes effect after the current print job is complete.

Note: When PSF is using the common message log, it has these sysplex limitations:

- The Printer Inventory and common message log cannot be shared across multiple systems. Therefore, if you configure PSF to print on the same printers from different z/OS systems, Infoprint Central users must open separate instances of their web browser so they can log on to multiple z/OS systems to see all print jobs on a printer's queue and to see all messages about a printer. This is because from each instance of a web browser, you can log on to only one z/OS system and see print jobs, printer queues, and Infoprint Server messages for that one z/OS system only.

- In JES3 systems only, if the systems in the sysplex share the JES spool, you can run Infoprint Server on only one system in the sysplex. If you want to run Infoprint Server on more than one system, PSF on each system must select different print jobs from the JES spool. To accomplish this, you must define different job-selection rules for PSF on each system. For example, PSF on system A might select print jobs in class P, while PSF on system B might select print jobs in class Q.

Chapter 8. PSF parameters

This information lists the PSF parameters that you can specify in the Printer Inventory, the PRINTDEV statement of the PSF startup procedure, and the AFPPARMS control statement.

Note: The Printer Inventory and the AFPPARMS control statement are used for deferred-printing mode; they are not used for direct-printing mode.

This information also describes the parameters required for attaching and sharing printers.

Specifying Printer Inventory parameters

To use the Printer Inventory, you must:

1. Make sure that INV=*piname* is the first parameter in the PARM field of the EXEC statement:

   ```
   //  EXEC PGM=APSPPIEP,PARM=('INV=piname')
   ```

   where *piname* is the four-character name of the Printer Inventory. See "Customizing PSF to use the Printer Inventory" on page 53 for more information.

2. Specify in the Printer Inventory the parameters you want to use. You can use the Printer Inventory Definition Utility (PIDU) or Infoprint Server ISPF panels to define parameters in the Printer Inventory.

   Note: Do not specify in the PRINTDEV statement those parameters available in the Printer Inventory; PSF ignores them if you are using the Printer Inventory.

3. To use the Infoprint Server common message log, select the **Log messages** parameter in the Printer Inventory. See Figure 31 on page 58. You do not need to restart PSF when you set or change this parameter. The update takes effect after the current print job is complete.

Figure 30 on page 58 shows an example of PSF parameters that are defined in a PIDU statement.

```
create psf-fss SAMPFSS
  tcpip-job-name = TCPIP
;
create fsa PRT1                         # For a channel-attached printer
  fsa-type = psf-channel
  channel-buffer-count = 1
  page-definition = A08682
  form-definition = A10110
  chars = {60D8}
  print-error-messages = yes
  print-error-messages-maximum = 16
  print-error-reporting = none
  trace-mode = internal
;
create fsa PRT2                         # For an SNA-attached printer
  fsa-type = psf-sna
  page-definition = A08682
  form-definition = A10110
  chars = {60D8}
  print-error-messages = yes
  print-error-messages-maximum = 16
  print-error-reporting = none
  trace-mode = internal
  failure-action = connect
  printer-disconnect-interval = 0
  printer-management-mode = immediate
  applid = PRT2APPL
  luname = PRT2LU
;
create fsa PRT3                         # For a TCP/IP-attached printer
  fsa-type = psf-tcpip
  page-definition = A08682
  form-definition = A10110
  chars = {60D8}
  print-error-messages = yes
  print-error-messages-maximum = 16
  print-error-reporting = none
  trace-mode = internal
  failure-action = connect
  printer-disconnect-interval = 0
  printer-management-mode = immediate
  printer-ip-address = 192.0.2.33
;
```

Figure 30. Sample PIDU statement for three printers

FSS parameters

Figure 31 on page 58 shows a sample ISPF panel for defining an FSS for PSF.

```
Add                              PSF FSS
 Command ==> _____

 FSS Name. . . PSFFSS
 Description . PSF                                                    (extend)

   TCP/IP job name. . . . _____
   NST trace dsname . . . _____
   PINST trace dsname . . _____
   Trace table size . . . 32
   _ Trace prompt
   _ Unicode enabled
   / Log messages
   / Blank truncation
   Restore blanks classes . . _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
                              _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
```

Figure 31. Sample ISPF Printer Inventory panel for defining an FSS for PSF

"FSS parameters" on page 58 describes the Functional Subsystem (FSS) parameters that you can define in the Printer Inventory and compares them to the parameters you can define on the PRINTDEV statement or the EXEC PARM statement of the PSF startup procedure, in PSF installation Exit 7, or in the AFPPARMS control statement. All parameters are optional unless specified otherwise. When you are

defining parameters on the ISPF panels, see the online help for information about each parameter. For complete information about defining parameters in the Printer Inventory, see *z/OS Infoprint Server Customization* or *z/OS Infoprint Server Printer Inventory for PSF*.

Table 6. FSS Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters. If you change the value for an FSS parameter, you must restart the FSS.

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description

Blank truncation
Values:
  / = Yes
  Blank = No
	blank-truncation
	Values:
	  yes
	  no
		PARM= (,,,,,,*blank_truncation*)
		Values:
		  BLNKTRNC
		  NOBLNKTRNC
			Specifies whether trailing blanks in non-page-mode data sets received from JES2 are truncated. Non-page-mode data sets include line-mode data, text data, mixed-mode data, and XML data. If you specify this parameter for composed-text page data sets (for example, output created by the Document Composition Facility), it has no effect.
			
			Note: The PRMODE= parameter, on both the JCL and JES2 output device initialization statements, does not affect blank truncation. To turn off blank truncation for some SYSOUT classes, see the **restore_blanks_classes** parameter.
			
			The default is **BLNKTRNC** (**yes**).

Log messages
Values:
  / = Yes
  Blank = No
	log-messages
	Values:
	  yes
	  no
			Indicates whether PSF writes messages to the Infoprint Server common message log. The default is No.
			
			Note: You do not need to restart PSF when you set or change this parameter. The update takes effect after the current print job is complete.

NST trace dsname	nst-trace-dsname	PARM= (*NSTddname*)	Specifies the name of the data set that PSF directs a notify subtask (NST) trace to. This name must be different than the data set name PSF directs an FSA external trace to. The trace data set must exist and be cataloged before the first PSF FSA in the FSS is started. **Note:** An NST trace is recorded only if an FSA internal or external trace of the page printing writer (PPWTR) component is also active for that FSA. For more information about using traces, see *PSF for z/OS: Diagnosis*.

Table 6. FSS Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters. If you change the value for an FSS parameter, you must restart the FSS. (continued)			
Printer Inventory: ISPF Panel	**Printer Inventory: PIDU**	**Equivalent Parameter**	**Description**
PINST trace dsname	pinst-trace-dsname		Specifies the name of the data set that PSF directs a Printer Inventory notify subtask (PINST) trace to. This name must be different than both the data set name PSF directs an FSA external trace to and the NST trace data set name. The trace data set must exist and be cataloged before the first PSF FSA in the FSS is started.
			Note: A PINST trace is recorded only if an FSA internal or external trace of the page printing writer (PPWTR) component is also active for that FSA.
			For more information about using traces, see *PSF for z/OS: Diagnosis*.

Table 6. FSS Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters. If you change the value for an FSS parameter, you must restart the FSS. *(continued)*

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description
Restore blanks classes	restore-blanks-classes	PARM= (,,,,,,,*restore_blanks_classes*) **Values: A–Z, 0–9**	Specifies the SYSOUT classes for which you do **not** want trailing blanks truncated for non-page-mode-data sets. Non-page-mode data sets include line-mode data, text data, mixed-mode data, and XML data. This attribute is used only when `blank-truncation=BLNKTRNC`. This attribute specifies up to 36 SYSOUT class values. Each class is one alphanumeric character. Do not separate the class values by commas or blanks. The value of the SYSOUT class of the data set received from JES is compared to the **restore-blanks-classes**. If the SYSOUT class matches one of these classes, trailing blanks are **not** truncated. Specify the same classes for which you specified BLNKTRC=NO on the JES2 OUTCLASS(v) statement before z/OS 2.4. For example, if your JES2 OUTCLASS statement specified BLNKTRNC=NO for: • OUTCLASS(K,L,Y), specify `restore_blanks_classes = KLY` • OUTLCLASS(S-Z), specify `restore_blanks_classes = STUVWXYZ` Or, specify this attribute for SYSOUT classes that have: • Non-page-mode data sets that require trailing blanks to be preserved • Data sets containing double-byte font characters
TCP/IP job name	tcpip-job-name	PARM= (,,,,*tcpip_name*)	Specifies the name of the TCP/IP startup procedure. If you changed the name of the TCP/IP startup procedure, specify the new name for this parameter. If you do not use this parameter, the FSA uses the default TCP/IP startup procedure name of **TCPIP**.

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description

Trace prompt
Values:
  / = Yes
  Blank = No
	trace-prompt
	Values:
	  yes
	  no
		PARM= (,,*prompt*)
		Values:
		  PROMPT
		  NOPROMPT
			Indicates whether an operator response is required each time the FSS starts to initialize the PSF operator interface. The default is **NOPROMPT** (**No**).
			
			For more information about using traces, see *PSF for z/OS: Diagnosis*.

Trace table size	trace-table-size	PARM= (,,,*pagecount*)	Specifies a number that indicates how many 4 KB pages of storage are allocated for the FSA internal trace table. This allocation occurs only if the trace mode is **Internal**, **Sync**, **Full**, **Limit**, or **IPDS**. **Values:** 1–999 The default is 32. For more information about using traces, see *PSF for z/OS: Diagnosis*.

Unicode enabled
Values:
  / = Yes
  Blank = No
	unicode-enabled
	Values:
	  yes
	  no
		PARM= (,,,,UNICODE)	Specifies that PSF is enabled to use the system conversion services that z/OS provides. This parameter is ignored because PSF 4.5 or later is always Unicode-enabled.

Table 6. FSS Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters. If you change the value for an FSS parameter, you must restart the FSS. *(continued)*

FSA parameters

Figure 32 on page 63 and Figure 33 on page 64 show sample pages of an ISPF panel that is used to define the PSF parameters for an SNA-attached FSA. To see how these parameters are defined in a PSF startup procedure, see the fields that are commented with "PI" in "Sample PSF startup procedure for four printers" on page 137 and "Sample PSF startup procedures for three separate printers" on page 140.

```
Add                          PSF FSA, SNA
Command ==> _____

FSA Name. . . PRT2____
Description . SNA-attached printer_____ (extend)
Location. . . Building 001_____ (extend)
                                                        More:     +
Operator security profile
    . . . PRT2PROF_____
Processing Information:
  _ Blank compression
  _ Consolidate IM1 images
  _ Inhibit recovery
  _ Close libraries when idle
  _ Capture inline resources
  _ Release data set when repositioning
  _ Suppress copy marks
  _ Issue intervention messages
  / CSE sheet eject
  _ CSE preserve page position
  _ Use Line-Mode Migration LINECT
  Above the bar storage . . _   1. None 2. Inline resources
  Check CSE fit . . . . . . _   1. No   2. First  3. All
  CSE orientation . . . . . _   1. Portrait  2. Landscape
  Eject to front facing . . _   1. None  2. Job  3. Document  4. Both
  Issue setup messages. . . 1   1. None  2. Burst  3. Forms  4. All
  Offset stacking . . . . . 3   1. None  2. Data set  3. Job
  Auxiliary files MO:DCA level . 1  1. None  2. IS/3  3. BPF match
  Default process mode. . . _____
  Resolution. . . . . . . . ___    (240, 300)
  Resource deletion ratio . ___    (0 - 100)

Resources:
  Form definition. . A10110__
  Page definition. . A08682__
  Character sets . . 60D8  ____  ____  ____
  Color map. . . . . _____
  Com setup member . _____
  Overlay. . . . . . _____
  / Prune double-byte fonts
  / Prune single-byte fonts
  _ Map to outline fonts
  _ Recover from font not found
  _ Send default character

Retained Resource Counts:
  Form definitions . _____
  Page definitions . _____
  Page segments. . . _____
  Fonts. . . . . . . _____
  Object containers. _____

Input Tray Substitution:
  Source tray:  Substitute trays:
                Simplex  Duplex

  ___           ___      ___
  ___           ___      ___
  ___           ___      ___
  ___           ___      ___

Error Reporting Values:
  _ Error disposition supported
  _ Send msgs to SYSOUT
  Print error messages . . 2  1. No  2. Yes
       Maximum messages. . 16   (0-999)
  Print error reporting. . 1  1. None  2. All  3. Character  4. Position
```

Figure 32. Sample ISPF Printer Inventory panel for defining an SNA-attached FSA (Page 1 of 2)

```
Add                           PSF FSA, SNA
Command ==> _____

FSA Name. . .  PRT2____
Description .  SNA-attached printer_____ (extend)
Location. . .  Building 001_____ (extend)
                                                            More:   - +
 Separator page:
    / Interrupt message page
      _ Mark page
      _ Offset page
      Copies . . __    (1-10)

Connection:

    Connect Interval. . . _____    (0-86400 seconds)
    Failure action. . . . 2  1. Stop  2. Wait for connect
    Applid. . . . . . . . PRT2APPL
    Disconnect action . . 2  1. Stop  2. Redrive
    Disconnect interval . 0____    (0-86400)
    Logmode . . . . . . . _____
    LU name . . . . . . . PRT2LU__
    Management mode . . . 1  1. Immediate  2. Dial in  3. Output available
    No response action. . _  1. Notify JES      2. Notify user
                             3. Notify operator  4. Terminate
       Notify . . . . . . _____
    Response timeout. . . _____   (0-86400)
    _ End SNA conversation

Printer Sharing:
    Release mode . . . . 3  1. Idle  2. Time  3. None
    Release interval . . _____   (0-86400 seconds)
    Acquire interval . . _____   (0-86400 seconds)

Security Labeling:
    Label data pages. . . . . _  1. Yes  2. No
    Label separator pages . . _  1. Yes  2. No
    Restrict printable area . _  1. Yes  2. No

Debugging:
    Dump:
       Code . . _____   Message ID . . _____   Count . . 1_   (1-99)
    Trace:
          Trace mode . . . . 2  1. None  2. Internal  3. Sync
                                4. Full   5. Limit     6. IPDS
          Trace table size . 32_   (1-999)
          FSA trace dsname . _____

3800 Compatibility:
    _ Override default font
    Set media origin to 3800 origin for:
       _ Data set
       _ Data set header
       _ Job header
       _ Job trailer
       _ Message data set
```

Figure 33. Sample ISPF Printer Inventory panel for defining an SNA-attached FSA (Page 2 of 2)

```
                      PSF FSA, TCP/IP

FSA Name. . . _____
Description . _____ (extend)
Location. . . _____ (extend)
Operator security profile
  . . . _____

Processing Information:
  _ Blank compression
  _ Consolidate IM1 images
  _ Inhibit recovery
  _ SNMP enabled
  _ Close libraries when idle
  _ Capture inline resources
  _ Release data set when repositioning
  _ Suppress copy marks
  / Issue intervention messages
  _ Highlight communications failure message
  _ CSE sheet eject
  _ CSE preserve page position
  _ Use Line-Mode Migration LINECT
  Above the bar storage. . . . . _  1. None  2. Inline resources
  Check CSE fit. . . . . . . . . _  1. No  2. First  3. All
  CSE orientation. . . . . . . . _  1. Portrait  2. Landscape
  Eject to front facing. . . . . _  1. None  2. Job  3. Document  4. Both
  Issue setup messages . . . . . _  1. None  2. Burst  3. Forms  4. All
  Offset stacking. . . . . . . . _  1. None  2. Data set  3. Job
  Auxiliary files MO:DCA level . 1  1. None  2. IS/3  3. BPF match
  Default process mode . . . . . _____
  Resolution . . . . . . . . . . ___   (240, 300)
  Resource deletion ratio. . . . ___   (0–100)

Resources:
  Form definition. . _____
  Page definition. . _____
  Character sets . . ____ ____ ____ ____
  Color map. . . . . _____
  Com setup member . _____
  Overlay. . . . . . _____
  / Prune double-byte fonts
  / Prune single-byte fonts
  _ Map to outline fonts
  _ Recover from font not found
  _ Send default character

Retained Resource Counts:
  Form definitions . _____
  Page definitions . _____
  Page segments. . . _____
  Fonts. . . . . . . _____
  Object containers. _____

Input Tray Substitution:
  Source tray:  Substitute trays:
                Simplex  Duplex

  ---           ---      ---
  ---           ---      ---
  ---           ---      ---
  ---           ---      ---

 Error Reporting Values:
  _ Error disposition supported
  _ Send msgs to SYSOUT
  Print error messages . . 2  1. No  2. Yes
      Maximum messages. . 16_   (0-999)
  Print error reporting. . 1  1. NONE  2. All 3. Character  4. Position

Separator page:
  / Interrupt message page
      _ Mark page
      _ Offset page
      Copies . . __  (1-10)
```

Figure 34. Sample ISPF Printer Inventory panel for defining a TCP/IP-attached FSA (Page 1 of 2)

```
   Connection:
     _ TCPIP 64K Buffer Support
     Connect interval . .  600     (0-86400 seconds)
     Failure action. . . . _  1. Stop  2. Wait for connect
     Disconnect action . . 2 1. Stop  2. Redrive
     Disconnect interval . _____    (0-86400)
     Management mode . . . 3 1. Immediate  2. Dial in  3. Output available
     No response action. . _  1. Notify JES      2. Notify user
                             3. Notify operator  4. Terminate
         Notify . . . . . . _____
     Response timeout. . . _____    (0-20864)
     Printer IP address. . _____
     Port number . . . . . 5001

   Printer Sharing:
     Release mode . . . . 3 1. Idle  2. Time  3. None
     Release interval . . _____   (0-86400 seconds)
     Acquire interval . . _____   (0-86400 seconds)

   Security Labeling:
     Label data pages. . . . . _  1. Yes  2. No
     Label separator pages . . _  1. Yes  2. No
     Restrict printable area . _  1. Yes  2. No

   Debugging:
     Dump:
       Code . . _____   Message ID . . _____   Count . . 1_   (1-99)
     Trace:
       _ Trace user data
         Trace mode . . . . 2  1. None  2. Internal  3. Sync
                              4. Full  5. Limit     6. IPDS
         Trace table size . 32_   (1-999)
         FSA trace dsname . _____

   3800 Compatibility:
     _ Override default font
     Set media origin to 3800 origin for:
       _ Data set
       _ Data set header
       _ Job header
       _ Job trailer
       _ Message data set
```

Figure 35. Sample ISPF Printer Inventory panel for defining a TCP/IP-attached FSA (Page 2 of 2)

Table 7 on page 66 describes the Functional Subsystem Application (FSA) parameters that you can define in the Printer Inventory and compares them to the parameters you can define on the PRINTDEV statement or the EXEC PARM statement of the PSF startup procedure, in PSF installation Exit 7, or in the AFPPARMS control statement. All parameters are optional unless specified otherwise. When you are defining parameters on the ISPF panels, see the online help for information about each parameter. For complete information about defining parameters in the Printer Inventory, see *z/OS Infoprint Server Customization* or *z/OS Infoprint Server Printer Inventory for PSF*.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description

Above the bar storage
 	
Values:
  1 = None
  2 = Inline resources
	above-the-bar-storage
	
	Values:
	  1 = none
	  2 = inline-resources
		above-the-bar-storage	Specifies whether PSF uses storage that is above the bar for processing inline resources.
			
			Values:
			  None: PSF stores inline resources above the 16 MB line but below the 2GB bar (the region area) during job processing.
			
			  `Inline resources`: PSF stores inline resources above the 2GB bar during job processing.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description

Acknowledgement level
 	
Values:
    1 = Page
    2 = Sheet
	acknowledgement-level
	
	Values:
	    page
	    sheet
		XTP7APSF	Indicates whether PSF requests an acknowledgment every sheet or every page.
			
			Values:
			    Page: An acknowledgment is requested for every page that is printed (default).
			    Sheet: An acknowledgment is requested for every sheet that is printed.
			
			Note: For channel-attached printers only.
			
			See "ACK per sheet" on page 196 and "XTP7APSF" on page 436 for more information about this parameter.

Acquire interval	printer-acquire-interval	ACQINTV	Specifies the number of seconds between the time PSF releases a printer and when PSF tries to acquire it again. The **Acquire interval** is only used when **Release mode=Time**. **Values:** 0–86400 The default is 0, which indicates that PSF tries to acquire the printer immediately.
Applid	applid	APPLID	Specifies the 1–8 character name of the VTAM application-program node for an FSA when PSF is printing in deferred-printing mode. Each PSF FSA must have a unique name in the VTAM network. This name must match these two values in the VTAM printer-network definition: • The name of the APPL statement in the PSF node definition for the application program • The value for the LOGAPPL parameter in the node definition for the printer For more information see "Defining a PSF application node" on page 22 and "Defining switched major nodes" on page 27. **Note:** Required for SNA-attached printers only.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description
Auxiliary files MO:DCA level **Values:** 　1 = None 　2 = IS/3 　3 = BPF match	auxiliary-files-modca-level **Values:** 　none 　is3 　bpf-match	auxiliary-files-modca-level	Specifies the MO:DCA interchange set level that auxiliary pages, such as separator pages and message data sets, support. **Values:** 　None: Auxiliary pages do not support a MO:DCA IS level (default). 　IS/3: Auxiliary pages are MO:DCA IS/3 compliant. 　BPF match: Auxiliary files contain the same BPF MO:DCA Interchange Set triplet and Function Set triplets that are specified on the BPF structured field in the user's data set. **Note:** Make sure that changes are made to the PRINTDEV statement for this FSA so that auxiliary pages are generated correctly. See "Setting up MO:DCA interchange set auxiliary pages" on page 116.
Blank compression **Values:** 　/ = Yes 　Blank = No	blank-compression **Values:** 　yes 　no	COMPRESS	Specifies whether PSF compresses data (blanks) in line data that is sent to printers. **Values:** 　Yes: Causes PSF to use an IPDS command to compress contiguous blanks. (The printer extracts the data.) 　No: PSF does not compress blanks (default). **Notes:** 1. This parameter probably does not improve data transmission for channel-attached printers. 2. Blank compression is a data-compression function in PSF. It reduces the amount of data that must be sent through the attachment (host-connected channel, SNA, or TCP/IP) to the printer. 3. PSF compresses any string of more than five contiguous blanks within line data with an IPDS control string. The printer replaces the IPDS control string with the correct number of blanks. 4. In certain conditions, this support might improve the throughput of SNA-attached or TCP/IP-attached printers. For more information, see "Blank compression for line data" on page 196.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description

Capture inline resources
Values:
    / = Yes
    Blank = No
	capture-inline-resources
	Values:
	    yes
	    no
		XTP7CINR	Indicates whether PSF tells a connected DPF or WPM to capture and store inline resources. The default is No, which indicates that DPF or WPM does not capture inline resources.
			
			⚠️ **Attention:** Allowing DPF or WPM to capture inline resources might result in unexpected output if a production job calls out the same resource name that someone modified and put inline in a previous job. Capturing inline resources causes them to be accessible beyond the data set with which they were sent and thus might produce undesirable results in future jobs.
			
			See "XTP7CINR" on page 436 for more information about setting this parameter with Exit 7.

Channel buffer count	channel-buffer-count	BUFNO	Specifies the number of 32 KB I/O buffers that are needed for processing jobs on a channel-attached printer. This value is multiplied by 32 KB (32768 bytes) to determine the total amount of fixed I/O buffer storage. **Values:** 1–10000 **Notes:** 1. Only for channel-attached printers that do not use RPM, DPF, WPM, or PSF direct. 2. For more information about I/O buffers, including recommended values, see "Buffers" on page 197.
Character sets	chars	CHARS	Specifies the last four characters of the coded font name that is used to print a data set on the printer. A font name must contain one to four alphanumeric or national characters; up to four default font names can be specified. For information about selecting fonts, see "How PSF selects resources" on page 174.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description

Check CSE fit
 	
Values:
  1 = No
  2 = First
  3 = All
	cse-check-fit
	
	Values:
	  no
	  first
	  all
		XTP7ACSE	Indicates how PSF checks the pages for cut-sheet emulation (CSE) mode.
			
			Values:
			  No: PSF does not check to see whether the page fits 2 up on the sheet (default).
			  `First`: PSF checks only the first page that is printed for a new copy group to see whether it fits 2 up on the sheet.
			  `All`: PSF checks front side of all pages to see whether they fit 2 up on the sheet.
			
			Note: If a page is checked and found not to fit 2 up on the sheet, the page is printed in normal 1_UP.
			
			See Chapter 15, "Using cut-sheet emulation mode," on page 207 and "XTP7ACSE" on page 430 for more information about this parameter.

Close libraries when idle
 	
Values:
  / = Yes
  Blank = No
	close-libraries-when-idle
	
	Values:
	  yes
	  no
		XTP7CLOS	Indicates whether PSF closes the resource libraries when the printer is idle for 60 seconds. The disconnect interval timer must be set to greater than 60 seconds or PSF does not close the resource libraries. Instead, PSF ends the session with the printer before it closes the resource libraries. The default is No.
			
			Closing the resource libraries causes the operating system to free the fixed storage below the 16 MB line that is required for I/O to the resource libraries. However, this function does add processor usage by closing and reopening the libraries between print activity. Closing the resource libraries is recommended only if you have many printers that are inactive much of the time.
			
			See "XTP7CLOS" on page 436 for more information about setting this parameter with Exit 7.

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description
Color map	color-map	COLORMAP	Specifies the member name (one to eight alphanumeric or national characters) of the printer default color mapping table. The color mapping table is an object container resource that is used to tell the printer what colors to use for various identifiable parts of the data stream. IBM recommends a prefix of M1 for color mapping table resources. The full member name must be specified because PSF does not add the prefix; for example, M1SYSTEM.
			PSF uses a hardcoded color mapping table that is named M1RESET to reset all color mapping table specifications in the printer when these conditions are in place:
			• PSF is attached to a device that supports the color mapping table object container.
			• The COLORMAP parameter is not specified in the OUTPUT JCL statement for the submitted job (see *PSF for z/OS: User's Guide* for more information about the OUTPUT JCL statement).
			• The Printer Inventory or the PRINTDEV JCL statement does not contain a **Color map** parameter.
			For printers not supporting color mapping table object container resources, this parameter is ignored.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)			
Printer Inventory: ISPF Panel	**Printer Inventory: PIDU**	**Equivalent Parameter**	**Description**
Com setup member	com-setup-member	COMSETUP	Specifies the member name (one to eight alphanumeric or national characters) of the printer default microfilm setup resource object container that is generated with the utility that is provided with your microfilm device. The **Com setup member** parameter is used only when output is being sent to a microfilm device. IBM recommends a prefix of H1 for microfilm setup resources, and the full member name must be specified because PSF does not add a prefix; for example, COMSETUP=H1SETUPS.
			When PSF is attached to a microfilm device, if the **Com setup member** parameter is missing from an AFP job that is submitted to that microfilm device, or if no member name is specified or provided in the Printer Inventory or the PRINTDEV, PSF ends the job. For more information about printing to microfilm, see Chapter 29, "Supporting AFP output on microfilm," on page 341.
			Note: Required for microfilm devices only; for non-microfilm devices, this parameter is ignored.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)			
Printer Inventory: ISPF Panel	**Printer Inventory: PIDU**	**Equivalent Parameter**	**Description**
Connect interval	printer-connect-interval	CONNINTV	Specifies the number of seconds during which PSF attempts to start a session or a dialog with a printer. When the connect interval expires and the connection is not complete, PSF issues a message and stops the FSA. For all attachments, if **Connect interval=0**, PSF attempts to connect for an unlimited time.
			For SNA, the connect interval is set when a session request is queued after a busy or not available condition occurs. If a VTAM condition occurs, such as LU NOT ACTIVE, PSF retries 16 times and then stops processing without using the connect interval.
			For TCP/IP, the connect interval is set when a printer connection is attempted (CONNECT, OPEN SESSION, ASSIGN).
			For IPDS, the connect interval is set when a dialog is restarted after a release (**Release mode=Idle|Time**).
			Values: 0–86400. The default values are:
			0 Attempt connection for an unlimited time; all attachments except TCP/IP.
			600 Attempt connection for 10 minutes; TCP/IP attachments only.

Consolidate IM1 images
Values:
    / = Yes
    Blank = No
	consolidate-im1-images
	Values:
	    yes
	    no
		XTP7C2SI	Indicates whether PSF consolidates a multiple-celled IM1 image into a single IOCA image. The default is No, which indicates that PSF converts a multiple-celled IM1 image to multiple IOCA images.

			This request is accepted only if the printer does not support IOCA replicate and trim. This change improves performance only if the image is a 240-pel IM1 celled image that requests replication. For any other case, this change might degrade performance.
			See "XTP7C2SI" on page 432 for more information about setting this parameter with Exit 7.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description

CSE orientation
Values:
    1 = Portrait
    2 = Landscape
	cse-orientation
	Values:
	    portrait
	    landscape
		XTP7WIDE	Indicates whether PSF generates portrait or landscape pages for printing in cut-sheet emulation (CSE) mode. The default is `Portrait`. PSF uses this parameter only if one of the 3800 compatibility media origin parameters is specified: **Data set**, **Data set header**, **Job header**, **Job trailer**, **Message data set**.
			
			See "Media origin" on page 335 and "XTP7WIDE" on page 435 for more information about this parameter.

CSE preserve page position
Values:
    / = Yes
    Blank = No
	cse-preserve-page-position
	Values:
	    yes
	    no
		XTP7CPPP	Indicates whether PSF preserves page placement after an error recovery reposition or operator command. This parameter applies only when PSF is printing in cut-sheet emulation (CSE) mode.
			
			If you specify Yes to preserve page placement, you must also specify Yes for the **CSE sheet eject** parameter; otherwise, you get an error message from the Printer Inventory. The default is No.
			
			See "Things to consider when using cut-sheet emulation" on page 208 and "XTP7CPPP" on page 438 for more information about setting this parameter with Exit 7.

CSE sheet eject
Values:
    / = Yes
    Blank = No
	cse-sheet-eject
	Values:
	    yes
	    no
		XTP7NSF	Specifies whether PSF tells the printer to eject to a new sheet on a data set boundary when cut-sheet emulation (CSE) is done. The default is No.
			
			See "Things to consider when using cut-sheet emulation" on page 208 and "XTP7NSF" on page 436 for more information about this parameter.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)			
Printer Inventory: ISPF Panel	**Printer Inventory: PIDU**	**Equivalent Parameter**	**Description**
Default process mode	default-process-mode	PRMODE XTP7PRMD	Indicates the default processing mode that PSF uses to print data sets containing both single-byte and double-byte fonts.
			Values: PSF ignores all values but these:
			SOSI1 Each shift-out, shift-in code is converted to a blank and a Set Coded Font Local text control.
			SOSI2 Each shift-out, shift-in code is converted to a Set Coded Font Local text control.
			SOSI3 The shift-in code is converted to a Set Coded Font Local text control and two blanks. A shift-out code is converted to a Set Coded Font Local text control.
			SOSI4 Each shift-out, shift-in code is to be skipped and not counted when offsets are calculated for the print data set. SOSI4 is used when double-byte character set (DBCS) text is converted from ASCII to EBCDIC. When SOSI4 is specified, the page definition offsets are correct after conversion; therefore, the user does not need to account for SOSI characters when FIELD offsets are computed. The data conversion that PSF makes for SOSI4 is the same as for SOSI2.
			Notes:
			1. PSF uses this parameter only if the PRMODE keyword is not specified on the OUTPUT JCL statement.
			2. If this parameter is not specified in the Printer Inventory, Exit 7, or the PRINTDEV statement, and the PRMODE keyword is not specified on the OUTPUT JCL statement, PSF defaults to either line data or MO:DCA, depending on the type of data stream.
			See PRMODE or "XTP7PRMD" on page 437 for more information about setting this parameter in the PRINTDEV statement or with Exit 7.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description

Disconnect action
 	
Values:
    1 = Stop
    2 = Redrive
	disconnect-action
	
	Values:
	    stop
	    redrive
		TIMEOUT	Specifies the action that PSF takes when the time that is specified by the **Disconnect interval** parameter expires and no output is available from JES.
			
			Values:
			    Stop: PSF stops the printer FSA, which can then be restarted only by an operator command.
			    Redrive: PSF redrives the printer FSA according to the value specified by the **Management mode** parameter (default).
			
			Note: Only for SNA-attached and TCP/IP-attached printers.

Disconnect interval	printer-disconnect-interval	DISCINTV	Specifies the number of seconds until PSF ends the session with the printer. When no output is available from JES for this time period, PSF issues a message and ends the session with the printer.
			
			For cut-sheet printers, PSF forces all pages from the last job to reach the stacker before it ends the session and allows that job to be purged from the spool. For continuous-forms printers, the value of the JES NPRO parameter determines whether PSF forces all pages from the last job to the stacker before it ends the session:
			
			NPRO > 0
			    PSF forces all pages to the stacker.
			
			NPRO = 0
			    PSF does not force all the pages to the stacker, but issues a message and ends the session. To force all pages to the stacker, send another print job to the printer or stop the PSF FSA.
			
			Values: 0–86400
			
			The default is 0.
			
			If the **Disconnect interval = 0**, PSF does not end the session with the printer when output is not available. (See Note "2" on page 77 for additional information.)
			
			If the **Release mode = Idle|Time**, the Disconnect interval is ignored. For more information, see "Selecting PSF parameters for shared printers" on page 118.
			
			See Notes.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)			
Printer Inventory: ISPF Panel	**Printer Inventory: PIDU**	**Equivalent Parameter**	**Description**
			Notes: 1. For SNA-attached and TCP/IP-attached printers only. TCP/IP attachment includes printers that are attached with the i-data 7193 IPDS Printer LAN Attachment (see Appendix M, "Installing the i-data 7913 IPDS Printer LAN Attachment," on page 473). 2. Some printers have a network inactivity timer. If the timer is enabled, the Disconnect interval value must be less than the value of the inactivity timer and greater than 0. Otherwise, a PSF abend occurs when the inactivity timer expires. The inactivity timer can be set with the printer-configuration procedure. 3. When some printers, such as the InfoPrint 60, become inoperative, a *local intervention timer* is started. When the timer expires, an action code X'22' negative acknowledgment reply (NACK) is sent to PSF. PSF then issues message APS923I and does the action that is specified in the **Failure action** parameter. If the system operator does not need this message, you can disable the timer. PSF then waits for the printer operator to correct the problem.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)			
Printer Inventory: ISPF Panel	**Printer Inventory: PIDU**	**Equivalent Parameter**	**Description**

Dump: Code	dump-code	DUMP	Identifies a reason code that causes a conditional memory dump of the PSF address space when the reason code occurs. The reason code can be:
			
			• A seven-character hexadecimal abend reason code for an abend that PSF does not normally create a memory dump, such as a restartable abend; the first three characters of an abend reason code are always ABD. For example:
			
			`ABD011C`
			
			See *PSF for z/OS: Messages and Codes* for a list of abend reason codes.
			
			• An eight-character hexadecimal PSF reason code; you can enter the hexadecimal characters only or the hexadecimal characters with a prefix of 0x. For example:
			
			`09600c00`
			`0x09600c00`
			
			See *PSF for z/OS: Diagnosis* for a list of PSF reason codes.
			
			• An integer 0–2147483647. For example:
			
			`157289480`
			
			Notes:
			
			1. If both a reason code and a message ID are specified, a dump occurs at the first occurrence of either one.
			
			2. For an explanation of restartable abends, see *PSF for z/OS: Diagnosis*.

Dump: Count	message-count-before-dump	DUMP	Specifies the number of times the message that is specified by the **Dump: Message ID** parameter is issued before PSF produces a conditional memory dump. **Values:** 1–99 The default is 1.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)			
Printer Inventory: ISPF Panel	**Printer Inventory: PIDU**	**Equivalent Parameter**	**Description**
Dump: Message ID	dump-message-id	DUMP	Identifies a PSF message that causes a conditional memory dump of the PSF address space when the message occurs. The dump occurs after the message is issued for the number of times that are specified by the **Dump: Count** parameter. **Syntax:** APS*nnnnt* ***nnnn*** 　Three to four-digit message number. ***t*** 　One of these type codes: 　**A** 　　Message requiring operator action. 　**I** 　　Information message. **Examples:** APS896I, APS2001A **Note:** If both a reason code and a message ID are specified, a dump occurs at the first occurrence of either one.
Eject to front facing **Values:** 　1 = None 　2 = Job 　3 = Document 　4 = Both	eject-to-front-facing **Values:** 　none 　job 　document 　both	XTP7ETFF XTP7EFFH	Indicates whether PSF is to tell your continuous-forms printer to do an eject to front facing before the job-header page, before the start of a new document, or both. **Values:** 　None: Eject to front facing is not done. 　Job: Eject to front facing is done before the job-header page. 　Document: Eject to front facing is done between documents in a data set. 　Both: Eject to front facing is done before the job-header page and between documents (default). See "XTP7ETFF" on page 431 and "XTP7EFFH" on page 431 for more information about setting this parameter with Exit 7.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description

End SNA conversation
Values:
   / = Yes
   Blank = No
	end-sna-conversation
	Values:
	   yes
	   no
		XTP7ENDC	Indicates whether PSF ends the SNA LU1 conversation between print jobs while it maintains the SNA session with the printer when the NPRO timer expires or after no job is available for one minute and the last page printed is stacked. If PSF ends the SNA LU1 conversation with the printer while it maintains its SNA session, the printer can accept a request for a print screen without deleting the resources that PSF loaded into the printer. The default is No, which indicates that PSF maintains the SNA LU1 conversation between print jobs.
			
			See "XTP7ENDC" on page 435 for more information about setting this parameter with Exit 7.

Error disposition supported
Values:
   / = Yes
   Blank = No
	error-disposition-supported
	Values:
	   yes
	   no
		PRTERROR	Indicates whether PSF accepts the error disposition that is specified on the OUTPUT JCL statement when PSF stops processing a data set because an error occurs during printing. The default is No.
			
			Note: The Printer Inventory values are equivalent to these PRTERROR values in the PRINTDEV statement:
			
			   Yes = HONOR
			   No = NOTHONOR

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description

Failure action
Values:
   1 = Stop
   2 = Wait for connect
	failure-action
	Values:
	   stop
	   connect
		FAILURE	Specifies PSF action after a printer failure, an SNA session failure, or an Internet Protocol network failure.
			
			If **Failure action=Stop**, PSF must be restarted by an operator command. If Failure action is the default: `Wait for connect` (ISPF) or `connect` (PIDU), and the printer is connected to another host when PSF attempts to establish connection, PSF waits for the printer in one of these ways:
			
			• For SNA, PSF queues a session request in VTAM, which responds when the printer becomes available. PSF waits up to the time specified by the **Connect interval** parameter.
			• For TCP/IP, PSF continuously retries until the printer becomes available. PSF retries up to the time specified by the **Connect interval** parameter.
			
			Notes:
			
			1. `Wait for connect` or `connect` minimizes the amount of data that needs to be reprinted when communication with the printer is restored after a failure condition.
			2. `Stop` ensures that the operator is aware that a failure condition occurred and that PSF must be restarted; printing restarts from the last checkpoint.

Fonts	retained-fonts	XTP7RCF	Specifies the maximum number of fonts that PSF retains in printer storage between print jobs. **Values:** 0–127 The default value depends on the type of printer and the amount of storage available in the printer. **Note:** When PSF retains fonts, PSF does not need to reload the same fonts for subsequent jobs. However, retaining fonts requires more printer storage. See "XTP7RCF" on page 433 for more information about this parameter.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description
Form definition (required)	form-definition	FORMDEF	Specifies a required 1–8 character default form definition. Form definition names in PSF must begin with an F1 prefix or PSF adds the prefix. If the name is three or more characters and the first two characters are the F1 prefix, PSF uses the name as is. Otherwise, if the name does not begin with the F1 prefix and is a maximum of six characters, PSF adds the F1 prefix to the name. However, if the form definition name begins with F1, which is part of the name and not the prefix, and is a maximum of six characters, you must include the prefix in the form definition name. For example, if the form definition name is F1A20, you must code F1F1A20 as the form definition. For information about how PSF selects a form definition, see "How PSF selects resources" on page 174.
Form definitions	retained-form-definitions	XTP7RFD	Specifies the maximum number of form definitions that PSF retains in virtual storage between print jobs. **Values:** 0–32767 The default is 6. **Note:** When PSF retains form definitions, PSF does not need to reload the same form definitions for subsequent jobs. However, retaining form definitions requires more virtual storage. See "XTP7RFD" on page 433 for more information about this parameter.
FSA trace dsname	fsa-trace-dsname	*ddname*	Specifies the data set that PSF directs an FSA trace to when the trace mode is Sync, Full, Limit, or IPDS. The trace data set must exist and be cataloged before the PSF FSA is started. For more information about using traces, see *PSF for z/OS: Diagnosis*.

Highlight communications failure message
Values:
  / = Yes
  Blank = No
	highlight-communications-failure-message
	Values:
	  yes
	  no
		XTP7HLCF	Indicates whether PSF highlights communications failure message APS6501I. The highlighted message remains until deleted. The default is No.
			
			Note: Only for TCP/IP-attached printers.
			
			See "XTP7HLCF" on page 438 for more information about this parameter.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description

Inhibit recovery
 	
Values:
  / = Yes
  Blank = No
	inhibit-recovery
	
	Values:
	  yes
	  no
		XTP7IHIB	Indicates whether normal PSF error recovery is to be inhibited. The default is No. See "Inhibiting PSF recovery" on page 326 for more information about inhibiting recovery.
			
			Notes:
			1. Because blank pages can occur between jobs on a continuous forms printer, inhibit error recovery only if you are using a cut-sheet printer.
			2. If an error occurs that requires recovery for pages that are already printed, such as a paper jam, the job is ended.
			
			See "XTP7IHIB" on page 435 for more information about setting this parameter with Exit 7.

Input Tray Substitution: Substitute trays: Duplex	tray-substitutions
	
	Syntax:
	  {*source* {*subsim subdup*}}
	
	  source = Source tray value
	  subsim = Simplex substitute tray value
	  subdup = Duplex substitute tray value
		XTP7DUPA
		XTP7DUPB
		XTP7DUPC
		XTP7DUPD
		XTP7SENS
			Specifies the mappings for bins for Side and Edge Sensitive Paper Handling.
			
			Values: 1–255
			
			Note: See Chapter 16, "Using Side and Edge Sensitive Paper Handling," on page 209 for more information about the Side and Edge Sensitive Paper Handling function and how to use this parameter.

Input Tray Substitution: Substitute trays: Simplex	tray-substitutions
	
	Syntax:
	  {*source* {*subsim subdup*}}
	
	  source = Source tray value
	  subsim = Simplex substitute tray value
	  subdup = Duplex substitute tray value
		XTP7SIMA
		XTP7SIMB
		XTP7SIMC
		XTP7SIMD
		XTP7SENS
			Specifies the mappings for bins for Side and Edge Sensitive Paper Handling.
			
			Values: 1–255
			
			Note: See Chapter 16, "Using Side and Edge Sensitive Paper Handling," on page 209 for more information about the Side and Edge Sensitive Paper Handling function and how to use this parameter.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description
Input Tray Substitution: Source tray	tray-substitutions **Syntax:** {*source* {*subsim subdup*}} *source* = Source tray value *subsim* = Simplex substitute tray value *subdup* = Duplex substitute tray value	XTP7BINA XTP7BINB XTP7BINC XTP7BIND XTP7SENS	Specifies the mappings for bins for Side and Edge Sensitive Paper Handling. **Values:** 1–255 **Note:** See Chapter 16, "Using Side and Edge Sensitive Paper Handling," on page 209 for more information about the Side and Edge Sensitive Paper Handling function and how to use this parameter.
Interrupt message page **Values:** / = Yes Blank = No	interrupt-message-page **Values:** print suppress	XTP7IPSU	Indicates whether the interrupt message page that PSF inserts in your printed output is printed. The default is Yes or `Print`. **Notes:** 1. Suppressing interrupt message pages is suggested when PSF is attached to a printer with Finisher capability; otherwise, interrupt message pages separate finished documents into two finished groups with the inserted page in the middle. 2. This parameter has no effect when microfilm is printed because PSF automatically suppresses the printing of interrupt message pages. See "Interrupt message page" on page 329 and "XTP7IPSU" on page 432 for more information about this parameter.
Interrupt message page: Copies	interrupt-message-page-copies	XTP7IPCC	Specifies the number of interrupt message page copies that PSF produces when the **Interrupt message page: Mark page** parameter is specified. **Values:** 1–10 The default is 1. **Note:** Copies are ignored if the device type is 3800. See "Interrupt message page" on page 329 and "XTP7IPCC" on page 430 for more information about this parameter.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description
Interrupt message page: Mark page **Values:** / = Yes Blank = No	mark-interrupt-message-page **Values:** yes no	XTP7IPMF	Indicates whether PSF marks the interrupt message page with form marks. The default is No. See "Interrupt message page" on page 329 and "XTP7IPMF" on page 431 for more information about this parameter.
Interrupt message page: Offset page **Values:** / = Yes Blank = No	offset-interrupt-message-page **Values:** yes no	XTP7IPOS	Indicates whether offset stacking is required for the interrupt message page. The default is No. See "Interrupt message page" on page 329 and "XTP7IPOS" on page 432 for more information about this parameter.
Issue intervention messages **Values:** / = Yes Blank = No	issue-intervention-messages **Values:** yes no	INTRVMSG XTP7IMSG	Indicates whether PSF issues a message to the operator console when the printer is SNA- or TCP/IP-attached and has a condition that requires intervention, such as a paper jam. See INTRVMSG and "XTP7IMSG" on page 438 for more information about this parameter.
Issue setup messages **Values:** 1 = None 2 = Burst 3 = Forms 4 = All	issue-setup-messages **Values:** none burst forms all	SETUP	Specifies the setup parameters for which JES issues a setup message when the printer is initialized and at the start of any job that specifies a change in a setup parameter from what is active for the printer. The default is None. **Note:** Only for SNA-attached or TCP/IP-attached printers. Setup messages are always issued for local channel-attached printers.
Label data pages **Values:** 1 = Yes 2 = No	label-data-pages **Values:** yes no	DPAGELBL	Indicates whether the security label is printed on each page of printed output. **Note:** A blank indicates that PSF sets the value to Yes if PSFMPL is active (default) and No if PSFMPL is not active. See *PSF for z/OS: Security Guide* for more information about this parameter.
Label separator pages **Values:** 1 = Yes 2 = No	label-separator-pages **Values:** yes no	SPAGELBL	Indicates whether the security label is printed on a separator page. **Note:** A blank indicates that PSF sets the value to Yes if PSFMPL is active (default) and No if PSFMPL is not active. See *PSF for z/OS: Security Guide* for more information about this parameter.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)			
Printer Inventory: ISPF Panel	**Printer Inventory: PIDU**	**Equivalent Parameter**	**Description**

Logmode	logmode	LOGMODE	Specifies the 1–8 character name of the VTAM logon-mode table entry, which defines the session parameters for an SNA-attached printer. (See "Defining the logon-mode table entry" on page 22 for more information.)
			
			Notes:
			
			1. Only for SNA-attached printers.
			
			2. Although Logmode is an optional parameter, it must not be used if the printer is in a cross-domain system and if the Printer Inventory or the PRINTDEV statement specifies **Management mode=Dial in**.
			
			3. If Logmode is used, the name must conform to standard JCL naming conventions.
			
			4. If Logmode is omitted or the name is left blank, session parameters must be specified by the DLOGMOD parameter of the LU statement for the printer in the VTAM network definition, and the name defaults to one of these:
			
			  • The name that is specified by the DLOGMOD parameter.
			  • The name that is passed by the secondary LU in the CINIT, if that LU is in another domain.

LU name	luname	LUNAME	Specifies the unique, 1–8 character, logical-unit name of an SNA-attached printer. The name must conform to standard JCL naming conventions. The name must also match the printer-node name in the LU statement of the VTAM network definition. For additional information, see "Defining switched major nodes" on page 27. **Note:** Required for SNA-attached printers only.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description
Management mode **Values:** 1 = Immediate 2 = Dial in 3 = Output available	printer-management-mode **Values:** immediate dialin outavail	MGMTMODE	Specifies how PSF controls an SNA-attached or TCP/IP-attached printer in deferred-printing mode. **Values:** `Immediate`: PSF starts a communication session with the printer immediately and then looks for output available on the JES spool. `Dial in`: PSF starts a session with the printer when the switched line is connected. LOGAPPL must be applied in the PSF network definition. Recommended for some printers that use a switched line. For more information about Dial in, see "Considerations for switched lines" on page 38. For TCP/IP-attached printers, Dial in is identical to Immediate. `Output available`: PSF starts a communication session with the printer only when output is available on the JES spool (default). This parameter can also be specified for same-domain leased (nonswitched) lines. PSF waits for LOGAPPL to be triggered; for example, by the printer LU becoming active. **Note:** Only for SNA-attached and TCP/IP-attached printers.
Map to outline fonts **Values:** / = Yes Blank = No	map-to-outline-fonts **Values:** yes no	MAP2OLN XTP7MTOF	Indicates whether PSF maps fonts to outline fonts. Use this parameter if your printer supports outline fonts, you have existing applications that use raster fonts, and you want to use outline fonts without changing the applications. The default is No. See MAP2OLN, "Mapping to outline fonts" on page 236, and "XTP7MTOF" on page 432 for more information about this parameter.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)			
Printer Inventory: ISPF Panel	**Printer Inventory: PIDU**	**Equivalent Parameter**	**Description**

No response action
Values:
   1 = Notify JES
   2 = Notify user
   3 = Notify operator
   4 = Terminate
	no-response-action
	Values:
	   notify-jes
	   notify-user
	   notify-operator
	   terminate
		NORESP	Specifies what action PSF takes when the time that is specified by the **Response timeout** parameter expires and a response is not received from the printer.
			Values:
			`Notify JES`: PSF notifies JES that an expected response was not received from the printer. This condition is also displayed by SDSF (default).
			`Notify user`: PSF sends a message to the user ID specified by the **No response action: Notify** parameter and to JES indicating that an expected response was not received from the printer.
			`Notify operator`: PSF sends a message to the system operator and to JES indicating that an expected response was not received from the printer.
			`Terminate`: PSF stops the printer FSA. The system operator must issue a command to restart the FSA. The data set active when the FSA is stopped is restarted from the last checkpoint.
			Notes:
			1. Only for SNA-attached and TCP/IP-attached printers.
			2. When the **No response action** parameter is specified, the **Response timeout** parameter must also be specified.

No response action: Notify	no-response-notify	NORESP	Specifies the user ID that PSF sends a message to when an expected response is not received from the printer before time expires. This parameter is used when `Notify user` is specified by the **No response action** parameter.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)			
Printer Inventory: ISPF Panel	**Printer Inventory: PIDU**	**Equivalent Parameter**	**Description**
Object containers	retained-object-containers	XTP7ROC	Specifies the maximum number of object containers that PSF retains in printer storage between print jobs. **Values:** 0–32767 The default is 0 for 3800 and 3820 printers. The default is 200 for all other printers. **Note:** For 3800 and 3820 printers, PSF does not retain object containers in printer storage; therefore, PSF ignores a value greater than 0. See "XTP7ROC" on page 433 for more information about this parameter.

Offset stacking
Values:
  1 = None
  2 = Data set
  3 = Job
	offset-stacking
	Values:
	  none
	  dataset
	  job
		XTP7JOG	Specifies whether offset stacking is done and if so, whether it is done on a data set boundary or a job boundary. If this parameter is not specified, PSF uses the JES COPYMARK parameter to control offset stacking.
			
			Values:
			  None: No offset stacking is done.
			  `Data set`: Offset stacking is done on a data set boundary.
			  Job: Offset stacking is done on a job boundary.
			
			See "Separating copy marking and offset stacking" on page 158 and "XTP7JOG" on page 435 for more information about this parameter.

Overlay	global-overlay	OVERLAY	Specifies the member name of a medium overlay to be placed on every sheet of output on the printer, including separator pages and message pages. The name is one to eight alphanumeric or national characters; the first character must be alphabetic or national. **Notes:** 1. PSF does not prefix the name given with O1. 2. This overlay is always loaded from the system library. If multiple system overlay libraries are defined for different resolutions, the overlay must be placed in each library concatenation list.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)			
Printer Inventory: ISPF Panel	**Printer Inventory: PIDU**	**Equivalent Parameter**	**Description**
Page definition (required)	page-definition	PAGEDEF	Specifies a required 1–8 character default page definition. Page definition names in PSF must begin with a P1 prefix or PSF adds the prefix. If the name is three or more characters and the first two characters are the P1 prefix, PSF uses the name as is. Otherwise, if the name does not begin with the P1 prefix and is a maximum of six characters, PSF adds the P1 prefix to the name. However, if the page definition name begins with P1, which is part of the name and not the prefix, and is a maximum of six characters, you must include the prefix in the page definition name. For example, if the page definition name is P1011, you must specify P1P1011 as the page definition. For information about how PSF selects a page definition, see "How PSF selects resources" on page 174.
Page definitions	retained-page-definitions	XTP7RPD	Specifies the maximum number of page definitions that PSF retains in virtual storage between print jobs. **Values:** 0–32767 The default is 6. **Note:** When PSF retains page definitions, PSF does not need to reload the same page definitions for subsequent jobs. However, retaining page definitions requires more virtual storage. See "XTP7RPD" on page 433 for more information about this parameter.
Page segments	retained-page-segments	XTP7RPS	Specifies the maximum number of page segments that PSF retains in printer storage between print jobs. **Values:** 0–32767 The default is 0. **Note:** When PSF retains page segments, PSF does not need to reload the same page segments for subsequent jobs. However, retaining page segments requires more printer storage. See "XTP7RPS" on page 433 for more information about this parameter.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)			
Printer Inventory: ISPF Panel	**Printer Inventory: PIDU**	**Equivalent Parameter**	**Description**
Port number	port-number	PORTNO	Specifies the port number with which PSF is to establish a connection. The port number identifies the appropriate internal process in the printer or in the i-data 7913 IPDS Printer LAN Attachment. For TCP/IP-attached printers, the port number must match the TCP/IP port number in the printer configuration. **Values:** 5001–65535 The default port is 5001. **Note:** For TCP/IP-attached printers only.

Printer IP address	printer-ip-address	IPADDR	Identifies the 1–250 character Internet Protocol (IP) address or host name of the TCP/IP-attached printer or of the i-data 7913 IPDS Printer LAN Attachment. Specify one of these:
			
			• Host name; for example, `PRTJDOE.SYSTEM`.
			• Dotted decimal address in the format *nnn.nnn.nnn.nnn*; for example, `192.0.2.33`
			• Colon hexadecimal address in the format: *nnnn:nnnn:nnnn:nnnn:nnnn:nnnn:nnnn:nnnn* For example, `2001:0db8:85a3:0000:0000:8a2e:0c22:384e`. These restrictions apply:
			
			  – Leading zeros in each hexadecimal value can be omitted. For example: `2001:db8:85a3:0:0:8a2e:c22:384e`
			  – One sequence of repeat zero values can be omitted. For example: `2001:db8:85a3::8a2e:c22:384e`
			  – The last two hexadecimal digits can be in dotted decimal notation. For example: `2001:db8:85a3::8a2e:12.34.56.78`
			
			Note: Required for TCP/IP-attached printers only.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)			
Printer Inventory: ISPF Panel	**Printer Inventory: PIDU**	**Equivalent Parameter**	**Description**

Print error messages
Values:
    1 = No
    2 = Yes
	print-error-messages
	Values:
	  no
	  yes
		PIMSG	Indicates whether all message groups generated in the processing of a data set are printed at the end of the data set.
			
			Values
			    No: Specifies that no message groups are to be printed unless an error occurs that forces a premature end to the printing of the data set. If that happens, the message group that describes the error that stopped processing is printed.
			
			    Yes: Specifies that all message groups that are generated in the processing of a data set are to be printed at the end of the data set, preceding any job-trailer separator pages. If a data set is not completed because of an error, message groups that are generated up to this error are printed, including the message group that describes the error that stopped processing (default).
			
			Notes:
			
			1. Reaching the message count value that is specified by the **Print error messages: Maximum message** parameter is one error that forces a premature end to printing.
			
			2. The PIMSG parameter that is specified on the OUTPUT statement in the startup procedure for the message data set, referenced by the MESSAGE keyword, also affects the rerouting of messages. For more information, see "Redirecting the message data set" on page 328.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description
Print error messages: Maximum messages	print-error-messages-maximum	PIMSG	Indicates the maximum number of message groups that are printed when a value is specified for the **Print error messages** parameter. When the maximum number is reached, processing of the data set stops and the data set is purged from the spool. The final count of printed messages might be more than the value for maximum messages if the message groups are generated for errors that are reported by the printer for pages that are sent before the message count is reached.
			Values: 0–999
			A value of 0 means that the data set is processed until complete and all message groups are printed unless an error occurs that stops processing. The default is 16.
			Note: Messages are printed depending on whether Yes or No is specified for the **Print error messages** parameter. For more information, see *PSF for z/OS: User's Guide*.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)			
Printer Inventory: ISPF Panel	**Printer Inventory: PIDU**	**Equivalent Parameter**	**Description**

Print error reporting
 	
Values:
    1 = None
    2 = All
    3 = Character
    4 = Position
	print-error-reporting
	
	Values:
	    none
	    all
	    character
	    position
		DATACK	Specifies whether the printer reports character and position errors to PSF. Character errors are caused by trying to use a code point that is not assigned to a character in a font. Position errors are caused by trying to print outside the valid printable area or off the logical page. Some printers use exception highlighting to mark position errors on the printed page.
			
			Values:
			
			None: Do not report any character or position errors to PSF (default). This value is equivalent to BLOCK in the PRINTDEV DATACK parameter.
			
			All: Report all character and position errors to PSF. This value is equivalent to UNBLOCK in the PRINTDEV DATACK parameter.
			
			Character: Report only character errors to PSF. This value is equivalent to BLKPOS in the PRINTDEV DATACK parameter.
			
			Position: Report only position errors to PSF. This value is equivalent to BLKCHAR in the PRINTDEV DATACK parameter.
			
			Notes:
			
			1. Regardless of what the **Print error reporting** parameter specifies, PSF defaults to All when it prints security resources. For more information about security resources, see *PSF for z/OS: Security Guide*.
			
			2. Unless **Print error messages=No**, Print error reporting also produces error messages.
			
			3. For microfilm devices, see Chapter 29, "Supporting AFP output on microfilm," on page 341.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description
Prune double-byte fonts **Values:** / = Yes Blank = No	prune-double-byte-fonts **Values:** yes no	XTP7PRND	Indicates whether PSF prunes double-byte raster fonts to reduce the amount of font data sent to the printer. Instead of sending all characters in a character set when PSF downloads a font, PSF matches the characters in the character set to the code points in the code page, and then sends only the characters that are referenced in the code page to the printer. Font pruning can save time that is needed to send fonts to a printer and reduce the amount of raster pattern storage that is used in the printer. The default is Yes. See "XTP7PRND" on page 431 for more information about setting this parameter with Exit 7. See "Font pruning of raster fonts" on page 188 for information about the effect of font pruning on printer performance.
Prune single-byte fonts **Values:** / = Yes Blank = No	prune-single-byte-fonts **Values:** yes no	XTP7PRNS	Indicates whether PSF prunes single-byte raster fonts to reduce the amount of font data sent to the printer. Instead of sending all characters in a character set when PSF downloads a font, PSF matches the characters in the character set to the code points in the code page, and then sends only the characters that are referenced in the code page to the printer. Font pruning can save time that is needed to send fonts to a printer and reduce the amount of raster pattern storage that is used in the printer. The default is Yes. See "XTP7PRNS" on page 431 for more information about setting this parameter with Exit 7. See "Font pruning of raster fonts" on page 188 for information about the effect of font pruning on printer performance.
Recover from font not found **Values:** / = Yes Blank = No	recover-from-font-not-found **Values:** yes no	MAP2OLN XTP7RFNF	Indicates whether PSF makes sure the outline font that is derived from the mapped font exists before proceeding. Use this parameter if your printer supports outline fonts, you requested that PSF map to outline fonts, and you do not want pages in your job ended because the outline font identified through the mapped font did not exist on the host. The default is No. See "Recovering from font not found" on page 236, MAP2OLN, and "XTP7RFNF" on page 432 for more information about this parameter.

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description
Release data set when repositioning **Values:** ／ = Yes Blank = No	release-ds-when-repositioning **Values:** yes no	XTP7RDSR	Indicates whether PSF releases spool data sets it obtained from JES when it is repositioning to handle an exception reported by the printer, such as a paper jam or a page adjust. Releasing data sets might reorder the print data sets when printing resumes. The default is No. See "Disposition of spool data sets during repositioning" on page 325 and **XTP7RDSR** in "XTP7 fields" on page 429 for more information about this parameter.
Release interval	printer-release-interval	RELINTV	Specifies the number of seconds after which PSF responds to a request to release a printer in the method that is specified by the **Release mode** parameter. The timer for the release interval is not started until PSF is at the next job boundary. **Note:** If a setup message is issued for a job, PSF considers that the job started and does not release the printer, even if the release timer expired. When the operator responds to the setup message and the job completes on the printer, PSF releases the printer. **Values:** 0–86400 The default is 0, which means that PSF releases the printer immediately.
Release mode **Values:** 1 = Idle 2 = Time 3 = None	printer-release-mode **Values:** idle time none	RELMODE	Specifies how PSF is to respond to a request to release the printer. A request to release a printer can be either an IPDS release dialog request or a VTAM release session request. For more information, see "Selecting PSF parameters for shared printers" on page 118. **Values:** `Idle`: PSF releases the printer when a request to release is received and the time that is specified by the **Release interval** parameter expired with no output on the spool for the printer. `Time`: PSF starts the timer for the release interval when a release request is received, even when more output exists on the spool. `None`: PSF does not release the printer (default).

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description

Resolution	resolution	XTP7FMT	Indicates the resolution at which the output was formatted. PSF uses this value to choose the associated resolution system library that is previously defined by the system programmer. If this parameter is blank, PSF uses the default system library.
			
			Values:
			
			  240
			      The data was formatted with resources at 240 pels per inch.
			
			  300
			      The data was formatted with resources at 300 pels per inch.
			
			Note: The resolution value that is specified is used for all jobs unless you override it with the Exit 7 BDSC or EDSC.
			
			See "Specifying format resolution" on page 217 and "XTP7FMT" on page 435 for more information about this parameter.

Resource deletion ratio	resource-deletion-ratio	resource-deletion-ratio	Specifies a number that determines whether PSF expedites the deletion of all fonts, overlays, object containers, and page segments at the end of data set processing. If the ratio of inline and user resources to the total number of loaded resources (expressed as a percentage) is greater than or equal to this parameter value, PSF expedites resource deletion. If the ratio percentage is less than this parameter value, standard methods are used to delete the resources.
			
			The values are:
			
			1–100
			    PSF expedites resource deletion if the resource ratio percentage is greater than or equal to this value. The default is 90.
			
			0
			    PSF does not use this parameter when it is deleting resources.
			
			For more information, see "Expediting resource deletion" on page 189.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)			
Printer Inventory: ISPF Panel	**Printer Inventory: PIDU**	**Equivalent Parameter**	**Description**
Response timeout	response-timeout	RESPTIME	Specifies the maximum number of seconds PSF waits for a response from the printer. **Values:** 0–86400. The default is 0, which means PSF waits indefinitely for a response. **Notes:** 1. Only for SNA and TCP/IP-attached printers. 2. The **Response timeout** parameter is only effective with TCP/IP after the TCP/IP session is established (CONNECT, ASSIGN, and START SESSION are completed successfully). 3. The **Response timeout** parameter is only effective with SNA when PSF is actually sending IPDS data (this does not include items such as BIND, ATTACH, DETACH, and UNBIND). 4. When the **Response timeout** parameter is specified, the **No response action** parameter must also be specified to tell PSF what action to take when no response is received within the time specified. 5. This timer is active and can time out during the time PSF is waiting for a printer to become ready after an interruption condition at the printer. 6. Response timeout might not be appropriate for faster printers. This function adds some processor usage to start and stop the timers. 7. If the printer has an intervention timer, consider how that timer interacts with this one when you choose the time values for both timers.

Restrict printable area
Values:
    1 = Yes
    2 = No
	restrict-printable-area
	Values:
	    yes
	    no
		SYSAREA	Indicates whether an area on each page of printed output is reserved for the security label. When an area is reserved for a security label, the printed output is shifted on each page. You cannot print data in the reserved area.
			
			Note: A blank indicates that PSF sets the value to Yes if PSFMPL is active (default) and No if PSFMPL is not active.
			
			See *PSF for z/OS: Security Guide* for more information about this parameter.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description
Send default character **Values:** / = Yes Blank = No	psf-send-default-character **Values:** yes no	XTP7SDCH	Indicates whether PSF passes the default character information to the printer by fully populating the outline single-byte code page. The default is No. See "Sending default characters in single-byte outline fonts" on page 234 for more information. See "XTP7SDCH" on page 436 for more information about setting this parameter with Exit 7.
Send msgs to SYSOUT **Values:** / = Yes Blank = No	send-messages-to-sysout **Values:** yes no	XTP7MDSD	Indicates whether PSF redirects a message data set as a SYSOUT data set to another CLASS or DEST for viewing or printing. See "Redirecting the message data set" on page 328 and "XTP7MDSD" on page 432 for more information about this parameter.
Set media origin to 3800 origin for: Data set **Values:** / = Yes Blank = No	set-3800-dataset-origin **Values:** yes no	XTP738MO	Indicates whether PSF sets the user's data set media origin on continuous-forms printers to the 3800 media origin (upper left corner). The default is No. See "Media origin" on page 335 and "XTP738MO" on page 432 for more information about this parameter.
Set media origin to 3800 origin for: Data set header **Values:** / = Yes Blank = No	set-3800-dataset-header-origin **Values:** yes no	XTP7DHMO	Indicates whether PSF sets the data set header media origin on continuous-forms printers to the 3800 media origin (upper left corner). The default is No. See "Media origin" on page 335 and "XTP7DHMO" on page 434 for more information about this parameter.
Set media origin to 3800 origin for: Job header **Values:** / = Yes Blank = No	set-3800-job-header-origin **Values:** yes no	XTP7JHMO	Indicates whether PSF sets the job header media origin on continuous-forms printers to the 3800 media origin (upper left corner). The default is No. See "Media origin" on page 335 and "XTP7JHMO" on page 434 for more information about this parameter.
Set media origin to 3800 origin for: Job trailer **Values:** / = Yes Blank = No	set-3800-job-trailer-origin **Values:** yes no	XTP7JTMO	Indicates whether PSF sets the job trailer media origin on continuous-forms printers to the 3800 media origin (upper left corner). The default is No. See "Media origin" on page 335 and "XTP7JTMO" on page 434 for more information about this parameter.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description
Set media origin to 3800 origin for: Message data set **Values:** / = Yes Blank = No	set-3800-messages-origin **Values:** yes no	XTP7MDMO	Indicates whether PSF sets the message data set media origin on continuous-forms printers to the 3800 media origin (upper left corner). The default is No. See "Media origin" on page 335 and "XTP7MDMO" on page 435 for more information about this parameter.
SNMP enabled **Values:** / = Yes Blank = No	snmp-enabled **Values:** yes no	snmp-enabled	Specifies whether PSF operator commands use Simple Network Management Protocol (SNMP) Version 1 for TCP/IP-attached printers for faster response times. The default is Yes. **Note:** When this parameter is set to Yes, make sure that the printer is SNMP enabled. Otherwise, if the printer is not SNMP enabled, a 20-second delay occurs before the printer starts up.
Suppress copy marks **Values:** / = Yes Blank = No	suppress-copy-marks **Values:** yes no	XTP7SUCM	Indicates whether PSF tells the printer not to print copy marks as specified in the JES initialization definitions. The default is No. See "Suppressing copy marks" on page 157 and "XTP7SUCM" on page 436 for more information about this parameter.
TCPIP 64K buffer support **Values:** / = Yes Blank = No	tcpip-64k-buffer-support **Values:** yes no	tcpip-64k-buffer-support	PSF runs with either 8K, 32K, or 64K send buffers. When attaching to the printer, PSF indicates that its maximum send buffer size is 32K or 64K, based on the setting of the **tcpip-64k-buffer-support** AFPPARMS parameter. The printer responds back indicating that it supports a maximum send buffer size of either 32K or 64K. If this value is No, PSF supports send buffers up to 32K. If it is Yes, PSF supports send buffers up to 32K. The default is No. If this flag indicates that PSF supports a maximum send buffer size of 32K, but the printer supports 64K send buffers, PSF abends with a 0E12 reason code and issues message APS934I, indicating that the value must be Yes. **Note:** Only forTCP/IP-attached printers.

Printer Inventory: ISPF Panel	Printer Inventory: PIDU	Equivalent Parameter	Description
Trace mode **Values:** 1 = None 2 = Internal 3 = Sync 4 = Full 5 = Limit 6 = IPDS	trace-mode **Values:** none internal sync full limit ipds	PARM= (*,type*) **Note:** Applies to all FSAs in the FSS.	Specifies the type of PSF tracing that is started during FSA initialization. For more information about using traces, see *PSF for z/OS: Diagnosis*. **Values:** None: No PSF tracing is started during PSF initialization. `Internal`: An internal trace is started (default). It reflects only the most recent history of PSF processing. `Sync`: An FSA SYNC external trace is started. An internal trace is also started. `Full`: An FSA full external trace is started. An internal trace is also started. `Limit`: A shortened FSA external trace is started along with an internal trace. `IPDS`: An FSA IPDS trace is started along with an internal trace. **Note:** The FSA trace dsname parameter is required when the trace mode is `Sync`, `Full`, `Limit`, or `IPDS`.
Trace table size	trace-table-size	PARM= (*,,,pagecount*) **Note:** Applies to all FSAs in the FSS.	Specifies a number that indicates how many 4 KB pages of storage are allocated for the FSA internal trace table. This allocation occurs only if the trace mode is `Internal`, `Sync`, `Full`, `Limit`, or `IPDS`. For more information about using traces, see *PSF for z/OS: Diagnosis*. **Values:** 1–999 The default is 32.
Trace user data **Values:** / = Yes Blank = No	trace-user-data **Values:** yes no	trace-user-data	Specifies whether user data is traced when a trace function is requested with PSF. **Values:** **Yes** User data is traced when user data is available. This is the default. **No** User data is not included during trace processing. **Note:** User data is always traced with SNA-attached printers. For more information about using traces, see *PSF for z/OS: Diagnosis*.

Table 7. FSA Printer Inventory parameters compared to AFPPARMS, EXEC PARM, Exit 7 (XTP7), or PRINTDEV parameters (continued)			
Printer Inventory: ISPF Panel	**Printer Inventory: PIDU**	**Equivalent Parameter**	**Description**
Use Line-Mode Migration LINECT **Values:** / = Yes Blank = No	use-line-mode-migration-linect **Values:** yes no	XTP7LCNT	Indicates whether PSF uses line count (LINECT) in Line-Mode Migration for calculating the number of lines on a page. **Values:** **Yes** LINECT is used. **No** LINECT is not used. This is the default.
3800 compatibility: Override default font **Values:** / = Yes Blank = No	override-3800-default-font **Values:** yes no	XTP7HDF	Indicates whether PSF tells the 3800 to replace the hardware default font with the first font in the current font list. The default is No. See "Font selection" on page 335 and "XTP7HDF" on page 431 for more information about this parameter.

Specifying PRINTDEV parameters

The PRINTDEV statement specifies parameters for printer initialization. In deferred-printing mode, PRINTDEV is among the JCL statements in the PSF startup procedure. In direct-printing mode, PRINTDEV is among the JCL statements associated with the application program. For JCL descriptions and sample PRINTDEV statements, see Chapter 9, "Using deferred-printing mode," on page 123, and Chapter 10, "Using direct-printing mode," on page 167.

Table 8 on page 102 describes the initialization parameters that can be included in a PRINTDEV statement. All parameters are optional unless specified otherwise.

Notes:

1. Many of the parameters in this table can be defined in the Infoprint Server Printer Inventory. When that is the case, the parameter is marked with an asterisk (*) and you are referred to Table 6 on page 59 or Table 7 on page 66. Using the Printer Inventory is preferred over specifying parameters in the PRINTDEV statement. When you use the Printer Inventory, those parameters that are marked with an * must be specified in the Printer Inventory if you want to use them; if they are specified in the PRINTDEV statement, they are ignored. For more information about the Printer Inventory, see Chapter 7, "Using the Infoprint Server Printer Inventory," on page 53.

2. If you specify parameters in the PRINTDEV statement and do not use the Printer Inventory, you must restart all the printers in the startup procedure when you change a parameter for an existing printer in the PRINTDEV.

Table 8. Initialization parameters for the PRINTDEV statement. (*) means that if you use the Printer Inventory, this parameter is ignored in the PRINTDEV and must be specified in the Printer Inventory.	
Parameter	**Description**
ACQINTV (*)	Specifies the acquire interval in seconds. See Acquire interval for more information. **Syntax:** ACQINTV=*nnnnn*, where *nnnnn* is a value 0–86400.

Parameter	Description
Table 8. Initialization parameters for the PRINTDEV statement. (*) means that if you use the Printer Inventory, this parameter is ignored in the PRINTDEV and must be specified in the Printer Inventory. _(continued)_	
Parameter	**Description**
APPLID (*) (required for SNA-attached printers only)	Specifies the name of the VTAM application-program node for a functional subsystem application (FSA) when PSF is printing in deferred-printing mode. See Applid for more information. **Syntax:** `APPLID=`_name_
BUFNO (*)	Specifies a factor for allocating the fixed I/O buffer storage used for channel-attached printers. See Channel buffer count for more information. **Syntax:** `BUFNO=`_nnnnn_, where _nnnnn_ is a value from 1 to 10000.
CHARS (*)	Specifies the last four characters of up to four default coded font names, which are used to print a data set on the printer. See Character sets for more information. **Syntax:** `CHARS=`(_fontname1_[, _fontname2_][,_fontname3_][,_fontname4_])
COLORMAP (*)	Specifies the member name of the printer default color mapping table. See Color map for more information. **Syntax:** `COLORMAP=`(_membername_)
COMPRESS (*)	Specifies whether PSF compresses data (blanks) in line data sent to printers. See Blank compression for more information. **Syntax:** `COMPRESS=YES|`<u>`NO`</u>
COMSETUP (*) (required for microfilm device only)	Specifies the member name of the printer default microfilm setup resource object container generated with the utility provided with your microfilm device. See Com setup member for more information. **Syntax:** `COMSETUP=`_membername_ For an example of a startup procedure that uses the COMSETUP parameter with a full member name, see Figure 100 on page 346 and Figure 101 on page 347.
CONNINTV (*)	Specifies the connect interval, in seconds, during which PSF attempts to start a session or a dialog with a printer. When the connect interval expires and the connection is not complete, PSF issues a message and stops the FSA. See Connect interval for more information. **Syntax:** `CONNINTV=`_nnnnn_, where _nnnnn_ is a value 0–86400.

Table 8. Initialization parameters for the PRINTDEV statement. (*) means that if you use the Printer Inventory, this parameter is ignored in the PRINTDEV and must be specified in the Printer Inventory. *(continued)*	
Parameter	**Description**

DATACK(*)	Specifies the character and position errors the printer reports to PSF. See Print error reporting for more information.
	
	Syntax: DATACK=<u>BLOCK</u>|UNBLOCK|BLKCHAR|BLKPOS
	
	DATACK=BLOCK
	    Specifies that the printer is not to report character or position errors to PSF.
	
	DATACK=UNBLOCK
	    Specifies that the printer is to report all character and position errors to PSF.
	
	DATACK=BLKCHAR
	    Specifies that the printer is not to report character errors to PSF. (The printer reports only position errors.)
	
	DATACK=BLKPOS
	    Specifies that the printer is not to report position errors to PSF. (The printer reports only character errors.)
	
	Regardless of what the PRINTDEV statement specifies, PSF defaults to DATACK=UNBLOCK when it prints security resources.

DISCINTV (*)	Specifies the disconnect interval in seconds. See Disconnect interval for more information. **Syntax:** DISCINTV=*nnnnn*, where *nnnnn* is a value 0–86400.
DPAGELBL (*)	Indicates whether the security label is printed on each page of printed output. See Label data pages for more information.
DSHDR	Identifies an OUTPUT statement that specifies the form definition and page definition used to format and print data set header pages. **Syntax:** DSHDR=*.label **Note:** Not valid for direct-printing mode.
DUMP (*)	Specifies that a conditional memory dump of the PSF address space is to be taken when a specific reason code, restartable abend, or message occurs. See Dump: Code, Dump: Count, Dump: Message ID for more information. **Syntax:** DUMP=([*reasoncode*|ABD0*nnn*][,*msgid,count*]) **Note:** For a description of the syntax of this parameter and an explanation of restartable abends, see *PSF for z/OS: Diagnosis*.
FAILURE (*)	Specifies PSF action after a printer failure, an SNA session failure, or an Internet Protocol network failure. See Failure action for more information. **Syntax:** FAILURE=STOP|<u>WCONNECT</u>
FDEFDD (required)	Identifies the DD statement that specifies the form definition libraries. **Syntax:** FDEFDD=*.label
FONTDD (required)	Identifies the DD statement that specifies the default font libraries for FOCA fonts. **Syntax:** FONTDD=*.label

Parameter	Description
Table 8. Initialization parameters for the PRINTDEV statement. (*) means that if you use the Printer Inventory, this parameter is ignored in the PRINTDEV and must be specified in the Printer Inventory. *(continued)*	
Parameter	**Description**
FONTPATH	Identifies the DD statement that specifies the font path libraries, which contain extended code pages or TrueType and OpenType fonts. **Syntax:** FONTPATH=*.*label* For more information about TrueType and OpenType fonts, see Chapter 22, "Using TrueType and OpenType Fonts," on page 243. For more information about extended code pages, see *PSF for z/OS: User's Guide*.
FONT240	Identifies the DD statement that specifies the 240-pel font libraries. **Syntax:** FONT240=*.*label*
FONT300	Identifies the DD statement that specifies the 300-pel font libraries. **Syntax:** FONT300=*.*label*
FORMDEF (*) (required)	Specifies a default form definition. See Form definition for more information. **Syntax:** FORMDEF=*fdefname*
INTRVMSG (*)	Indicates whether PSF issues a message to the operator console when the printer is SNA- or TCP/IP-attached and has a condition that requires intervention, such as a paper jam. See Issue intervention messages for more information. **Syntax:** INTRVMSG=YES|NO **Note:** PSF uses this parameter in the PRINTDEV statement only if you are not using the Printer Inventory or Exit 7.
IPADDR (*) (required for TCP/IP-attached printers only)	Identifies the Internet Protocol (IP) address or host name of the TCP/IP-attached printer or of the i-data 7913 IPDS Printer LAN Attachment. See Printer IP address for more information. **Syntax:** IPADDR='*ip-address*' or '*host-name*'
JOBHDR	Identifies the OUTPUT statement that specifies the form definition and page definition used to format and print job-header separator pages. **Syntax:** JOBHDR=*.*label* **Note:** Not valid for direct-printing mode.
JOBTRLR	Identifies the OUTPUT statement that specifies the form definition and page definition used to format and print job-trailer separator pages. **Syntax:** JOBTRLR=*.*label* **Note:** Not valid for direct-printing mode.
LOGMODE (*)	Specifies the name of the VTAM logon-mode table entry, which defines the session parameters for an SNA-attached printer. See Logmode for more information. **Syntax:** LOGMODE=*name*
LUNAME (*) (required for SNA-attached printers only)	Specifies the unique logical-unit name of an SNA-attached printer. See LU name for more information. **Syntax:** LUNAME=*name*

Table 8. Initialization parameters for the PRINTDEV statement. (*) means that if you use the Printer Inventory, this parameter is ignored in the PRINTDEV and must be specified in the Printer Inventory. (continued)	
Parameter	**Description**

MAP2OLN (*)	Indicates whether PSF maps fonts to outline fonts. Use this parameter if your printer supports outline fonts, you have existing applications that use raster fonts, and you want to use outline fonts without changing the applications. See <u>Map to outline fonts</u> and <u>Recover from font not found</u> for more information.
	
	Syntax: `MAP2OLN=YES|(YES,CONT)|(YES,QUIT)|`<u>`NO`</u>
	
	Syntax:
	
	YES or Y
	    PSF maps fonts to outline fonts and stops processing the print job if a requested raster font cannot be mapped to an outline font.
	
	(YES,CONT) or (Y,CONT)
	    PSF maps fonts to outline fonts and continues printing a job if a requested raster font cannot be mapped to an outline font. PSF sends the requested raster font to the printer.
	
	(YES,QUIT) or (Y,QUIT)
	    PSF maps fonts to outline fonts and stops processing the print job if a requested raster font cannot be mapped to an outline font.
	
	NO or N
	    PSF does not map raster fonts to outline fonts.
	
	Note: PSF uses this parameter in the PRINTDEV statement only if you are not using the Printer Inventory or Exit 7.

MESSAGE	Identifies the OUTPUT statement that specifies the form definition and page definition used to format and print messages. **Syntax:** `MESSAGE=*.`*label* **Note:** For microfilm devices, see Chapter 29, "Supporting AFP output on microfilm," on page 341.
MGMTMODE (*)	Specifies how PSF controls an SNA-attached or TCP/IP-attached printer in deferred-printing mode. See <u>Management mode</u> for more information. **Syntax:** `MGMTMODE=`<u>`OUTAVAIL`</u>`|DIALIN|IMMED`

Parameter	Description
	Table 8. Initialization parameters for the PRINTDEV statement. (*) means that if you use the Printer Inventory, this parameter is ignored in the PRINTDEV and must be specified in the Printer Inventory. *(continued)*
Parameter	**Description**

NORESP (*)	Specifies the action that PSF takes when the time that is specified by the RESPTIME parameter expires before a response is received from the printer. See No response action and No response action: Notify for more information.
	
	Syntax: NORESP=NOTIFY|NOTIFY(*node.userid*)|NOTIFY(OPERATOR)|TERMINATE
	
	NORESP=NOTIFY
	    PSF notifies JES that an expected response was not received from the printer. This condition is also displayed by SDSF.
	
	NORESP=NOTIFY(node.userid)
	    PSF sends a message to the specified user ID indicating that an expected response was not received from the printer. The node and period can be omitted if the user ID is on the node on which PSF is running. PSF also notifies JES of the missing response.
	
	NORESP=NOTIFY(OPERATOR)
	    PSF sends a message to the system operator that indicates that an expected response was not received from the printer. PSF also notifies JES of the missing response.
	
	NORESP=TERMINATE
	    PSF stops the printer FSA. The system operator must issue a command to restart the FSA. The data set that is active when the FSA is stopped is restarted from the last checkpoint.

NPRO	Specifies whether a nonprocess runout (NPRO) occurs at the end of data set processing. An NPRO forces printed output from the paper path to the stacker. Specify NPRO only when printing on continuous forms. If NPRO is 0 (or if no NPRO parameter is specified), PSF does not initiate an NPRO. If NPRO is 1, PSF initiates an NPRO. **Syntax:** NPRO=0|1 **Notes:** 1. This parameter is used only for direct-printing mode with channel-attached printers. 2. In deferred-printing mode, the value for NPRO is derived from the JES NPRO definition; the PSF NPRO parameter has no effect. 3. The only valid values are 0 and 1. Any other value results in job scheduler messages (IEF), indicating that the numeric exceeds the maximum that is allowed in the NPRO field and that the job failed with a JCL error. 4. In direct-printing mode, a SETPRT macro that is issued within an application program might cause an NPRO, even though data set processing is not complete. 5. For microfilm devices, see Chapter 29, "Supporting AFP output on microfilm," on page 341.

Parameter	Description
Table 8. Initialization parameters for the PRINTDEV statement. (*) means that if you use the Printer Inventory, this parameter is ignored in the PRINTDEV and must be specified in the Printer Inventory. *(continued)*	
Parameter	**Description**

OBJCONDD (required with COMSETUP and COLORMAP)	Identifies the DD statement that specifies the object container library. The object container library contains:
	
	• Microfilm setup resources specified in the COMSETUP parameter for printing on a microfilm device. For more information, see Chapter 29, "Supporting AFP output on microfilm," on page 341.
	
	• Color mapping table resources that are specified in the OBJCONDD parameter for printing to color printers that support color mapping. For more information, see:
	
	  – Chapter 6, "Working with libraries," on page 49
	  – *Mixed Object Document Content Architecture Reference*
	
	• Encapsulated PostScript resources that are included by an IOB structured field.
	
	Syntax: OBJCONDD=*.*label*

OBJCPATH	Identifies the one- to eight-character name of the DD statement that specifies the paths for object container libraries, which contain data object resources, such as color management resources (CMRs). **Syntax:** OBJCPATH=*.*label*
OVERLAY (*)	Specifies the member name of a medium overlay to be placed on every sheet of output on the printer, including separator pages and message pages. See Overlay for more information. **Syntax:** OVERLAY=*overlayname* **Example:** OVERLAY=O1GLOBAL
OVLYDD	Identifies the DD statement that specifies the default overlay libraries. Required if OVLY240 or OVLY300 is specified (the default library is used when print data does not give a resolution); otherwise, optional if no attempt is made to print an overlay from the system overlay library. **Syntax:** OVLYDD=*.*label*
OVLY240	Identifies the DD statement that specifies the 240-pel overlay libraries. **Syntax:** OVLY240=*.*label*
OVLY300	Identifies the DD statement that specifies the 300-pel overlay libraries. **Syntax:** OVLYDD=*.*label*
PAGEDEF (*) (required)	Specifies the default page definition. (This page definition is also used if one is not specified in JOBHDR, JOBTRLR, DSHDR, and MESSAGE parameters.) See Page definition for more information. **Syntax:** PAGEDEF=*pdefname*
PDEFDD (required)	Identifies the DD statement that specifies the page definition libraries. **Syntax:** PDEFDD=*.*label*

Table 8. Initialization parameters for the PRINTDEV statement. (*) means that if you use the Printer Inventory, this parameter is ignored in the PRINTDEV and must be specified in the Printer Inventory. *(continued)*	
Parameter	**Description**
PIMSG (*)	Indicates whether all message groups generated in the processing of a data set are printed at the end of the data set. Also indicates the maximum number of message groups that are printed. See Print error messages and Print error messages: Maximum messages for more information.
	Syntax: PIMSG=(YES [,*nnn*] |NO [,*nnn*])
	nnn indicates the maximum number of messages and specifies that after PSF generates *nnn* message groups, it is to end processing of the data set and purge the data set from the spool. The final count of printed messages might be more than *nnn* message groups if the message groups are generated for errors that are reported by the printer for pages that are sent before the message count is reached. A value of 0 allows data set printing to continue, regardless of the number of message groups that are generated, unless an error that stops processing occurs. The default is 16.
PORTNO (*)	Specifies the port number with which PSF is to establish a connection. See Port number for more information.
	Syntax: PORTNO=*nnnnn*, where *nnnnn* is a value 5001–65535. The default value is 5001.
PRMODE (*)	Indicates the default processing mode that PSF uses to print data sets containing both single-byte and double-byte fonts. See Default process mode for more information.
	Syntax: PRMODE=SOSI1|SOSI2|SOSI3|SOSI4
	PRMODE=SOSI1 specifies that each shift-out, shift-in code is converted to a blank and a Set Coded Font Local text control.
	PRMODE=SOSI2 specifies that each shift-out, shift-in code is converted to a Set Coded Font Local text control.
	PRMODE=SOSI3 specifies that the shift-in code is converted to a Set Coded Font Local text control and two blanks. A shift-out code is converted to a Set Coded Font Local text control.
	PRMODE=SOSI4 specifies that each shift-out, shift-in code is to be skipped and not counted when offsets are calculated for the print data set. SOSI4 is used when double-byte character set (DBCS) text is converted from ASCII to EBCDIC. When SOSI4 is specified, the page definition offsets are correct after conversion; therefore, the user does not need to account for SOSI characters when FIELD offsets are computed. The data conversion that PSF makes for SOSI4 is the same as for SOSI2.
	Notes:
	1. PSF uses this parameter only if you are not using the Printer Inventory or Exit 7 and the PRMODE keyword is not specified on the OUTPUT JCL statement.
	2. If this parameter is not specified in the PRINTDEV statement, the Printer Inventory, or Exit 7, and the PRMODE keyword is not specified on the OUTPUT JCL statement, PSF defaults to either line data or MO:DCA-P, depending on the type of data stream.
PRTERROR (*)	Controls whether the PRTERROR keyword is accepted on the OUTPUT JCL statement. See Error disposition supported for more information.
	Syntax: PRTERROR=HONOR|NOTHONOR
	PRTERROR=HONOR specifies that PSF accepts PRTERROR if specified on the OUTPUT JCL statement for data sets printed on this printer.
	PRTERROR=NOTHONOR specifies that PSF ignores PRTERROR if specified on the OUTPUT JCL statement for data sets printed on this printer. This is the default.

Parameter	Description
Table 8. Initialization parameters for the PRINTDEV statement. (*) means that if you use the Printer Inventory, this parameter is ignored in the PRINTDEV and must be specified in the Printer Inventory. *(continued)*	
Parameter	**Description**
PSEGDD	Identifies the DD statement that specifies the default page segment libraries. In addition to page segments, the page segment library might contain BCOCA, GOCA, IOCA, and PTOCA resources. Required if PSEG240 or PSEG300 is specified (the default library is used when print data does not give a resolution); otherwise, optional if no attempt is made to print a page segment from the system page segment library. **Syntax:** PSEGDD=*.*label*
PSEG240	Identifies the DD statement that specifies the 240-pel page segment libraries. **Syntax:** PSEG240=*.*label*
PSEG300	Identifies the DD statement that specifies the 300-pel page segment libraries. **Syntax:** PSEG300=*.*label*
RELINTV (*)	Specifies the release interval, in seconds. See Release interval for more information. **Syntax:** RELINTV=*nnnnn*, where *nnnnn* is a value 0–86400.
RELMODE (*)	Specifies the manner in which PSF is to respond to a request to release the printer. See Release mode for more information. **Syntax:** RELMODE=IDLE|TIME|NONE
RESPTIME (*)	Specifies the maximum number of seconds PSF waits for a response from the printer. See Response timeout for more information. **Syntax:** RESPTIME=*nnnnn*, where *nnnnn* is a value 0–86400.
SDEFDD	See *PSF for z/OS: Security Guide* for a description of this parameter.
SDEF240	See *PSF for z/OS: Security Guide* for a description of this parameter.
SDEF300	See *PSF for z/OS: Security Guide* for a description of this parameter.
SETUP (*)	Specifies the setup parameters for which JES issues a setup message when the printer is initialized and at the start of any job that specifies a change in a setup parameter from what is active for the printer. See Issue setup messages for more information. **Syntax:** SETUP=FORMS|BURST|(FORMS,BURST)
SFONTDD	See *PSF for z/OS: Security Guide* for a description of this parameter.
SFONT240	See *PSF for z/OS: Security Guide* for a description of this parameter.
SFONT300	See *PSF for z/OS: Security Guide* for a description of this parameter.
SOVLYDD	See *PSF for z/OS: Security Guide* for a description of this parameter.
SOVLY240	See *PSF for z/OS: Security Guide* for a description of this parameter.
SOVLY300	See *PSF for z/OS: Security Guide* for a description of this parameter.
SPAGELBL (*)	Indicates whether the security label is printed on a separator page. See Label separator pages for more information.
SPSEGDD	See *PSF for z/OS: Security Guide* for a description of this parameter.
SPSEG240	See *PSF for z/OS: Security Guide* for a description of this parameter.
SPSEG300	See *PSF for z/OS: Security Guide* for a description of this parameter.

Table 8. Initialization parameters for the PRINTDEV statement. (*) means that if you use the Printer Inventory, this parameter is ignored in the PRINTDEV and must be specified in the Printer Inventory. (continued)	
Parameter	**Description**
SYSAREA (*)	Indicates whether an area on each page of printed output is reserved for the security label. See Restrict printable area for more information.
TIMEOUT (*)	Specifies the action that PSF is to take after a timeout when no output is available from JES. See Disconnect action for more information. **Syntax:** TIMEOUT=<u>REDRIVE</u>|STOP
TRACE (*)	Specifies PSF tracing. For more information about using traces, see *PSF for z/OS: Diagnosis*. See **NST trace dsname**, **Trace mode**, **Trace prompt**, and **Trace table size** in Table 6 on page 59 and FSA trace dsname for more information. **Syntax:** TRACE=<u>YES</u>|NO

Specifying AFPPARMS control statement parameters

Additional parameters that PSF uses are contained in an AFPPARMS control statement. You can specify this control statement with the AFPPARMS DD name in a startup procedure (see "Sample PSF startup procedure for four printers" on page 137); for example:

```
//AFPPARMS    DD DSN=FSS.PDS.AFPPARMS,DISP=SHR
```

The AFPPARMS control statement that is specified with the AFPPARMS DD name is a partitioned data set. It is recommended that the data set name end with a qualifier of AFPPARMS. Also, use DISP=SHR so that multiple FSAs can access the AFPPARMS data set.

How the AFPPARMS data set is allocated

Table 9 on page 111 shows how the AFPPARMS data set specified with the AFPPARMS DD name in the startup procedure is allocated.

Table 9. Allocation of AFPPARMS data set attributes			
Attribute	**Value**	**Type**	**Description**
DCB=DSORG=	PO	Required	Data set organization
DSNTYPE=	LIBRARY	Required	Data set defined as PDSE
DCB=RECFM=	*xx*	Required	Any value except "U"
DCB=LRECL=	*nnnn*	Required	Maximum bytes in the record (see Note "1" on page 111)
DISP=	SHR	Required	Data set can be shared
SPACE=	(CYL,(*nn*,1,10))	Required	Direct access storage device (DASD) cylinders that are required to process data (see Note "2" on page 112)

Notes:

1. Any record size is valid; however, use a small record size, such as DCB=LRECL=120, to conserve DASD space.

2. Space requirements depend on the type of DASD that is used and on the number of FSAs that are specified in the startup procedure.

How members in the data set are specified

The AFPPARMS control statement can contain a defaults member, named PSFDEF, and members with the same names as the FSA names specified in the startup procedure. These members are optional, but you can use them to associate AFPPARMS parameters to specific FSAs. You can use an FSA member name to specify parameters that apply to that FSA. For example, if you want to associate AFPPARMS parameters to an FSA named PRT619, which is specified in the startup procedure, you specify the parameters in a member that is named PRT619 in the AFPPARMS control statement. PSF finds the member in the data set and uses the AFPPARMS parameters to process the spool data sets for the PRT619 FSA.

You can also use the PSFDEF member to specify default parameters that apply to all FSAs in the FSS. However, if a member for a specific FSA is specified, it is also used and overrides the same parameters that are specified in the PSFDEF member. For example:

- The PSFDEF member specifies this parameter:

```
auxiliary-files-modca-level = none
```

- The PRT619 member specifies this parameter:

```
auxiliary-files-modca-level = IS3
```

- Therefore, the parameter value PSF uses is:

```
auxiliary-files-modca-level = IS3
```

Selection hierarchy for AFPPARMS parameters

PSF uses the AFPPARMS parameters that are specified in the Printer Inventory when the EXEC statement specifies the Printer Inventory in the PSF startup procedure. Otherwise, PSF selects the AFPPARMS parameters in this order from member names in the AFPPARMS control statement that is specified in the PSF startup procedure:

1. User's OUTPUT JCL statement
2. FSA member name
3. Defaults member name, PSFDEF

Note: If PSF uses the Printer Inventory to select AFPPARMS parameters, the FSA and defaults member names are ignored.

Syntax of the AFPPARMS control statement

The syntax guidelines for the AFPPARMS control statement are:

- A parameter is a *keyword=value* pair.
- Spaces are allowed after the keyword and before the value (on either side of the =).
- Only one parameter is allowed on a specific line.
- A parameter can span multiple lines. For example, the keyword can be on the first line, = can be on the next line, and the value can be on the third line.
- Parameter keywords and values can be specified in uppercase, lowercase, or mixed case.
- Comments are delimited with #. PSF ignores everything on a line after the # character.

Parameters for the AFPPARMS control statement

shows the parameters that are valid in the AFPPARMS control statement, which is specified by the AFPPARMS DD name in the startup procedure or on a job-by-job basis with the

AFPPARMS parameter on the OUTPUT JCL statement. For information about specifying the AFPPARMS parameter on the OUTPUT statement, see *PSF for z/OS: User's Guide*.

Notes:

1. If you specify a parameter in the AFPPARMS control statement in an FSA or PSFDEF member when you are using the Printer Inventory, the parameter in the AFPPARMS control statement is ignored.

2. When you change a parameter for the defaults member in the AFPPARMS control statement, you must restart all the FSAs in the startup procedure.

Table 10. AFPPARMS control statement parameters. An X in a column indicates that the parameter can be specified by the AFPPARMS DD name in the startup procedure or the AFPPARMS parameter on the OUTPUT JCL statement.

Parameter	Description	Startup procedure	OUTPUT JCL

`above-the-bar-storage={inline-resource|none}`	Specifies whether PSF uses storage that is above the bar for processing inline resources.
	
	The values are:
	
	inline-resources
	    PSF stores inline resources above the 2 GB bar during job processing.
	none
	    PSF stores inline resources above the 16 MB line but below the 2 GB bar (in the region area) during job processing. This is the default.
		X	X

Parameter	Description	Startup procedure	OUTPUT JCL
`auxiliary-files-modca-level={is3｜bpf-match｜none}`	Specifies the MO:DCA interchange set level that auxiliary files, such as separator pages and message files, support. **Note:** In PSF, use this parameter only if the printer supports only TrueType and OpenType fonts. The values are: **bpf-match** Auxiliary files contain the same Begin Print File (BPF) MO:DCA interchange set triplet and function set triplet that are specified on the BPF structured field in the user's data set. **is3** Auxiliary files are MO:DCA IS/3 compliant. **none** Auxiliary files do not support a MO:DCA IS level. This is the default. **Notes:** 1. Make sure that changes are made to the PRINTDEV statement for this FSA so that auxiliary files are generated correctly. See "Setting up MO:DCA interchange set auxiliary pages" on page 116. 2. See Auxiliary files MO:DCA level for the parameter you specify when you are using the Printer Inventory.	X	

Table 10. AFPPARMS control statement parameters. An X in a column indicates that the parameter can be specified by the AFPPARMS DD name in the startup procedure or the AFPPARMS parameter on the OUTPUT JCL statement. *(continued)*

Parameter	Description	Startup procedure	OUTPUT JCL

`resource-deletion-ratio=`*nnn*	Specifies a number that determines whether PSF expedites the deletion of all fonts, overlays, object containers, and page segments at the end of data set processing. If the ratio of inline and user resources to the total number of loaded resources (expressed as a percentage) is greater than or equal to this parameter value, PSF expedites resource deletion. If the ratio percentage is less than this parameter value, standard methods are used to delete the resources.
	
	The values are:
	
	1-100
	    PSF expedites resource deletion if the resource ratio percentage is greater than or equal to this value. The default is 90.
	
	0
	    PSF does not use this parameter when it is deleting resources.
	
	For more information, see "Expediting resource deletion" on page 189.
		X	

`snmp-enabled={`<u>`yes`</u>`	no}`	Specifies whether PSF operator commands use Simple Network Management Protocol (SNMP) Version 1 for TCP/IP-attached printers for faster response times. The default is yes. **Notes:** 1. When this parameter is set to yes, make sure that the printer is SNMP enabled. Otherwise, if the printer is not SNMP enabled, a 20-second delay occurs before the printer starts up. 2. See SNMP enabled for the parameter you specify when you are using the Printer Inventory.	X	

Parameter	Description	Startup procedure	OUTPUT JCL
`tcpip-64k-buffer-support={no│yes}`	PSF runs with either 8K, 32K, or 64K send buffers. When attaching to the printer, PSF indicates that its maximum send buffer size is 32K or 64K, based on the setting of the **tcpip-64k-buffer-support** AFPPARMS parameter. The printer responds back indicating that it supports a maximum send buffer size of either 32K or 64K. If this value is OFF (No), PSF supports send buffers up to 32K. If it is ON (Yes), PSF supports send buffers up to 32K. The default is OFF (No). If this flag indicates that PSF supports a maximum send buffer size of 32K, but the printer supports 64K send buffers, PSF abends with a 0E12 reason code and issues message APS934I, indicating that the value must be Yes.	X	
`trace-user-data={yes│no}`	Specifies whether user data is traced when a trace function is requested with PSF. User data is traced when yes is specified, and user data is available. The default is yes. **Notes:** 1. User data is always traced with SNA-attached printers. 2. See Trace user data for the parameter you specify when you are using the Printer Inventory. 3. For more information about using traces, see *PSF for z/OS: Diagnosis*.	X	

APSJPRM1 sample member for AFPPARMS

The APSJPRM1 sample member, which contains AFPPARMS parameters, is in SYS1.SAMPLIB. You can either view the sample online or print it. You can also copy the sample to your AFPPARMS data set to use as the defaults member or as the member for one of your FSAs.

Setting up MO:DCA interchange set auxiliary pages

PSF can add these auxiliary pages to the output data stream:

- Separator pages, such as job header, data set header, and job trailer
- Interrupt message pages
- Message data sets

You must configure PSF to create auxiliary files at a specific interchange set level in these cases:

- The transforms that are processing the IPDS output require a specific interchange set level of IPDS.

- You are using AFP Download Plus to transmit the same jobs that you are printing, and you want the printed jobs to match the transmitted jobs.

If all your print jobs are MO:DCA IS/3, see "Generating MO:DCA IS/3 auxiliary pages" on page 117. Otherwise, if your print jobs contain different MO:DCA interchange set levels, see "Matching MO:DCA auxiliary pages" on page 117.

Generating MO:DCA IS/3 auxiliary pages

You can use the following procedure to generate IS/3 auxiliary pages with page definitions that use only TrueType and OpenType fonts, and then set a parameter so that PSF checks that the auxiliary pages contain only TrueType and OpenType fonts:

1. Create or update a PRINTDEV statement in the startup procedure to specify separator page and message OUTPUT statements with page definitions that use only TrueType and OpenType fonts, not FOCA fonts. For example, the OUTPUT statement can define one of these sample page definitions that are included with PSF:

   - P1TT6462
   - P1TT6483

   See PRT4 in "Sample PSF startup procedure for four printers" on page 137 for an example of the startup procedure. See *PSF for z/OS: User's Guide* for more information about page definitions.

2. Set the Printer Inventory Auxiliary files MO:DCA level parameter to IS/3 or the AFPPARMS auxiliary-files-modca-level parameter to `is3`.

   When PSF is processing auxiliary pages, it checks that the pages contain only TrueType and OpenType fonts. If they contain FOCA fonts, PSF stops processing the auxiliary pages and issues an error message.

Matching MO:DCA auxiliary pages

If the print jobs that are processed by an FSA are a mix of MO:DCA IS/3, MO:DCA IS/3 with function sets, and print jobs that do not specify a MO:DCA interchange set level, the interchange set specification on the BPF and BDT structured fields of the auxiliary files must match the interchange set specified on the print file. You can use the following procedure if you are using IPDS transforms that require the auxiliary files to comply with the print file's interchange set level:

1. Create or update AFP separator pages that are compliant with the MO:DCA interchange set levels of the jobs you are printing (instead of using the default line data separator pages).

2. Follow Step "1" on page 117 in "Generating MO:DCA IS/3 auxiliary pages" on page 117 to update the startup procedure for the message data set.

3. Set the Printer Inventory Auxiliary files MO:DCA level parameter to BPF match or the AFPPARMS auxiliary-files-modca-level parameter to `bpf-match`.

AFP Download Plus might add auxiliary files, such as separator pages and message files, to a job. You can use the following procedure to make sure that the interchange set specification on the BPF and BDT structured fields of the auxiliary files matches the interchange set specified on the print file:

1. Follow Step "1" on page 117 in "Generating MO:DCA IS/3 auxiliary pages" on page 117 to update the startup procedure for both separator pages and the message data set.

2. Set the Printer Inventory Auxiliary files MO:DCA level parameter to BPF match or the AFPPARMS auxiliary-files-modca-level parameter to `bpf-match`.

Note: In this procedure, some auxiliary files might be created that are not compliant with the interchange set that is specified on the print file; however, separator pages for IS/3 compliant print jobs are ensured to use only TrueType and OpenType fonts. For more information about processing specific MO:DCA interchange set compliant files in AFP Download Plus, see *PSF for z/OS: AFP Download Plus*.

Using PSF parameters to attach and share printers

The Printer Inventory and the PRINTDEV statement are used to specify parameters for:

- Dedicated network-attached printers
- Shared printers with other print drivers

Only certain PSF parameters are used to attach and share printers. The other PSF parameters are not needed and can be set to whatever you want.

Hint: Use the sample startup procedure that is included with PSF in SYS1.PROCLIB and modify it with the required parameters. For example, to share a TCP/IP-attached printer, use APSWPROR or APSWPROT.

Selecting PSF parameters for a dedicated network-attached printer

When your printer is attached to a single PSF on an SNA or Internet Protocol network, specify the PSF parameters that are listed in Table 11 on page 118, depending on whether you are using the Printer Inventory or the PRINTDEV statement. PSF parameters that are not listed in Table 11 on page 118 can be set to whatever you want.

Table 11. PSF parameters for a dedicated network-attached printer

Printer Inventory	PRINTDEV Statement
Management mode=Immediate	MGMTMODE=IMMED
Connect interval=0 **Note:** Unless you want to detect when session startup is delayed excessively.	CONNINTV=0 (See note under Printer Inventory parameter.)
Disconnect interval=0	DISCINTV=0
Failure action=Wait for connect	FAILURE=WCONNECT

Selecting PSF parameters for shared printers

PSF supports the sharing of printers with other print drivers by three different methods:

- JES spool-driven
- VTAM release request
- Port switching

PSF releases a printer only at the end of a job, so that output is never split.

Note: If a setup message is issued for a job, PSF considers that the job started and does not release the printer, even if the release timer expired. When the operator responds to the setup message and the job completes on the printer, PSF releases the printer.

JES spool-driven

This method is valid for SNA-attached or TCP/IP-attached printers. A session with the printer is maintained while output exists on the JES spool and the printer is available. When no more output exists on the spool, and the disconnect interval expires, PSF ends the session with the printer. When more work exists on the spool for this printer, PSF attempts to restart the session. After the session is restarted, PSF must reload the resources required for the print jobs. With this method, PSF cannot determine whether another print driver is active.

This method is requested by specifying the PSF parameters that are listed in Table 12 on page 119, depending on whether you are using the Printer Inventory or the PRINTDEV statement. PSF parameters that are not listed in Table 12 on page 119 can be set to whatever you want.

Table 12. PSF parameters for JES spool-driven printer sharing	
Printer Inventory	**PRINTDEV Statement**
Management mode=Output available	MGMTMODE=OUTAVAIL
Disconnect interval=*nn*	DISCINTV=*nn*
Disconnect action=Redrive	TIMEOUT=REDRIVE
Failure action=Wait for connect	FAILURE=WCONNECT
Release mode=None	RELMODE=NONE
Note: See "Recommended values for PSF timers" on page 121 for values for *nn*.	

VTAM release request

This method is valid only for SNA-attached printers. A session with a printer is maintained until VTAM indicates that another print driver in the VTAM network requested a session with the printer. The session with the printer is then ended according to the release mode specified. After the session is restarted, PSF must reload any resources required for the print jobs.

If the release mode is IDLE, PSF ends the session when a VTAM request to release is received and no output exists on the spool for the printer for the release interval specified. When more work exists on the spool for this printer, PSF attempts to restart the session. This method is requested by specifying the PSF parameters that are listed in Table 13 on page 119, depending on whether you are using the Printer Inventory or the PRINTDEV statement. PSF parameters that are not listed in Table 13 on page 119 can be set to whatever you want.

Table 13. PSF parameters for VTAM release request printer sharing in IDLE mode	
Printer Inventory	**PRINTDEV Statement**
Management mode=Output available	MGMTMODE=OUTAVAIL
Release mode=Idle	RELMODE=IDLE
Release interval=*nn*	RELINTV=*nn*
Note: See "Recommended values for PSF timers" on page 121 for values for *nn*.	

If the release mode is TIME when a VTAM request to release is received, PSF waits for the time that is specified by the release interval, and then ends the session. PSF attempts to restart the session when the acquire interval expires, in accordance with the management mode specified. This method is requested by specifying the PSF parameters in Table 14 on page 119, depending on whether you are using the Printer Inventory or the PRINTDEV statement. PSF parameters that are not listed in Table 14 on page 119 can be set to whatever you want.

Table 14. PSF parameters for VTAM release request printer sharing in TIME mode	
Printer Inventory	**PRINTDEV Statement**
Release mode=Time	RELMODE=TIME
Release interval=*nn*	RELINTV=*nn*
Acquire interval=*nn*	ACQINTV=*nn*
Note: See "Recommended values for PSF timers" on page 121 for values for *nn*.	

Port switching

This method is valid for printers that support multiple active ports. The two types of port switching are printer-controlled and IPDS.

Printer-controlled port switching

Printers that support multiple ports but do not use IPDS dialogs, such as the IBM Network Printer Models 12, 17, and 24, do printer-controlled port switching. These printers can do port switching without informing PSF. For these printers, release mode and release interval have no effect.

IPDS port switching

Printers that support multiple active ports and IPDS dialogs, such as the 3130 printer with Release 2 microcode, do IPDS port switching. This method is the most efficient for printer-sharing because print resources are retained in the printer while the printer is switched to another port. When the printer indicates that another port requested control, PSF ends its dialog in accordance with the release mode specified.

If the release mode is IDLE, PSF ends the dialog when an IPDS request to release is received and no output exists on the spool for the printer for the time specified by the release interval. PSF attempts to restart the dialog when more work exists on the spool for this printer. This method is requested by specifying the PSF parameters in Table 15 on page 120, depending on whether you are using the Printer Inventory or the PRINTDEV statement. PSF parameters that are not listed in Table 15 on page 120 can be set to whatever you want.

Table 15. PSF parameters for IPDS port switching printer sharing in IDLE mode	
Printer Inventory	**PRINTDEV Statement**
Release mode=Idle	RELMODE=IDLE
Release interval=*nn*	RELINTV=*nn*
Management mode=Output available	MGMTMODE=OUTAVAIL
Note: See "Recommended values for PSF timers" on page 121 for values for *nn*.	

If the release mode is TIME when an IPDS request to release is received, PSF waits for the specified release interval, and then ends the dialog. PSF attempts to restart the dialog when the acquire interval expires. This method is requested by specifying the PSF parameters Table 16 on page 120, depending on whether you are using the Printer Inventory or the PRINTDEV statement. PSF parameters that are not listed in Table 16 on page 120 can be set to whatever you want.

Table 16. PSF parameters for IPDS port switching printer sharing in TIME mode	
Printer Inventory	**PRINTDEV Statement**
Release mode=Time	RELMODE=TIME
Release interval=*nn*	RELINTV=*nn*
Acquire interval=*nn*	ACQINTV=*nn*
Management mode=Output available	MGMTMODE=OUTAVAIL
Note: See "Recommended values for PSF timers" on page 121 for values for *nn*.	

Note: When a release or acquire interval of 0 is specified, the related action is taken immediately.

Recommended values for PSF timers

When a PSF printer is shared with other printer drivers, the values that are wanted for the PSF timers vary depending on the:

- Job mix expected through the printer drivers
- Share capabilities of the other printer drivers
- Priorities that are given to the job streams

Verify that the printer driver that is sharing the printer is also disconnecting the session or releasing the printer when it finishes printing or after a designated period.

As a starting point, use the timer values in Table 17 on page 121 when appropriate, depending on whether you are using the Printer Inventory or the PRINTDEV statement.

Table 17. Recommended PSF timer values for sharing printers	
Printer Inventory	**PRINTDEV Statement**
Connect interval=300 (specified in most cases)	CONNINTV=300 (specified in most cases)
Disconnect interval=240	DISCINTV=240
Release interval=240	RELINTV=240
Acquire interval=30	ACQINTV=30

Chapter 9. Using deferred-printing mode

PSF supports AFP printers as system output devices for deferred printing under the Job Entry Subsystem (JES). Some system printers are managed by a JES output writer that operates entirely in the JES address space. However, PSF-attached printers are defined as functional subsystem applications (FSAs) that run under a separate address space called a PSF functional subsystem (FSS). A functional subsystem interface (FSI) maintains communication between the FSS and the JES global address space.

To use a PSF printer as a system output device for deferred printing, you must first do these tasks:

1. Define one or more PSF FSSs.
2. Define FSAs for each PSF printer. For more information, see "Defining PSF FSSs and FSAs for printers" on page 123.
3. Make sure that you have a startup procedure for each PSF FSS. For more information, see "PSF startup procedures for AFP printers" on page 135.

After you complete Steps "1" on page 123 through "3" on page 123, the operator can enter commands to start each FSA. JES starts the FSS automatically when the first FSA it controls is started.

The PSF FSA dynamically allocates a channel-attached printer or establishes a session for an SNA-attached or TCP/IP-attached printer, and manages the printer until the FSA is stopped. The JES operator can control the FSA-managed printer by using JES operator commands. For information about the JES commands the operator uses to start and stop PSF FSAs, see "Starting PSF FSAs" on page 146 and "Stopping PSF FSAs and FSSs" on page 147. For more information about the JES commands the operator uses to control PSF, see the JES commands publication for the level of JES you are using.

You can also use System Display and Search Facility (SDSF) commands or the commands of a comparable product to manage FSAs. see *z/OS SDSF Operation and Customization* for more information about SDSF commands.

Note: For configurations that include PSF that is attached to a distributed print function (DPF), JES deletes the print data set from the host spool after PSF transmits a data set to the DPF spool. The host operator cannot control the print data set after it is transmitted to DPF.

Defining PSF FSSs and FSAs for printers

To use a PSF printer for deferred printing under JES, you must first code JES initialization statements to define PSF FSSs and FSAs for PSF printers. For JES2, the FSS is defined by an FSS statement, and each printer FSA or range of printer FSAs is defined by a PRT*nnnn* statement. For JES3, the FSS is defined by an FSSDEF statement, and each printer FSA is defined by a DEVICE statement. For information about defining the FSS and printer FSA to JES, see "Sample initialization statements for JES2" on page 124 and "Sample initialization statements for JES3" on page 129.

The following number of FSSs and FSAs are supported:

- A maximum of 32767 FSSs for JES2 and 2000 FSSs for JES3
- A maximum of 128 FSAs per FSS (see Note 1)

Notes:

1. The number of FSAs that can process concurrently might be less than the number of FSAs you can define. The number of FSAs that can process concurrently depends on several factors, including:

    - The virtual region size below the 16 MB line
    - The number and the size of form definitions and page definitions
    - The number of buffers specified for channel-attached printers
    - The block sizes of the font, page segment, overlay, form definition, and page definition libraries

To determine the maximum number of FSAs per FSS, see the JES initialization and tuning publication for the level of JES you are using.

2. If you want to maintain a high level of system redundancy for your high-speed production printers, you need to consider running a single FSA per FSS. In case a CANCEL of an FSS is required, only one high-speed printer is affected.

3. If you start many FSAs, most of which are not actively printing at any one time, you might want to specify that the resource libraries are to be closed when a printer is idle. See Close libraries when idle.

For more information, see "Selecting region sizes" on page 194.

In addition to the FSS(*fss_name*) and PRT(*nnnn*) statements, the JES2 OUTCLASS(*v*) initialization statement may have considerations for PSF for z/OS with regards to trailing blank truncation on input records.

- On the JES2 OUTCLASS statement, prior to z/OS 2.4 with the following JES2 APARs/PTFs applied:
  - OA58718/UJ03249
  - OA57466/UJ03238
  - OA58699/UJ90000

  there is a trailing blanks truncation parameter: BLNKTRNC={YES | NO} (YES is the default). It specifies whether (YES) or not (NO) JES2 truncates trailing blanks for non-page-mode data sets in this output class. Non-page-mode data includes line-mode and XML. This parameter affects the length of incoming records received from JES2.

  Specifically, if BLNKTRNC = YES, the non-page-mode data records do not contain the padding blanks up to the length of the lrecl of the fixed block data set allocation.

- If you are configuring PSF on z/OS 2.4 with the following JES2 APARs applied:
  - OA60528
  - OA60605

  or a later release of z/OS and you specified JES2 OUTCLASS BLNKTRNC=NO for at least one SYSOUT class, you need to specify these JCL parameters on the EXEC statement of your startup procedure to continue to achieve this function:

  - If you specified the JES2 OUTCLASS, BLNKTRNC parameter equal to NO for **all** the output classes, specify NOBLNKTRNC for the **blank_truncation** parameter.
  - If you only specified JES2 OUTCLASS, BLNKTRNC=NO for **some** of your output classes:
    - Specify BLNKTRNC for the **blank_truncation** parameter or permit it to default.
    - Specify each class for which you specified BLNKTRNC=NO on the JES2 OUTCLASS statement in the **restore_blanks_classes** parameter.

  See the **blank_truncation** and **restore_blanks_classes** parameters under "PARM parameters on the EXEC statement" on page 144.

  Note: The default for the **blank_truncation** parameter is BLNKTRNC.

Sample initialization statements for JES2

Figure 36 on page 125 shows sample JES2 statements for a channel-attached printer (PRT1), an SNA-attached printer (PRT2), and a TCP/IP-attached printer (PRT3), and a TCP/IP-attached printer that generates auxiliary pages that use only TrueType and OpenType fonts (PRT4).

Note: These statements are samples only. For specific coding for JES parameters, see the *z/OS JES2 Initialization and Tuning Reference*.

```
FSS(FSS1) PROC=SAMPPROC,HASPFSSM=HASPFSSM,AUTOSTOP=YES
PRT1      FSS=FSS1,MODE=FSS,PRMODE=(LINE,PAGE,SOSI1),
          UNIT=00E,CLASS=A,UCS=0,SEP=YES,SEPDS=YES,CKPTPAGE=100,
          START=NO,MARK=YES,NPRO=99,TRKCELL=YES
PRT2      FSS=FSS1,MODE=FSS,PRMODE=(LINE,PAGE,SOSI1),
          CLASS=B,UCS=0,SEP=YES,SEPDS=NO,CKPTPAGE=100,
          START=NO,MARK=YES
PRT3      FSS=FSS1,MODE=FSS,PRMODE=(LINE,PAGE,SOSI1),
          CLASS=C,UCS=0,SEP=YES,SEPDS=NO,CKPTPAGE=100,
          START=NO,MARK=YES,TRKCELL=YES
PRT4      FSS=FSS1,MODE=FSS,PRMODE=(PAGE),
          CLASS=D,UCS=0,SEP=YES,SEPDS=NO,CKPTPAGE=100,
          START=NO,MARK=YES,TRKCELL=YES
```

Figure 36. Sample JES2 initialization statements for four printers

Parameters for the JES2 initialization statement

"FSS statement" on page 125 and "PRTnnnn statement" on page 125 describe the JES2 initialization statements that are shown in Figure 36 on page 125, which are typical of the JES2 statements used with PSF. For more detailed descriptions of all JES2 initialization statements, see the *z/OS JES2 Initialization and Tuning Reference*.

The descriptions tell which statements and parameters are required. In the programming syntax that is shown for each parameter, defaults are underscored.

Note: When coding JES initialization statements, consider the characteristics of each printer. For example, BURST applies only to continuous-forms printers with a burster-trimmer-stacker installed.

FSS statement

The FSS initialization statement is optional but recommended. If it is omitted, JES2 generates a default FSS for that device with the same name as the device. The FSS initialization statement defines an FSS to JES2. The FSS statement is associated with one or more PRT*nnnn* statements that define each printer FSA. The FSS statement contains the following parameters for PSF:

FSS(*fssname*)
Specifies the unique 1–8 character name of a particular FSS. This name is referenced in the PRT*nnnn* statement for each printer FSA that is associated with this FSS. When you start the first printer that has an FSA definition for that FSS, an FSS address space is created for PSF. PSF manages this FSS and the printer FSA for the first printer. If you start a second printer with an FSA definition for the same FSS, PSF manages a separate FSA for the second printer in that FSS.

PROC
Specifies a procedure for starting the PSF FSS. The procedure, which must be defined before that FSS is started, is a member of a system procedure library, such as SYS1.PROCLIB. For more information, see "PSF startup procedures for AFP printers" on page 135.

HASPFSSM
Specifies the 1–8 character name of the load module that is loaded into the PSF FSS address space. This load module contains the various FSI service routines that JES2 supplies. For PSF, specify the default value (HASPFSSM=HASPFSSM).

AUTOSTOP=YES|NO
Specifies whether the FSS stops automatically after you stop all FSAs.

PRTnnnn statement

A PRT*nnnn* statement, which is required to define each printer FSA or range of printer FSAs, is associated with an FSS statement. For information about specifying a range of printers on the PRT*nnnn* statement, see the *z/OS JES2 Initialization and Tuning Reference*. Each printer FSA must have a unique PRT*nnnn* name that must match the label on the CNTL, ENDCNTL, and PRINTDEV statements for the PSF startup procedure specified in the PROC parameter. For more information, see "Sample PSF startup procedure for four printers" on page 137.

Note: The label on the PRINTDEV statement must be 8 characters or fewer. PSF accepts the PRT*nnnnn*, PRINT*nnn*, and PRINTER*n* formats. In this publication, PRT*nnnn* is used to represent the printer definition statement; *nnnn* can be 1–5 digits.

The PRT*nnnn* parameters create JES2 default values that are used unless other values are specified in the application program JCL. The PRT*nnnn* statement contains the following parameters:

BURST=YES|NO
> Specifies whether to thread the paper to a burster-trimmer-stacker (BTS) for bursting and offset stacking, or to send the paper to a continuous-forms stacker (CFS) for stacking. This parameter applies only to continuous-forms printers such as the 3800, 3900, or 4100 printer with a post-processing device installed.
>
> **Note:** With a 3800 printer, PSF always tells JES to issue burst setup messages for jobs with BURST specified, regardless of whether a BTS is installed. However, with a 3900 or 4100 printer, PSF tells JES to issue burst setup messages only if a BTS is specified as installed when the FSA is started. If a BTS is installed or removed while PSF is active, PSF cannot tell JES to issue (or not to issue) burst setup messages. To receive JES burst setup messages, always specify that a BTS is installed on the 3900 or 4100 printer whenever you first start the printer. If a BTS is installed or removed, you must restart the FSA to receive (or not receive) JES burst setup messages.

CKPTMODE=PAGE|SEC
> Specifies whether checkpointing is to be done by page count (PAGE) or by time (SEC). If you specify both CKPTPAGE and CKPTSEC, the CKPTMODE parameter determines which value is used. When you are printing to microfilm output, specify CKPTMODE=SEC. For additional information about printing to microfilm, see Chapter 29, "Supporting AFP output on microfilm," on page 341.
>
> **Note:** PSF can use only checkpoint information that is created by the same level of PSF that is currently printing the data set. For example, if a data set was checkpointed by PSF 4.6.0, but is now being printed by PSF 4.7.0, the checkpoint data is ignored and the data set is restarted from the beginning.

CKPTPAGE
> Specifies the number of pages between data set checkpoints. If checkpoint intervals are too frequent, printer performance is significantly reduced. If intervals are too infrequent, and a PSF error occurs, the job must be reprocessed from the last checkpoint. In this case, printer performance is significantly reduced if PSF errors continue to occur. For printing to paper output, IBM recommends a starting value of 100. For printing to microfilm output, specify CKPTSEC instead of CKPTPAGE. For additional information about printing to microfilm, see Chapter 29, "Supporting AFP output on microfilm," on page 341.
>
> **Note:** PSF can use only checkpoint information that is created by the same level of PSF that is currently printing the data set. For example, if a data set was checkpointed by PSF 4.6.0, but is now being printed by PSF 4.7.0, the checkpoint data is ignored and the data set is restarted from the beginning.

CKPTSEC
> Specifies the seconds between checkpoints in the data set. When you are printing to microfilm output, specify a CKPTSEC keyword instead of CKPTPAGE. For additional information about printing to microfilm, see Chapter 29, "Supporting AFP output on microfilm," on page 341.
>
> **Note:** PSF can use only checkpoint information that is created by the same level of PSF that is currently printing the data set. For example, if a data set was checkpointed by PSF 4.6.0, but is now being printed by PSF 4.7.0, the checkpoint data is ignored and the data set is restarted from the beginning.

CLASS
> Specifies the output classes processed by the printer. You might want to consider the handling of disabled mechanisms by defining print classes.

COPYMARK=DATASET|JOB|CONSTANT|NONE
> Specifies how a printer is to increment copy marks or offset stacking.

DATASET

The printer is to increment the copy marks or offset stacking on a data set boundary.

JOB

The printer is to increment the copy marks or offset stacking on a job boundary.

CONSTANT

Copy marks are constant and no offset stacking is done.

NONE

No copy marks are used and no offset stacking is done.

For more information about copy marking, see "Copy marking and offset stacking" on page 156.

FCB

Specifies the name of the default JES2 page definition. If this parameter is not included, the default page definition can be specified by an NIFCB parameter in the JES2 PRINTDEF statement.

Note: If you want to select a page definition from the PSF startup procedure instead of the JES2 default, do not specify an FCB parameter on the PRT*nnnn* statement or an NIFCB parameter in the JES2 PRINTDEF statement.

FLASH

Specifies the name of the default JES forms flash for a 3800 printer. If this parameter is not included, the default forms flash can be specified by an NIFLASH parameter on the JES2 PRINTDEF statement.

Note: If you want to use the FLASH parameter that is specified in the form definition specified in the PRINTDEV statement, do not define a FLASH parameter in the JES2 PRINTDEF statement.

FSS

Specifies the FSS for this device and must match the name specified on the corresponding FSS statement. This parameter is required.

MARK=YES|NO

Specifies that forms marking is to be placed on job trailer pages. If MARK=NO is specified, no forms marking is placed on trailer pages. For the 3800 printer only, if MARK=NO is specified, the printer marks the carrier strip rather than the forms. For more information, see "Forms marking" on page 153 and "Carrier-strip marking" on page 155.

Note: This parameter replaces the MARK|NOMARK parameter in JES2.

MODE

Specifies that the printer is managed by an FSS (MODE=FSS). This parameter is required.

Note: For a 3800 printer that is running in 3800-1 compatibility mode, specify MODE=JES.

NPRO

Specifies the interval after which output in the paper path is forced to the stacker with a nonprocess run out.

This parameter is used only for continuous-forms printers and microfilm devices. For more information about printing to microfilm, see Chapter 29, "Supporting AFP output on microfilm," on page 341.

If NPRO is nonzero when the printer is stopped and the PSF FSA stops, PSF causes the printer to do an NPRO immediately without waiting for the NPRO time to expire. This action ensures that all data reaches the printer stacker before PSF releases the last data set to JES as complete.

If NPRO=0, PSF never sends an NPRO request to the printer, even when the printer is stopped and the PSF FSA stops. For some duplex continuous-forms printers, the back sides of some pages might not be printed when the printer indicates to PSF that the last page is printed. If a print problem occurs after PSF stops processing, data might be lost. Even if you do not typically want NPROs, you might want to set a large NPRO value to ensure that all data is complete when PSF stops. This setting is recommended for duplex continuous-forms printers. The maximum NPRO value that is allowed for JES2 is 3600 seconds (1 hour).

If NPRO=0 and the PSF DISCINTV (or **Disconnect interval**) parameter is being used, PSF does not force all the pages of the last job to the stacker when the DISCINTV timer expires. Specify a nonzero NPRO value to force the last pages of the last job to the stacker when the DISCINTV timer expires. For more information about the DISCINTV parameter, see Disconnect interval.

If NPRO=0 and the printer supports the Universal Printer Pre- and Post-Processing Interface (UP3I), the last pages of the last job sent to the printer are NOT forced through the post-processing equipment when the PSF FSA is stopped. This action can result in an incomplete job that is found in the UP3I equipment. Therefore, make sure that you manually force all pages to the stacker before you stop the PSF FSA to prevent lost pages or an incomplete job in the post processor.

PRMODE
Specifies the data set processing modes supported for a printer. List all the PRMODE values that the printer is to accept.

If PRMODE is designated as a JES work-selection (WS) criterion for output selection, only jobs with a PRMODE that matches one of the following values with special meanings is selected for processing by the printer:

LINE
Specifies that line-format data sets are to be printed.

PAGE
Specifies that composed-page data sets can be printed.

SOSI*n*
Specifies a shift-out/shift-in processing mode of SOSI1, SOSI2, SOSI3, or SOSI4 for double-byte fonts.

PRMODE can be assigned to a print job on the OUTPUT JCL statement, on the PRINTDEV statement, in the Printer Inventory (**Default process mode** parameter), or in Exit 7 (XTP7PRMD). If PRMODE is not coded, JES examines the print records and assigns a value of PAGE to data sets that contain structured fields, or LINE when data sets do not contain structured fields. For this reason, if PRMODE is designated as a JES work-selection (WS) criteria, specify both LINE and PAGE for printers on which both line-format and composed-page data sets are to be printed.

For a print job that uses double-byte fonts, you must specify a PRMODE of either SOSI1, SOSI2, SOSI3, or SOSI4. If your installation uses double-byte fonts, specify the appropriate SOSI*n* values for the printers that support double-byte fonts. If PRMODE is designated as a JES work-selection criterion, all other PRMODE values that you defined in your installation to control job processing must be specified for printers. For more information about the PRMODE parameter, see *PSF for z/OS: User's Guide*.

PRESELCT=YES|NO
Specifies whether jobs are preselected for printers.

YES
Specifies that two output elements or jobs are selected for each printer, which means that while one job is printing, a second job is also assigned to that printer. This preselected job is not available for printing on any other printer.

NO
Specifies that only one output element or job are to be selected for each printer.

SEP=YES|NO
SEP=YES specifies that job-header and job-trailer separator pages are to be produced; SEP=NO specifies that no separator pages are to be produced.

Note: If a distributed print function (DPF) is installed, you might need to specify SEP=YES. For more information, see "Distributed print function (DPF) considerations" on page 153.

For microfilm devices, specify the SEP=NO parameter.

SEPDS=YES|NO
SEPDS=YES specifies that formatted data set header separator pages are to be produced; SEPDS=NO specifies that no formatted data set header separator pages are to be produced.

Note: For microfilm devices, specify the SEPDS=NO parameter.

START=<u>YES</u>|NO

START=NO specifies that the printer is to be started by operator command. START=YES specifies that the printer (if it is ready) is to start automatically when JES2 begins processing. If you specify START=YES, and the printer is unavailable when JES2 is started, the printer does not start unless you also specify the UNIT parameter to specify a unit address.

TRKCELL=YES|<u>NO</u>

Specifies whether track-cell despooling is to be used with this printer. TRKCELL=NO (the default) specifies that track-cell despooling is not used. TRKCELL=YES is recommended.

UCS

Specifies a default font. "How PSF selects resources" on page 174 shows a table of how fonts are selected in PSF.

Note: To select a font in the startup procedure instead of the JES2 default, code UCS=0 on the PRT*nnnn* statement.

UNIT

Specifies the real device address (unit address) of the printer. If you enter START and specify a unit address, the printer starts automatically when JES2 begins processing, even if the printer is offline during the initialization of JES. This parameter applies only to host-connected, channel-attached non-SNA printers.

Sample initialization statements for JES3

Figure 37 on page 129 shows sample JES3 statements for a host-connected, channel-attached, non-SNA printer (JNAME=PRT1), an SNA-attached printer (JNAME=PRT2), and a TCP/IP-attached printer (JNAME=PRT3).

Note: These statements are samples only. For specific coding for JES parameters, see the *z/OS JES3 Initialization and Tuning Reference*.

For a host-connected, channel-attached, non-SNA printer:

```
FSSDEF,TYPE=WTR,FSSNAME=FSS1,PNAME=SAMPPRO1
DEVICE,DTYPE=PRTAFP1,JNAME=PRT1,JUNIT=(00E,SYS1,UR,ON),
       XTYPE=(PRT1,UR),XUNIT=(00E,SYS1,UR,OFF),FSSNAME=FSS1,
       MODE=FSS,PM=(LINE,PAGE,SOSI1),CHARS=(YES,60D8),DYN=(NO),
       CARRIAGE=(YES,A868),CKPNTPG=100,HEADER=YES,
       BURST=(YES,M),NPRO=99,WC=(A),FORMS=(NO,NARROW),FEATURES=(SS),
       STACKER=(NO,S)

For an SNA-attached printer:

FSSDEF,TYPE=WTR,FSSNAME=FSS2,PNAME=SAMPPRO2,SYSTEM=SYS1,TERM=NO
DEVICE,DTYPE=PRTAFP1,JNAME=PRT2,JUNIT=(,SYS1,,OFF),FSSNAME=FSS2,
       MODE=FSS,PM=(LINE,PAGE,SOSI1),CHARS=(YES,60D8),
       CARRIAGE=(YES,A868),CKPNTPG=100,HEADER=YES,WC=(B)

For a TCP/IP-attached printer:

FSSDEF,TYPE=WTR,FSSNAME=FSS3,PNAME=SAMPPRO3,SYSTEM=SYS1,TERM=NO
DEVICE,DTYPE=PRTAFP1,JNAME=PRT3,JUNIT=(,SYS1,,OFF),FSSNAME=FSS3,
       MODE=FSS,PM=(LINE,PAGE,SOSI1),CHARS=(YES,60D8),
       CARRIAGE=(YES,A868),CKPNTPG=100,HEADER=YES,WC=(C)
```

Figure 37. Sample JES3 initialization statements for three printers

Figure 38 on page 130 shows sample JES3 statements for running multiple FSAs under a single FSS.

```
FSSDEF,TYPE=WTR,FSSNAME=J3PSFSW,PNAME=J3PSFSW
DEVICE,DTYPE=PRTAFP1,JNAME=PRT12,XTYPE=(PRM,UR),CKPNTPG=5,
       DGROUP=LOCAL,JUNIT=(390,SYS1,UR,OFF),XUNIT=(390,SYS1,UR,ON),
       FSSNAME=J3PSFSW,MODE=FSS,PM=(LINE,PAGE),WS=(CL),WC=(M)
DEVICE,DTYPE=PRTAFP1,JNAME=PRT17,CKPNTPG=5,
       DGROUP=VOYAGER,JUNIT=(,SYS1,,OFF,,SYS2,,OFF),
       FSSNAME=J3PSFSW,MODE=FSS,PM=(LINE,PAGE),WS=(CL),WC=(Y)
```

Continuation characters are not shown.

Figure 38. Sample initialization statements for JES3

Parameters for the JES3 initialization statement

"FSSDEF statement" on page 130 and "DEVICE statement" on page 130 describe the JES3 initialization statements that are shown in Figure 37 on page 129, which are typical of the JES3 statements used with PSF. For more detailed descriptions of all the JES3 initialization statements, see the *z/OS JES3 Initialization and Tuning Reference*.

The descriptions show which statements and parameters are required. In the programming syntax that is shown for each parameter, defaults are underscored.

Note: When coding JES initialization statements, keep the individual characteristics of each printer in mind. For example, STACKER=S applies only to printers that are using continuous forms with a burster-trimmer-stacker installed.

FSSDEF statement

The FSSDEF initialization statement is optional but recommended. If it is omitted, JES3 generates a default FSS for the printer that is started.

The FSSDEF statement can contain the following parameters:

TYPE
Specifies that the FSS is an output writer for deferred printing (TYPE=WTR). This parameter is required.

FSSNAME
Specifies the unique name of a particular FSS. This parameter is required.

PNAME
Specifies a procedure for starting a specific PSF FSS. The procedure, which must be defined before that FSS is started, is a member of the procedure library defined by the STCPROC parameter of the STANDARDS statement or of the IATPLBST procedure library (the default). Different FSSDEF initialization statements can refer to the same startup procedure.

The PNAME parameter can specify either a startup procedure supplied with PSF, or one of your organization's procedures. In the examples that are shown in Figure 37 on page 129, SAMPPRO2 is the PNAME for the host-connected, channel-attached, non-SNA printer and SAMPPRO3 is the PNAME for the SNA-attached printer. SAMPPRO1 and SAMPPRO2 are shown in "Sample PSF startup procedures for three separate printers" on page 140.

SYSTEM
Specifies the JES3 processor on which the FSS is to run. The name must be the same as the one that is specified on the NAME parameter of the MAINPROC statement for the processor.

TERM=YES|NO
YES specifies that the FSS is to be stopped if the JES3 global address space ends by a *RETURN or *DUMP operator command.

DEVICE statement

A DEVICE statement is required for each printer. The DEVICE parameters create JES3 default values that are used unless other values are specified in the JCL application program. The DEVICE statement can contain the following parameters:

BURST=(YES|NO [,M])
Specifies whether trailer pages are printed and marked.

YES
Specifies that trailer pages are to be printed.

NO
Suppresses the printing of trailer pages.

M
Specifies that the edges of the trailer pages are to be marked.

Notes:

1. This parameter is not related to the burster-trimmer-stacker (BTS) feature installed on continuous-forms printers. To specify burster-trimmer-stacker options, see STACKER.

2. If DPF is installed, you must specify BURST=(YES). For more information, see "Distributed print function (DPF) considerations" on page 153.

CARRIAGE=(YES|NO,*aaaa*)
Specifies the JES default page definition.

YES
Specifies that the page definition can be changed during startup procedure.

NO
Specifies that the page definition cannot be changed during the startup procedure.

aaaa
Specifies that the unprefixed name (1–4 characters) of the page definition is to be used as a default. The name that is specified here, or the JES3 system default, is the PSF default page definition. For information about selecting PSF defaults, see "How PSF selects resources" on page 174.

CHARS
Specifies a default JES3 font. For a chart that shows how fonts are selected in PSF, see "How PSF selects resources" on page 174.

CKPNTPG
Specifies the number of pages between data set checkpoints. Specify no more frequent checkpoints than you need to monitor a print job. If checkpoint intervals are too frequent, printer performance is significantly reduced. If intervals are too few, and a PSF error occurs, the job needs to be reprocessed from the last checkpoint. For printing to paper, a starting value of 100 is recommended. For printing to microfilm output, specify a CKPNTSEC keyword instead of CKPNTPG. For additional information about printing to microfilm, see Chapter 29, "Supporting AFP output on microfilm," on page 341.

Note: PSF can use only checkpoint information that is created by the same level of PSF that is currently printing the data set. For example, if a data set was checkpointed by PSF 4.6.0, but is now being printed by PSF 4.7.0, the checkpoint data is ignored and the data set is restarted from the beginning.

CKPNTSEC
Specifies the seconds between data set checkpoints.

Notes:

1. For printing to microfilm output, specify the CKPNTSEC parameter. For additional information about printing to microfilm, see Chapter 29, "Supporting AFP output on microfilm," on page 341.

2. PSF can use only checkpoint information that is created by the same level of PSF that is currently printing the data set. For example, if a data set was checkpointed by PSF 4.6.0, but is now being printed by PSF 4.7.0, the checkpoint data is ignored and the data set is restarted from the beginning.

COPYMARK=C|N|J
Specifies that you want vertical separator lines printed on the edges of the output or that you want offset stacking done. These separator lines help identify the beginning and end of each job or data set

in a stack of output. A change in the thickness of the copy mark indicates the beginning of a new job, a new data set, or one of multiple copies of a data set.

C

Specifies that copy marks are to be printed or offset stacking is to be done for each copy of a data set. For multiple copies of a data set, JES3 increments the thickness of the copy mark after each copy.

N

Specifies that you do not want to use copy marks or offset stacking. Although JES3 continues to print a vertical line on the edges of output from 3800 printers, the thickness does not vary.

J

Specifies that you want copy marks printed or offset stacking done on a job basis. A change in the thickness of the copy mark indicates the beginning of a new job.

For more information about copy marking, see "Copy marking and offset stacking" on page 156.

DTYPE

Identifies these printer device types:

- DTYPE=PRT38003 for a 3800 Model 3 printer
- DTYPE=PRT38003 for a 3800 Model 6 printer
- DTYPE=PRT38008 for a 3800 Model 8 printer
- DTYPE=PRT3820 for a 3820 printer
- DTYPE=PRTAFP1 for all other printers and microfilm devices

This parameter is required.

DYNAMIC=YES|NO

Specifies whether you want this device started dynamically when work is available to print, to punch, or to be sent to a remote printer or punch. For all devices except those devices that run under the control of an output writer FSS, the default is YES. For devices that run under the control of an output writer FSS, the default is NO. For more information, see the *z/OS JES3 Initialization and Tuning Guide*.

FEATURES

FEATURES=SS specifies that the printer has a burster-trimmer-stacker.

FLASH

Specifies whether forms flash is used (3800 printer only), and specifies the name of the forms overlay to be used.

Note: If you want to use the FLASH parameter that is specified in the form definition specified in the PSF PRINTDEV statement, do not define a FLASH parameter in the JES3 DEVICE statement.

FORMS

Specifies whether forms can be changed between print jobs; specifies the initial form loaded in the printer.

FSSNAME

Specifies a unique FSS for this printer DEVICE statement. The value must match the value coded for the FSSNAME parameter in the corresponding FSSDEF statement.

HEADER=YES|NO

Specifies whether job and data set header pages are printed.

For microfilm devices, specify HEADER=NO.

Note: If DPF is installed, you might need to specify HEADER=YES. For more information, see "Distributed print function (DPF) considerations" on page 153.

JNAME

Specifies the name of the printer FSA. Each printer FSA must have a unique JNAME. This unique JNAME is 8 characters or fewer and must correspond to its label on the CNTL, ENDCNTL, and PRINTDEV statements for the PSF startup procedure specified in the PNAME parameter. For more

information, see "Sample PSF startup procedures for three separate printers" on page 140. This parameter is required.

JUNIT
Specifies:

- The device address (host-connected, channel-attached, non-SNA printers only)
- The name of the processor to which the device is attached
- A destination class for messages about the device
- Whether the device is initially online or offline

This parameter is required.

Note: Do not specify a device address for SNA-attached or TCP/IP-attached printers.

MODE=FSS
Specifies that the printer is managed by an FSS. This parameter is required.

NPRO
Specifies the interval after which output in the paper path is forced to the stacker with a nonprocess run out.

This parameter is used only for continuous-forms printers and microfilm devices. For more information about printing to microfilm, see Chapter 29, "Supporting AFP output on microfilm," on page 341.

If NPRO is nonzero when the printer is stopped and the PSF FSA stops, PSF causes the printer to do an NPRO immediately without waiting for the NPRO time to expire. This action ensures that all data reaches the printer stacker before PSF releases the last data set to JES as complete.

If NPRO=0, PSF never sends an NPRO request to the printer, even when the printer is stopped and the PSF FSA stops. For some duplex continuous-forms printers, the back sides of some pages might not be printed when the printer indicates to PSF that the last page is printed. If a print problem occurs after PSF stops processing, data might be lost. Even if you do not typically want NPROs, you might want to set a large NPRO value to ensure that all data is complete when PSF stops. This setting is recommended for duplex continuous-forms printers. The maximum NPRO value that is allowed for JES3 is 9999 seconds (2 hours 46 minutes).

If NPRO=0 and the PSF DISCINTV (or **Disconnect interval**) parameter is being used, PSF does not force all the pages of the last job to the stacker when the DISCINTV timer expires. Specify a nonzero NPRO value to force the last pages of the last job to the stacker when the DISCINTV timer expires. For more information about the DISCINTV parameter, see Disconnect interval.

If NPRO=0 and the printer supports the Universal Printer Pre- and Post-Processing Interface (UP3I), the last pages of the last job sent to the printer are NOT forced through the post-processing equipment when the PSF FSA is stopped. This can result in an incomplete job that is found in the UP3I equipment. Therefore, make sure that you manually force all pages to the stacker before you stop the PSF FSA to prevent lost pages or an incomplete job in the post processor.

PDEFAULT=NONE|CHARS|FCB|CHARS,FCB
Specifies whether the JES3 default values are used during FSS processing or they are ignored and the PRINTDEV value for CHARS, FCB, or both are used.

NONE
Specifies that the FSS uses all the JES3 default values; no PRINTDEV values are used.

CHARS
Specifies that the JES3 default for characters (UCS or train) is ignored and the PRINTDEV value for CHARS is used when the WS=U parameter is specified on the DEVICE statement.

FCB
Specifies that the JES3 default for carriage (FCB or CARR) is ignored and the PRINTDEV value for PAGEDEF is used when the WS=C parameter is specified on the DEVICE statement.

CHARS,FCB

> Specifies that the JES3 default for characters (UCS or train) and carriage (FCB or CARR) are ignored. PSF uses the PRINTDEV value for CHARS when the WS=U parameter is specified on the DEVICE statement and the PRINTDEV value for PAGEDEF when the WS=C parameter is specified on the DEVICE statement

Note: JES3 always sends its default values to the FSS when PDEFAULT is not specified.

PM

> Specifies which data set processing mode is supported. PM=LINE specifies a printer that can print line format data sets. PM=PAGE specifies a printer that can print MO:DCA-P data sets. PM=SOSI1, PM=SOSI2, PM=SOSI3, or PM=SOSI4 specifies a printer that can process double-byte fonts. For more information about SOSI processing, see *PSF for z/OS: User's Guide*.

STACKER

> Specifies whether output stacking can change, and whether the paper is to be threaded to the burster-trimmer-stacker for bursting and offset stacking (S) or sent to the continuous-forms stacker (C). This parameter is only for a continuous-forms printer with a burster-trimmer-stacker installed.

WC

> Specifies the print output class that the printer is to process. You might want to consider handling disabled mechanisms by defining print classes. For more information, see "Disabled mechanisms" on page 159.

WS=(*work_selection_criteria*)

> Specifies the work-selection criteria for this FSA; separate each value with a comma. See the *z/OS JES3 Initialization and Tuning Reference* for the valid values and the default.
>
> For a PSF FSA, consider these values:
>
> **C**
> > Causes JES3 to pass the FCB name specified on the OUTPUT JCL statement to the FSA.
>
> **CL**
> > Specifies that the FSA selects only those data sets with the same class as specified in the WC parameter of this statement.
>
> **D**
> > Specifies that the FSA selects only those data sets with the same destination name as in the DGROUP parameter of this statement.
>
> **F**
> > Causes JES3 to pass the form name specified as a JCL parameter to the FSA.
>
> **U**
> > Causes JES3 to pass the font name specified as a JCL parameter to the FSA.
>
> **Notes:**
>
> 1. This parameter is optional but recommended. If you omit it, the FSA selects output data sets for processing according to default work-selection criteria.
>
> 2. If you want PSF to use the FCB specified in the FCB JCL parameter to format output, specify WS=C as one of the work-selection criteria. Otherwise, JES3 does not pass the FCB name specified in JCL to the FSA.
>
> 3. If you want PSF to use the form name that is specified in the FORMS JCL parameter to select printer definitions in the Printer Inventory, specify WS=F as one of the work-selection criteria. Otherwise, JES3 does not pass the form name specified in JCL to the FSA.
>
> 4. If you want PSF to use the page definition that is specified in the PAGEDEF (or FCB) JCL parameter, specify WS=C as one of the work-selection criteria. Otherwise, JES3 does not pass the page definition name specified in JCL to the FSA.
>
> 5. If you want PSF to use the font that is specified in the CHARS (or UCS) JCL parameter, specify WS=U as one of the work-selection criteria. Otherwise, JES3 does not pass the font name specified in JCL to the FSA.

XTYPE

Specifies the characteristics of the FSS-managed printer as it is used by jobs in execution. This parameter is valid only for channel-attached printers, and is required if the JUNIT parameter specifies a device address. XTYPE, if used, must precede the XUNIT parameter.

Note: XTYPE is not valid for SNA-attached or TCP/IP-attached printers.

XUNIT

Specifies the device address, the name of the processor to which the device is attached, a destination code for messages about the device, and the initial status of the device—online or offline. It also specifies whether the device is initially online or offline.

This parameter is valid only for channel-attached printers, and is required if the JUNIT parameter specifies a device address.

Note: XUNIT is not valid for SNA-attached or TCP/IP-attached printers.

PSF startup procedures for AFP printers

Before you start a PSF FSS, you must have a cataloged startup procedure in a system procedure library, such as SYS1.PROCLIB. This procedure specifies PSF initialization parameters and libraries that contain system and installation resources. The name of this procedure can be specified in the FSSDEF statement of the JES initialization deck. If the name is omitted, JES3 supplies a default, which is chosen for the defined printer.

Note: You can define the initialization parameters in the Printer Inventory instead of the startup procedure. Using the Printer Inventory is more efficient than using the startup procedure because when you change parameters in the Printer Inventory, you do not need to restart all the printers in the startup procedure; you need to restart only the printer for which you changed parameters. For more information about using the Printer Inventory, see Chapter 7, "Using the Infoprint Server Printer Inventory," on page 53.

Several PSF startup procedures are supplied with PSF. You can modify the startup procedures that are supplied with PSF or write your own startup. For sample JES2 and JES3 PSF startup procedures, see "Sample PSF startup procedures" on page 135. Publications about the initialization and tuning of JES contain details about procedure libraries.

When you install PSF, you must create or update the required libraries that are referenced by the startup procedure, and you must consider any need to increase the region size, as specified in the EXEC statement, to accommodate resources and an increased number of printer FSAs.

The startup procedure can also specify defaults that cannot be set with JES initialization statements for printer FSA definitions. For more information about JES defaults, see "Defining PSF FSSs and FSAs for printers" on page 123 and "How PSF selects resources" on page 174.

Sample PSF startup procedures

"Sample PSF startup procedure for four printers" on page 137 shows a sample startup procedure for four printer definitions. PRT1 is a channel-attached printer. PRT2 is an SNA-attached printer. PRT3 is any supported TCP/IP-attached printer, including any printer attached by use of the i-data IPDS Printer LAN attachment. PRT4 is a TCP/IP-attached printer that generates separator pages and message data sets that use only TrueType and OpenType fonts.

The JES2 initialization for the FSS is shown in Figure 36 on page 125. The JES2 initialization statements must have the following relationships with the associated PSF startup procedure:

- The name of the PSF startup procedure must be specified in the PROC option of the FSS statement. For this example, the procedure name is SAMPPROC.
- In the PSF startup procedure, the labels for the CNTL, PRINTDEV, and ENDCNTL statements must match the JES2 PRT*nnnn* statements associated with the FSS.

"Sample PSF startup procedures for three separate printers" on page 140 shows startup procedures for three separate printer definitions: PRT1 is a channel-attached printer, PRT2 is an SNA-attached printer, and PRT3 is a TCP/IP-attached printer.

The JES3 initialization statements for the FSSs and printer FSAs are shown in Figure 37 on page 129. The JES3 initialization statements must have the following relationships with the associated PSF startup procedure:

- The name of the PSF startup procedure must be specified in the PNAME parameter of the FSSDEF statement. In Figure 37 on page 129, the FSSDEF statement that is named FSS1 specifies procedure SAMPPRO1; FSS2 specifies procedure SAMPPRO2.
- In the PSF startup procedure, the labels for the CNTL, PRINTDEV, and ENDCNTL statements must match the JNAME values in the JES3 DEVICE statements associated with the FSS.

The fields in "Sample PSF startup procedure for four printers" on page 137 and "Sample PSF startup procedures for three separate printers" on page 140 that are commented with PI represent PSF parameters that you must define in the Printer Inventory when you are using the Printer Inventory; do not specify them in the startup procedure because PSF ignores them. To specify that you are using the Printer Inventory, INV=*piname* must be the first parameter in the PARM field of the EXEC statement:

```
//   EXEC PGM=APSPPIEP,PARM=('INV=piname')
```

where *piname* is the four-character name of the Printer Inventory. The INV=*piname* parameter indicates that PSF uses the Printer Inventory and obtains parameters from the specified Printer Inventory database for each printer in the startup procedure. All other parameters in the PARM field are ignored when PSF uses the Printer Inventory. For more information about defining parameters in the Printer Inventory, see "Specifying Printer Inventory parameters" on page 57.

Keep in mind: The examples in "Sample PSF startup procedure for four printers" on page 137 and "Sample PSF startup procedures for three separate printers" on page 140 show how startup procedures might be modified, but they do not necessarily use the recommended values. For recommended values, see the PSF sample startup procedures that are provided in SYS1.PROCLIB. Table 18 on page 136 lists the sample procedures in SYS1.PROCLIB.

Table 18. Member names for PSF startup procedures in SYS1.PROCLIB	
Member Name	**Description**
APSWPROA	PSF startup procedure for channel-attached printers. This sample specifies 240-pel resolution font libraries and all PRINTDEV defaults.
APSWPROB	PSF startup procedure for SNA-attached printers. This sample specifies 240-pel resolution font libraries and contains several comment sections that show different ways to attach the printer for sharing between other print drivers.
APSWPROC	PSF startup procedure for 3800 printers.
APSWPROF	PSF startup procedure for TCP/IP-attached printers. This sample specifies multiple resolution libraries for fonts, page segments, and overlays, and provides several printer sharing alternatives.
APSWPROM	PSF startup procedure for microfilm devices.
APSWPROR	PSF startup procedure for TCP/IP-attached printers. This sample specifies multiple resolution libraries for fonts, page segments, and overlays, and contains several comment sections that show different ways to attach the printer for sharing between other print drivers.
APSWPROT	PSF startup procedure for TCP/IP-attached printers. This sample specifies 240-pel resolutions font libraries, all PRINTDEV defaults, and does not share the printer.

Sample PSF startup procedure for four printers

```
//SAMPPROC PROC
//* THE FOLLOWING STARTUP PROCEDURE DEFINES THE JCL AND PRINTDEV
//* STATEMENTS FOR FOUR DIFFERENT PSF-SUPPORTED PRINTERS.
//*
//STEP01   EXEC PGM=APSPPIEP,REGION=4M,TIME=1440,PARM=(,,,,TCPIP,)
//*
//JOBHDR   OUTPUT PAGEDEF=V06483,     /* JOB HEADER PAGE                  */
//         FORMDEF=A10120,CHARS=60D8  /*   FORMDEF: ALTERNATIVE BIN       */
//JOBHIS3  OUTPUT PAGEDEF=TT6483      /* JOB HEADER SEPARATOR USING TRUETYPE */
//         FORMDEF=I30110             /*   FONTS SPECIFIED IN THE PAGEDEF  */
//JOBTLR   OUTPUT PAGEDEF=V06483,     /* JOB TRAILER PAGE                 */
//         FORMDEF=A10110,CHARS=60D8  /*   FORMDEF: MAIN BIN               */
//JOBTIS3  OUTPUT PAGEDEF=TT6483      /* JOB TRAILER SEPARATOR USING TRUETYPE */
//         FORMDEF=I30110             /*   FONTS SPECIFIED IN THE PAGEDEF  */
//DSHDR    OUTPUT PAGEDEF=V06483,     /* DATA SET SEPARATOR               */
//         FORMDEF=A10110,CHARS=60D8  /*   FORMDEF: MAIN BIN               */
//DSHIS3   OUTPUT PAGEDEF=TT6483      /* DATA SET SEPARATOR USING TRUETYPE */
//         FORMDEF=I30110             /*   FONTS SPECIFIED IN THE PAGEDEF  */
//MSGDS    OUTPUT PAGEDEF=A08682,     /* MESSAGE DATA SET                 */
//         FORMDEF=A10110,CHARS=60D8  /*   FORMDEF: MAIN BIN               */
//MSGIS3   OUTPUT PAGEDEF=TT6462,     /* MESSAGE DATA SET USING TRUETYPE  */
//         FORMDEF=I30110             /*   FONTS SPECIFIED IN THE PAGEDEF  */
//*
//PRTINFO DD DSN=SAMPPROC.PRTINFO,DISP=SHR   /* PRINTER INFORMATION       */
//AFPPARMS DD DSN=FSS.PDS.AFPPARMS,DISP=SHR
//*
```

Figure 39. Sample PSF startup procedure for four printers (Part 1 of 3)

```
//FONT01   DD DSN=SYS1.FONTLIBB,DISP=SHR  /* SYSTEM 240 PEL FONTS          */
//         DD DSN=SYS1.SFNTILIB,DISP=SHR  /* SYSTEM OUTLINE FONTS          */
//         DD DSN=INST.FONTLIB,DISP=SHR   /* INSTALLATION USER FONTS       */
//FONT02   DD DSN=SYS1.FONT300,DISP=SHR   /* SYSTEM 300 PEL FONTS          */
//         DD DSN=SYS1.SFNTILIB,DISP=SHR  /* SYSTEM OUTLINE FONTS          */
//         DD DSN=INST.FONTLIB,DISP=SHR   /* INSTALLATION USER FONTS       */
//*
//PSEG01   DD DSN=INST.R240.PSEGLIB,DISP=SHR /* INSTALLATION PAGE SEGMENTS */
//*                                          FORMATTED FOR 240 PEL PRINTERS
//         DD DSN=SPEC.PSEGLIB,DISP=SHR      /* SPECIAL PAGE SEGMENTS     */
//*
//PSEG02   DD DSN=INST.R300.PSEGLIB,DISP=SHR /* INSTALLATION PAGE SEGMENTS */
//*                                          FORMATTED FOR 300 PEL PRINTERS
//*
//OLAY01   DD DSN=INST.R240.OVERLIB,DISP=SHR /* INSTALLATION OVERLAYS     */
//*                                          FORMATTED FOR 240 PEL PRINTERS
//*
//OLAY02   DD DSN=INST.R300.OVERLIB,DISP=SHR /* INSTALLATION OVERLAYS     */
//*                                          FORMATTED FOR 300 PEL PRINTERS
//*
//PDEF01   DD DSN=SYS1.PDEFLIB,DISP=SHR        /* SYSTEM PAGE DEFINITIONS  */
//         DD DSN=INST.PDEFLIB,DISP=SHR        /* INSTALLATION PAGE DEFS   */
//*
//FDEF01   DD DSN=SYS1.FDEFLIB,DISP=SHR        /* SYSTEM FORM DEFINITIONS  */
//         DD DSN=INST.FDEFLIB,DISP=SHR        /* INSTALLATION FORM DEFS   */
//*
//DOF1     DD PATH='/usr/lpp/fonts/worldtype'  /* DATA OBJECT FONT LIBRARY */
//         DD PATH='/u/fonts/myfonts/'
//*
//OBJ01    DD PATH='/u/objc/doresource/'       /* OBJECT CONTAINER LIBRARY */
//         DD PATH='/u/objc/cmresource/'
//*

//PRT1     CNTL                      /* Channel-attached printer        */
//PRT1     PRINTDEV FONTDD=*.FONT01, /* DEFAULT FONT LIBRARY DD         */
//         OVLYDD=*.OLAY01,          /* DEFAULT OVERLAY LIBRARY DD      */
//         PSEGDD=*.PSEG01,          /* DEFAULT SEGMENT LIBRARY DD      */
//         PDEFDD=*.PDEF01,          /* PAGEDEF LIBRARY DD              */
//         FDEFDD=*.FDEF01,          /* FORMDEF LIBRARY DD              */
//         FONTPATH=*.DOF1,          /* DATA OBJECT FONT LIBRARY DD     */
//         OBJCPATH=*.OBJ01,         /* OBJECT CONTAINER LIBRARY DD     */
//         JOBHDR=*.JOBHDR,          /* JOB HEADER SEPARATOR OUTPUT     */
//         JOBTRLR=*.JOBTLR,         /* JOB TRAILER SEPARATOR OUTPUT    */
//         DSHDR=*.DSHDR,            /* DATA SET HEADER SEPARATOR       */
//         MESSAGE=*.MSGDS,          /* MESSAGE DATA SET OUTPUT         */
//         BUFNO=1,                  /* NUMBER OF WRITE DATA BUFFERS - PI */
//         PAGEDEF=A08682,           /* DEVICE PAGEDEF DEFAULT - PI     */
//         FORMDEF=A10110,           /* DEVICE FORMDEF DEFAULT - PI     */
//         CHARS=60D8,               /* DEVICE DEFAULT FONT - PI        */
//         PIMSG=(YES,16),           /* ACCUMULATE DATA SET MESSAGES - PI */
//         DATACK=UNBLOCK,           /* UNBLOCK DATA CHECKS - PI        */
//         TRACE=YES,                /* BUILD INTERNAL TRACE - PI       */
//PRT1     ENDCNTL
//*
//PRT2     CNTL                      /* SNA-attached printer            */
//PRT2     PRINTDEV FONTDD=*.FONT01, /* DEFAULT FONT LIBRARY DD         */
//         OVLYDD=*.OLAY01,          /* DEFAULT OVERLAY LIBRARY DD      */
//         PSEGDD=*.PSEG02,          /* DEFAULT SEGMENT LIBRARY DD      */
//         PDEFDD=*.PDEF01,          /* PAGEDEF LIBRARY DD              */
//         FDEFDD=*.FDEF01,          /* FORMDEF LIBRARY DD              */
//         JOBHDR=*.JOBHDR,          /* JOB HEADER SEPARATOR OUTPUT     */
//         JOBTRLR=*.JOBTLR,         /* JOB TRAILER SEPARATOR OUTPUT    */
//         MESSAGE=*.MSGDS,          /* MESSAGE DATA SET OUTPUT         */
//         PAGEDEF=A08682,           /* DEVICE PAGEDEF DEFAULT - PI     */
//         FORMDEF=A10110,           /* DEVICE FORMDEF DEFAULT - PI     */
//         CHARS=60D8,               /* DEVICE DEFAULT FONT - PI        */
```

Figure 40. Sample PSF startup procedure for four printers (Part 2 of 3)

```
//            PIMSG=(YES,16),           /* ACCUMULATE DATA SET MESSAGES - PI */
//            DATACK=BLOCK,             /* BLOCK DATA CHECKS - PI           */
//            TRACE=YES,                /* BUILD INTERNAL TRACE - PI        */
//            FAILURE=WCONNECT,         /* VTAM: ATTEMPT RECONNECT - PI     */
//            DISCINTV=0,               /* VTAM: NO TIME OUT - PI           */
//            MGMTMODE=IMMED,           /* VTAM: MAINTAIN SESSION - PI      */
//            APPLID=PRT2APPL,          /* VTAM: APPLICATION PGM NODE - PI  */
//            LUNAME=PRT2LU             /* VTAM: LOGICAL UNIT NODE - PI     */
//PRT2       ENDCNTL
//*
//PRT3       CNTL                      /* TCP/IP-attached printer          */
//PRT3       PRINTDEV FONTDD=*.FONT02, /* FONT LIBRARY DD                  */
//            FONT240=*.FONT01,         /* 240 PEL FONT LIBRARY DD          */
//            FONT300=*.FONT02,         /* 300 PEL FONT LIBRARY DD          */
//            OVLYDD=*.OLAY01,          /* OVERLAY LIBRARY DD               */
//            OVLY240=*.OLAY01,         /* 240 PEL OVERLAY LIBRARY DD       */
//            OVLY300=*.OLAY02,         /* 300 PEL OVERLAY LIBRARY DD       */
//            PSEGDD=*.PSEG01,          /* SEGMENT LIBRARY DD               */
//            PSEG240=*.PSEG01,         /* 240 PEL SEGMENT LIBRARY DD       */
//            PSEG300=*.PSEG02,         /* 300 PEL SEGMENT LIBRARY DD       */
//            PDEFDD=*.PDEF01,          /* PAGEDEF LIBRARY DD               */
//            FDEFDD=*.FDEF01,          /* FORMDEF LIBRARY DD               */
//            FONTPATH=*.DOF1,          /* DATA OBJECT FONT LIBRARY DD      */
//            JOBHDR=*.JOBHDR,          /* JOB HEADER SEPARATOR OUTPUT      */
//            JOBTRLR=*.JOBTLR,         /* JOB TRAILER SEPARATOR OUTPUT     */
//            MESSAGE=*.MSGDS,          /* MESSAGE DATA SET OUTPUT          */
//            PAGEDEF=A08682,           /* DEVICE PAGEDEF DEFAULT - PI      */
//            FORMDEF=A10110,           /* DEVICE FORMDEF DEFAULT - PI      */
//            CHARS=60D8,               /* DEVICE DEFAULT FONT - PI         */
//            PIMSG=(YES,16),           /* ACCUMULATE DATA SET MESSAGES - PI */
//            DATACK=BLOCK,             /* BLOCK DATA CHECKS - PI           */
//            TRACE=YES,                /* BUILD INTERNAL TRACE - PI        */
//            FAILURE=WCONNECT,         /* ATTEMPT RECONNECT - PI           */
//            DISCINTV=0,               /* NO TIME OUT - PI                 */
//            MGMTMODE=IMMED,           /* MAINTAIN SESSION - PI            */
//            IPADDR='192.0.2.33'       /* IP ADDRESS OF PRINTER - PI       */
//            SETUP=FORMS               /* SETUP MESSAGE - PI               */
//PRT3       ENDCNTL
//*
//PRT4       CNTL                      /* TCP/IP-attached printer for      */
//*                                       MO:DCA IS3 level AFP jobs
//PRT4       PRINTDEV FONTDD=*.FONT02, /* FONT LIBRARY DD                  */
//            FONT240=*.FONT01,         /* 240 PEL FONT LIBRARY DD          */
//            FONT300=*.FONT02,         /* 300 PEL FONT LIBRARY DD          */
//            FONTPATH=*.DOF1,          /* TrueType FONT LIBRARY DD         */
//            OVLYDD=*.OLAY02,          /* OVERLAY LIBRARY DD               */
//            OVLY240=*.OLAY01,         /* 240 PEL OVERLAY LIBRARY DD       */
//            OVLY300=*.OLAY02,         /* 300 PEL OVERLAY LIBRARY DD       */
//            PSEGDD=*.PSEG02,          /* SEGMENT LIBRARY DD               */
//            PSEG240=*.PSEG01,         /* 240 PEL SEGMENT LIBRARY DD       */
//            PSEG300=*.PSEG02,         /* 300 PEL SEGMENT LIBRARY DD       */
//            PDEFDD=*.PDEF01,          /* PAGEDEF LIBRARY DD               */
//            FDEFDD=*.FDEF01,          /* FORMDEF LIBRARY DD               */
//            MESSAGE=*.MSGIS3,         /* MESSAGE DATA SET OUTPUT          */
//            JOBHDR=*.JOBHIS3,         /* JOB HEADER SEPARATOR OUTPUT      */
//            JOBTRLR=*.JOBTIS3,        /* JOB TRAILER SEPARATOR OUTPUT     */
//            DSHDR=*.DSHIS3,           /* DATASET SEPARATOR                */
//            FORMDEF=I30110,           /* DEVICE FORMDEF DEFAULT - PI      */
//            TRACE=YES,                /* BUILD INTERNAL TRACE - PI        */
//            DATACK=BLOCK,             /* BLOCK DATA CHECKS - PI           */
//            PIMSG=(YES,16),           /* ACCUMULATE DATA SET MESSAGES - PI */
//            FAILURE=WCONNECT,         /* ATTEMPT RECONNECT - PI           */
//            DISCINTV=0,               /* NO TIME OUT - PI                 */
//            MGMTMODE=IMMED,           /* MAINTAIN SESSION - PI            */
//            IPADDR='192.0.2.34'       /* IP ADDRESS OF PRINTER - PI       */
//            SETUP=FORMS               /* SETUP MESSAGE - PI               */
//PRT4       ENDCNTL
```

Figure 41. Sample PSF startup procedure for four printers (Part 3 of 3)

Sample PSF startup procedures for three separate printers

```
//SAMPPRO1 PROC
//* THE FOLLOWING STARTUP PROCEDURE DEFINES THE JCL AND PRINTDEV
//* STATEMENTS FOR PRT1, A CHANNEL-ATTACHED PRINTER.
//*
//STEP01   EXEC PGM=APSPPIEP,REGION=4M,PARM=(,INTR,,,,)
//JOBHDR OUTPUT PAGEDEF=V06483,          /* JOB HEADER PAGE            */
//       FORMDEF=A10120,CHARS=60D8       /*   FORMDEF: ALTERNATIVE BIN */
//JOBTLR OUTPUT PAGEDEF=V06483,          /* JOB TRAILER PAGE           */
//       FORMDEF=A10110,CHARS=60D8       /*   FORMDEF: MAIN BIN        */
//DSHDR  OUTPUT PAGEDEF=V06483,          /* DATA SET SEPARATOR         */
//       FORMDEF=A10110,CHARS=60D8       /*   FORMDEF: MAIN BIN        */
//MSGDS  OUTPUT PAGEDEF=A08682,          /* MESSAGE DATA SET           */
//       FORMDEF=A10110,CHARS=60D8       /*   FORMDEF: MAIN BIN        */
//*
//PRTINFO DD DSN=SAMPPRO1.PRTINFO,DISP=SHR   /* PRINTER INFORMATION    */
//*
//FONT01  DD DSN=SYS1.FONTLIBB,DISP=SHR     /* SYSTEM FONTS            */
//        DD DSN=INST.FONTLIB,DISP=SHR      /* INSTALLATION USER FONTS */
//*
//PSEG01  DD DSN=INST.PSEGLIB,DISP=SHR      /* INSTALLATION PAGE SEGMENTS*/
//        DD DSN=SPEC.PSEGLIB,DISP=SHR      /* SPECIAL PAGE SEGMENTS   */
//*
//OLAY01  DD DSN=INST.OVERLIB,DISP=SHR      /* INSTALLATION OVERLAYS   */
//*
//PDEF01  DD DSN=SYS1.PDEFLIB,DISP=SHR      /* SYSTEM PAGE DEFINITIONS */
//        DD DSN=INST.PDEFLIB,DISP=SHR      /* INSTALLATION PAGE DEFS  */
//*
//FDEF01  DD DSN=SYS1.FDEFLIB,DISP=SHR      /* SYSTEM FORM DEFINITIONS */
//        DD DSN=INST.FDEFLIB,DISP=SHR      /* INSTALLATION FORM DEFS  */
//*
//DOF1    DD PATH='/usr/lpp/fonts/worldtype' /* DATA OBJECT FONT LIBRARY */
//        DD PATH='/u/fonts/ttypefont/'
//        DD PATH='/u/fonts/myfonts/'
//*
//PRT1     CNTL
//PRT1     PRINTDEV FONTDD=*.FONT01, /* FONT LIBRARY DD                 */
//            OVLYDD=*.OLAY01,        /* OVERLAY LIBRARY DD             */
//            PSEGDD=*.PSEG01,        /* SEGMENT LIBRARY DD             */
//            PDEFDD=*.PDEF01,        /* PAGEDEF LIBRARY DD             */
//            FDEFDD=*.FDEF01,        /* FORMDEF LIBRARY DD             */
//            FONTPATH=*.DOF1,        /* DATA OBJECT FONT LIBRARY DD    */
//            JOBHDR=*.JOBHDR,        /* JOB HEADER SEPARATOR OUTPUT    */
//            JOBTRLR=*.JOBTLR,       /* JOB TRAILER SEPARATOR OUTPUT   */
//            DSHDR=*.DSHDR,          /* DATA SET HEADER SEPARATOR      */
//            MESSAGE=*.MSGDS,        /* MESSAGE DATA SET OUTPUT        */
//            BUFNO=1,                /* NUMBER OF WRITE DATA BUFFERS - PI */
//            PAGEDEF=A08682,         /* DEVICE PAGEDEF DEFAULT - PI    */
//            FORMDEF=A10110,         /* DEVICE FORMDEF DEFAULT - PI    */
//            CHARS=60D8,             /* DEVICE DEFAULT FONT - PI       */
//            PIMSG=(YES,16),         /* ACCUMULATE DATA SET MESSAGES - PI */
//            DATACK=UNBLOCK,         /* UNBLOCK DATA CHECKS - PI       */
//            TRACE=YES,              /* BUILD INTERNAL TRACE - PI      */
//*  DEFAULT PAGEDEF AND CHARS ARE SPECIFIED BY JES3                   */
//*          PAGEDEF=A868            /* COPY OF A08682 - PI            */
//*          CHARS=60D8             /* DEVICE DEFAULT FONT - PI        */
//PRT1     ENDCNTL
```

Figure 42. Sample PSF startup procedure for channel-attached printer

```
//SAMPPRO2 PROC
//* THE FOLLOWING STARTUP PROCEDURE DEFINES THE JCL AND PRINTDEV
//* STATEMENTS FOR PRT2, AN SNA-ATTACHED PRINTER.
//*
//STEP01   EXEC PGM=APSPPIEP,REGION=4M,TIME=1440,PARM=(,INTR)
//JOBHDR OUTPUT PAGEDEF=V06483,         /* JOB HEADER PAGE              */
//        FORMDEF=A10120,CHARS=60D8    /*    FORMDEF: ALTERNATIVE BIN  */
//JOBTLR OUTPUT PAGEDEF=V06483,         /* JOB TRAILER PAGE            */
//        FORMDEF=A10110,CHARS=60D8    /*    FORMDEF: MAIN BIN         */
//DSHDR  OUTPUT PAGEDEF=V06483,         /* DATA SET SEPARATOR          */
//        FORMDEF=A10110,CHARS=60D8    /*    FORMDEF: MAIN BIN         */
//MSGDS  OUTPUT PAGEDEF=A08682,         /* MESSAGE DATA SET            */
//        FORMDEF=A10110,CHARS=60D8    /*    FORMDEF: MAIN BIN         */
//*
//PRTINFO DD DSN=SAMPPRO2.PRTINFO,DISP=SHR   /* PRINTER INFORMATION    */
//*
//FONT01  DD DSN=SYS1.FONTLIBB,DISP=SHR  /* SYSTEM FONTS               */
//         DD DSN=INST.FONTLIB,DISP=SHR   /* INSTALLATION USER FONTS   */
//*
//PSEG02  DD DSN=INST.PSEGLIB,DISP=SHR   /* INSTALLATION PAGE SEGMENTS */
//*
//OLAY01  DD DSN=INST.OVERLIB,DISP=SHR   /* INSTALLATION OVERLAYS      */
//*
//PDEF01  DD DSN=SYS1.PDEFLIB,DISP=SHR   /* SYSTEM PAGE DEFINITIONS    */
//         DD DSN=INST.PDEFLIB,DISP=SHR   /* INSTALLATION PAGE DEFS    */
//*
//FDEF01  DD DSN=SYS1.FDEFLIB,DISP=SHR   /* SYSTEM FORM DEFINITIONS    */
//         DD DSN=INST.FDEFLIB,DISP=SHR   /* INSTALLATION FORM DEFS    */
//*
//PRT2     CNTL
//PRT2     PRINTDEV FONTDD=*.FONT01, /* FONT LIBRARY DD                */
//           OVLYDD=*.OLAY01,            /* OVERLAY LIBRARY DD         */
//           PSEGDD=*.PSEG02,            /* SEGMENT LIBRARY DD         */
//           PDEFDD=*.PDEF01,            /* PAGEDEF LIBRARY DD         */
//           FDEFDD=*.FDEF01,            /* FORMDEF LIBRARY DD         */
//           JOBHDR=*.JOBHDR,            /* JOB HEADER SEPARATOR OUTPUT */
//           JOBTRLR=*.JOBTLR,           /* JOB TRAILER SEPARATOR OUTPUT */
//           DSHDR=*.DSHDR,              /* DATA SET HEADER SEPARATOR  */
//           MESSAGE=*.MSGDS,            /* MESSAGE DATA SET OUTPUT    */
//           FORMDEF=A10110,             /* DEVICE FORMDEF DEFAULT - PI */
//           PIMSG=(YES,16),             /* ACCUMULATE DATA SET MESSAGES - PI */
//           DATACK=BLOCK,               /* BLOCK DATA CHECKS - PI     */
//           TRACE=YES,                  /* BUILD INTERNAL TRACE - PI  */
//           FAILURE=WCONNECT,           /* VTAM: ATTEMPT RECONNECT - PI */
//           DISCINTV=0,                 /* VTAM: NO TIME OUT - PI     */
//           MGMTMODE=IMMED,             /* VTAM: MAINTAIN SESSION - PI */
//           APPLID=PRT2APPL,            /* VTAM: APPLICATION PGM NODE - PI */
//           LUNAME=PRT2LU               /* VTAM: LOGICAL UNIT NODE - pi */
//*  DEFAULT PAGEDEF AND CHARS ARE SPECIFIED BY JES3                   */
//*           PAGEDEF=A868                /* COPY OF A08682 - PI        */
//*           CHARS=60D8                  /* JES3 DEFAULT FONT - PI     */
//PRT2     ENDCNTL
```

Figure 43. Sample PSF startup procedure for SNA-attached printer

```
//SAMPPRO3 PROC
//* THE FOLLOWING STARTUP PROCEDURE DEFINES THE JCL AND PRINTDEV
//* STATEMENTS FOR PRT3, A TCP/IP-ATTACHED PRINTER.
//*
//STEP01    EXEC PGM=APSPPIEP,REGION=4M,TIME=1440,PARM=(,,,,TCPIP,)
//JOBHDR    OUTPUT PAGEDEF=V06483,        /* JOB HEADER PAGE             */
//          FORMDEF=A10120,CHARS=60D8   /*   FORMDEF: ALTERNATIVE BIN   */
//JOBTLR    OUTPUT PAGEDEF=V06483,        /* JOB TRAILER PAGE            */
//          FORMDEF=A10110,CHARS=60D8   /*   FORMDEF: MAIN BIN          */
//DSHDR     OUTPUT PAGEDEF=V06483,        /* DATA SET SEPARATOR          */
//          FORMDEF=A10110,CHARS=60D8   /*   FORMDEF: MAIN BIN          */
//MSGDS     OUTPUT PAGEDEF=A08682         /* MESSAGE DATA SET            */
//          FORMDEF=A10110,CHARS=60D8   /* FORMDEF: MAIN BIN            */
//*
//PRTINFO DD DSN=SAMPPRO3.PRTINFO,DISP=SHR   /* PRINTER INFORMATION      */
//*
//FONT01  DD DSN=SYS1.FONTLIBB,DISP=SHR      /* SYSTEM FONTS             */
//        DD DSN=INST.FONTLIB,DISP=SHR       /* INSTALLATION USER FONTS  */
//*
//PSEG02  DD DSN=INST.PSEGLIB,DISP=SHR       /* INSTALLATION PAGE SEGMENTS*/
//*
//OLAY01  DD DSN=INST.OVERLIB,DISP=SHR       /* INSTALLATION OVERLAYS    */
//*
//PDEF01  DD DSN=SYS1.PDEFLIB,DISP=SHR       /* SYSTEM PAGE DEFINITIONS  */
//        DD DSN=INST.PDEFLIB,DISP=SHR       /* INSTALLATION PAGE DEFS   */
//*
//FDEF01  DD DSN=INST.FDEFLIB,DISP=SHR       /* SYSTEM FORM DEFINITIONS  */
//        DD DSN=INST.FDEFLIB,DISP=SHR       /* INSTALLATION FORM DEFS   */
//*
//DOF2    DD PATH='/usr/lpp/fonts/worldtype' /* DATA OBJECT FONT LIBRARY */
//        DD PATH='/u/fonts/ttypefont/'
//*
//OBJ01   DD PATH='/u/objc/doresource/'  /* OBJECT CONTAINER LIBRARY */
//        DD PATH='/u/objc/cmresource/'
//*
//PRT3     CNTL
//PRT3     PRINTDEV FONTDD=*.FONT01, /* FONT     LIBRARY DD               */
//           OVLYDD=*.OLAY01,           /* OVERLAY  LIBRARY DD           */
//           PSEGDD=*.PSEG02,           /* SEGMENT  LIBRARY DD           */
//           PDEFDD=*.PDEF01,           /* PAGEDEF  LIBRARY DD           */
//           FDEFDD=*.FDEF01,           /* FORMDEF  LIBRARY DD           */
//           FONTPATH=*.DOF2,           /* DATA OBJECT FONT LIBRARY DD   */
//           OBJCPATH=*.OBJ01,          /* OBJECT CONTAINER LIBRARY DD   */
//           JOBHDR=*.JOBHDR,           /* JOB HEADER SEPARATOR OUTPUT   */
//           JOBTRLR=*.JOBTLR,          /* JOB TRAILER SEPARATOR OUTPUT  */
//           DSHDR=*.DSHDR,             /* DATA SET HEADER SEPARATOR     */
//           MESSAGE=*.MSGDS,           /* MESSAGE DATA SET OUTPUT       */
//           FORMDEF=A10110,            /* DEVICE FORMDEF DEFAULT - PI   */
//           PIMSG=(YES,16),            /* ACCUMULATE DATA SET MESSAGES - PI */
//           DATACK=BLOCK,              /* BLOCK DATA CHECKS - PI        */
//           TRACE=YES,                 /* BUILD INTERNAL TRACE - PI     */
//           FAILURE=WCONNECT,          /* ATTEMPT RECONNECT - PI        */
//           DISCINTV=0,                /* NO TIME OUT - PI              */
//           MGMTMODE=IMMED,            /* MAINTAIN SESSION - PI         */
//           IPADDR='192.0.2.33'        /* IP ADDRESS OF PRINTER - PI    */
//           SETUP=FORMS                /* SETUP MESSAGE - PI            */
//*  DEFAULT PAGEDEF AND CHARS ARE SPECIFIED BY JES3               */
//*         PAGEDEF=A868               /* COPY OF A08682 - PI           */
//*         CHARS=60D8                 /* JES3 DEFAULT FONT - PI         */
//PRT3     ENDCNTL
```

Figure 44. Sample PSF startup procedure for TCP/IP-attached printer

JCL for PSF startup procedures

This information describes the statements and the parameters shown in "Sample PSF startup procedure for four printers" on page 137 and "Sample PSF startup procedures for three separate printers" on page 140.

nnnnnnnn PROC
   Specifies the name of the startup procedure. The procedure name must be specified in the JES2 FSS or the JES3 FSSDEF initialization statement.

EXEC
   Specifies the name of the PSF program and establishes the size of the region, timeout limits, and PARM options. For more information about region size, see "Selecting region sizes" on page 194. To

prevent TIMEOUT abends, you can include TIME=NOLIMIT. For a description of the options you can specify in the PARM field, see "PARM parameters on the EXEC statement" on page 144.

OUTPUT
Specifies the page definitions, form definitions, and fonts that are used to format and print different pages in a print job or data set. In the examples, the pages include job and data set header pages, job-trailer pages, and pages used for printing messages.

Table 19 on page 143 shows which keywords are *accepted* on OUTPUT statements in the PSF startup procedure. All other keywords that are not listed in the table are *ignored* on OUTPUT statements in the PSF startup procedure.

Table 19. Accepted keywords on OUTPUT statements				
Keyword	**Job Header**	**Job Trailer**	**Data Set Header**	**Message Data Set**
CHARS	Accepted	Accepted	Accepted	Accepted
CLASS	Ignored	Ignored	Ignored	Accepted
COLORMAP	Accepted	Accepted	Accepted	Accepted
COMSETUP	Accepted	Accepted	Accepted	Accepted
COPIES	Accepted	Accepted	Accepted	Accepted
DEST	Ignored	Ignored	Ignored	Accepted
FORMDEF	Accepted	Accepted	Accepted	Accepted
PAGEDEF	Accepted	Accepted	Accepted	Accepted
PIMSG	Ignored	Ignored	Ignored	Accepted
RESFMT	Accepted	Accepted	Accepted	Accepted
TRC	Accepted	Accepted	Accepted	Accepted

For information about how to specify OUTPUT so that the messages are not printed, see "Redirecting the message data set" on page 328.

DD
Identifies the system libraries that contain resources (fonts, page segments, overlays, page definitions, and form definitions) used for printing. Also identifies the data set where printer information is saved.

Note: If you are using Overlay Generation Language, any 3800 double-byte fonts must be in a library separate from other double-byte fonts.

CNTL
Specifies the beginning of program control statements for the printer. This statement must precede the PRINTDEV statement, and has no parameters.

For JES3, the label on the CNTL statement must correspond to the label on the ENDCNTL and PRINTDEV statements and to the name specified on the JNAME parameter on the JES3 DEVICE initialization statement. For JES2, the label on the CNTL, ENDCNTL, and PRINTDEV statements must correspond to the printer name on the PRT*nnnn* initialization statement.

PRINTDEV
Specifies printer-initialization parameters. A PRINTDEV statement is entered for each printer FSA belonging to the FSS that names this PSF startup procedure. The label on the PRINTDEV statement must match the label on the JES2 PRT*nnnn* initialization statement, or the JNAME parameter on the JES3 DEVICE initialization statement.

For sample JES2 and JES3 initialization statements, see Figure 36 on page 125 and Figure 37 on page 129. For a description of the PRINTDEV parameters, see "Specifying PRINTDEV parameters" on page 102. Many of the PRINTDEV parameters can be defined in the Printer Inventory. For more information

about using Printer Inventory parameters instead of PRINTDEV parameters, see Chapter 7, "Using the Infoprint Server Printer Inventory," on page 53.

ENDCNTL
Specifies the end of program control statements for the printer.

This statement must follow the PRINTDEV statement and has no parameters.

The label on the ENDCNTL statement must match the label on the CNTL and PRINTDEV statements and the PRT*nnnn* statement (under JES2), or the JNAME parameter (under JES3).

PARM parameters on the EXEC statement

The parameters that you specify in the PARM field of the EXEC statement depend on whether the Printer Inventory is used:

Printer Inventory not used:
Some optional PSF parameters can be specified in the PARM field of the EXEC statement when the Printer Inventory is not used:

```
//  EXEC PGM=APSPPIEP,PARM=(NSTddname,trace_type,prompt,trace_size,tcpip_name,UNICODE,
blank_truncation,restore_blanks_classes)
```

Notes:

1. You must specify the parameters in the correct order in the PARM field. If you do not want to specify a value for a specific parameter, type a comma in its place. For example, if you want to specify PSF prompting, type:

   ```
   //  EXEC PGM=APSPPIEP,PARM=(,,PROMPT)
   ```

2. The UNICODE parameter is no longer used. If you specify the value, it is ignored.

3. If you need more than the limit of 100 characters or prefer, you can replace PARM= with PARMDD= to specify these parameters. See the *z/OS MVS JCL Reference* for more information on the syntax for using PARMDD on the EXEC statement.

Printer Inventory used:
To use the Printer Inventory, you must specify the Printer Inventory name as the first parameter in the PARM field:

```
//  EXEC PGM=APSPPIEP,PARM=('INV=piname')
```

For more information about using the Printer Inventory, see Chapter 7, "Using the Infoprint Server Printer Inventory," on page 53. For more information about the PARM parameters you can specify in the Printer Inventory, see **NST trace dsname**, **TCP/IP job name**, **Trace mode**, **Trace prompt**, and **Trace table size** in Table 6 on page 59.

The PARM parameters are:

INV=*piname*
Specifies the four-character name of the Printer Inventory that is specified in the Infoprint Server configuration file. This parameter indicates that PSF uses the Printer Inventory and obtains parameters from the specified Printer Inventory for each printer in the startup procedure.

NSTddname
Specifies the name of a DD statement that defines the output data set in which to record the NST trace output. This name must conform to the standard JCL DD naming conventions.

This parameter specifies that the NST trace is to start during PSF initialization if you specified TRACE=YES on the PRINTDEV statement and defined the NST trace data set in a DD statement. If you want the NST trace to be started dynamically, do not specify the *NSTddname* parameter.

The equivalent parameter in the Printer Inventory is the **NST trace dsname** parameter.

trace_type
Specifies the FSA trace to start:

FULL
Specifies an FSA full external trace.

SYNC
Specifies an FSA SYNC external trace.

INTR
Specifies an FSA internal trace and, if the printer is SNA-attached, a PPCC internal trace. INTR is the default value.

LIMIT
Specifies a shortened FSA external trace.

IPDS
Specifies an FSA IPDS external trace.

The equivalent parameter in the Printer Inventory is the **Trace mode** parameter.

Notes:

1. For FULL, LIMIT, and IPDS, an FSA internal trace is also started.
2. If you want to start the FSA trace dynamically, use the default or specify INTR.

prompt
Specifies whether an operator response is required to initialize the PSF operator interface:

PROMPT
Specifies that each time the FSS is initialized, the operator is to receive a message, APS620A, that prompts the operator to issue a response, which notifies PSF to initialize the PSF operator interface. The response can be any PSF operator interface command; it is directed to all FSAs or to the notify subtask.

Thus the operator can type commands, such as those to start FSI or FSA component traces, before PSF starts processing data sets. Prompting occurs even if the startup procedure does not include tracing specifications.

NOPROMPT
Specifies that the PSF operator interface is to be initialized automatically. No operator response is required. NOPROMPT is the default value.

The equivalent parameter in the Printer Inventory is the **Trace prompt** parameter.

trace_size
Specifies the number of 4 KB pages of storage to allocate for each internal FSA trace table. Valid values are 1–999. The default is 32 (128 KB). This allocation occurs only if PSF internal tracing is active.

Note: When the number of pages that are specified is more than 32, and the result of adding the REGION increase to the current REGION specification on the EXEC statement is greater than 32 MB, increase the REGION size on your EXEC statement to this new total value. If the new value of REGION causes storage problems, use REGIONX instead of REGION to specify how storage is distributed above and below the 16 MB line.

To determine how large an increase is needed due to tracing, use the following equation, where *number of FSAs active* is the maximum number of FSAs active while the FSS is running:

```
REGION increase = (number of FSAs active) x 4 KB x (pagecount - 32)
```

The equivalent parameter in the Printer Inventory is the **Trace table size** parameter.

tcpip_name
Specifies the name of the TCP/IP address space. If this parameter is not coded, PSF uses the default name of TCPIP.

The equivalent parameter in the Printer Inventory is the **TCP/IP job name** parameter.

UNICODE
Specifies that PSF is enabled to use the system conversion services that z/OS provides. This parameter is ignored because PSF 4.5 or later is always Unicode-enabled. The equivalent parameter in the Printer Inventory is the **Unicode enabled** parameter.

blank_truncation
Specifies whether trailing blanks are truncated for non-page-mode records received from JES2:

NOBLNKTRNC
Trailing blanks are **not** truncated for all non-page-mode data sets.

BLNKTRNC
Trailing blanks are truncated for all non-page-mode data sets. BLNKTRNC is the default value.

Non-page-mode data sets include line-mode data, text data, mixed-mode data, and XML data. If you specify this parameter for composed-text page data sets (for example, output created by the Document Composition Facility), it has no effect. Note that the PRMODE= parameter, on both the JCL and JES2 output device initialization statements, does not affect blank truncation. To turn off blank truncation for some SYSOUT classes, see the **restore_blanks_classes** parameter.

Specify NOBLNKTRNC when all data sets processed by this FSS require right-hand blank padding to print properly.

The equivalent parameter in the Printer Inventory is the **Blank truncation** parameter.

restore_blanks_classes
This parameter is only used when `blank_truncation = BLNKTRNC`. It specifies the SYSOUT classes for which you do **not** want trailing blanks truncated for non-page-mode data sets. Non-page-mode data sets include line-mode data, text data, mixed-mode data, and XML data.

The value is a string of 1–36 numeric (0–9) or uppercase alphabetic (A–Z) characters. Each character represents a SYSOUT class. Do not separate the class values by commas or blanks. The value of the SYSOUT class of the data set received from JES is compared to the **restore_blanks_classes** value. If the SYSOUT class matches one of these classes, trailing blanks are not truncated.

Specify the same classes for which you specified BLNKTRNC=NO on the JES2 OUTCLASS(*v*) statement before z/OS 2.4. For example, if your JES2 OUTCLASS statement specified BLNKTRNC=NO for:

- OUTCLASS(K,L,Y), specify `restore_blanks_classes = KLY`
- OUTLCLASS(S-Z), specify `restore_blanks_classes = STUVWXYZ`

Alternatively, specify this parameter for SYSOUT classes that have:

- Non-page-mode data sets that require trailing blanks to be preserved
- Data sets containing double-byte font characters

The equivalent parameter in the Printer Inventory is the **Restore blanks classes** parameter.

Starting PSF FSAs

Under JES2, a PSF FSA can start in either of two ways:

- An operator can enter the START DEVICE JES2 command:

  $S *fsa_name*, where *fsa_name* is the name of the PRT*nnnn* statement, such as PRT0001.
- When JES2 is started, PSF starts automatically if the PRT*nnnn* statement for the printer FSA contains the START=YES parameter.

Under JES3, a PSF FSA can start in either of two ways:

- An operator issues the CALL WRITER JES3 command:

  CALL WTR,OUT=fsa_name, where *fsa_name* is the JNAME parameter of the DEVICE statement.

- When printing is scheduled for a PSF printer, JES3 starts PSF.

    If you do not want PSF to stop after each job, you can define the PSF FSS as a JES3-called writer, keeping the address space active.

Stopping PSF FSAs and FSSs

You can stop a PSF FSA in several ways:

- Use the following JES2 or JES3 commands to stop the FSA after the current data set is finished printing:

    JES2:

    ```
    $P fsa_name
    ```

    where *fsa_name* is the name of the PRT*nnnn* statement, such as PRT0001. The $P command stops the FSA after the current data set is finished printing.

    JES3:

    ```
    *VARY,fsa_name,OFFLINE
    *CANCEL,fsa_name
    ```

    where *fsa_name* is the JNAME parameter of the DEVICE statement. The *VARY command makes the FSA unavailable for printing data sets but allows the current data set to finish; the *CANCEL command stops the FSA.

- Use the following JES2 or JES3 commands to stop an FSA immediately:

    JES2:

    ```
    $P fsa_name
    $I fsa_name
    ```

    or

    ```
    $P fsa_name
    $E fsa_name
    ```

    where *fsa_name* is the name of the PRT*nnnn* statement, such as PRT0001. The $P command stops the FSA after the current data set; the $I command interrupts the current data set and requeues the job to resume printing on the page where it was interrupted; and the $E command interrupts the current data set and requeues the job to resume printing from the beginning. Both the $P command and either the $I command or $E command must be used together to stop an FSA immediately.

    JES3:

    ```
    *CANCEL,fsa_name,T
    ```

    where *fsa_name* is the JNAME parameter of the DEVICE statement. The *CANCEL command cancels the current data set; the T option stops the FSA after the current data set is canceled.

- If you are unable to stop or cancel a printer by use of the JES commands, enter this command to stop the printer FSA:

    ```
    {MODIFY|F} fss_name,FORCE,fsa_name
    ```

    where *fss_name* is the name of the FSS that manages the printer FSA you want to stop and *fsa_name* is the printer FSA you want to stop.

    If the FORCE command is unsuccessful on JES3 systems, use this JES3 command to stop the printer FSA:

    ```
    *FAIL,PRTXX
    ```

You can use the following JES2 or JES3 commands to stop a PSF FSS after all FSAs under its control are stopped:

JES2:
> If the FSS was configured with the AUTOSTOP=YES option on the JES2 FSS(*fsa_name*) statement, JES2 stops the FSS automatically after you stopped all FSAs. If AUTOSTOP=YES was not specified, you must stop the FSS with this command:

```
C fss_name
```

JES3:
> JES automatically stops the FSS after you stop all FSAs under its control.

Issuing other operator commands

In order for JES operator commands to take effect immediately on any printer, regardless of attachment, you must first stop the printer and clear the printer buffers. See your printer documentation to see whether the printer has this capability and how to do it.

Attention: When you issue any JES operator command, use the CLEAR BUFFERS option at the printer before you issue the JES command; otherwise, pages might be lost. Pages might also be lost if you use the PRINT ADJUST/ONE PAGE PRINT function without using the CLEAR BUFFERS option as the last function at the printer.

For example, if the JES RESTART command was issued, the PSF message data set (if present) or the job trailer page might be lost. After the job is restarted, the message data set and job trailer page are printed when the job is complete.

Identifying output from a printer

To identify all output from a printer with an overlay to be printed on every sheet of output, including separator pages, specify the OVERLAY keyword on the PRINTDEV statement. This keyword might be used when you are running on a test system to identify the output as test data.

Separating jobs

Depending on the printer model, users' jobs can be separated by these:

- Job separator pages
- Forms marking
- Carrier-strip marking
- Copy-marking or offset stacking

Job separation is controlled by using JES initialization statements or JES operator commands.

Job separator pages

PSF creates and prints job-header and job-trailer separator pages before and after each print job. Whether job separator pages are printed is specified with the following JES initialization statements:

- Under JES2, the SEP parameter determines whether job-header and job-trailer separator pages are printed.
- Under JES3, the HEADER parameter specifies printing of job-header pages, and the BURST parameter specifies job-trailer pages.

For additional information, see "Parameters for the JES2 initialization statement" on page 125 and "Parameters for the JES3 initialization statement" on page 130.

The content of the header and trailer pages is controlled through PSF installation Exit 1 (APSUX01 assembly-language exit or APSUC01 C-language exit) and Exit 2 (APSUX02 or APSUC02). PSF supplies default versions of these exits. Figure 45 on page 149 shows a sample header page produced by the Exit 1 module supplied with PSF.

Figure 45. Example of a default PSF job-header page (Exit 1)

PSF also provides optional replacements for Exit 1 and Exit 2 that create header and trailer pages that contain fewer characters. A sample header page that is created by the optional separator page exit (APSUX01P/APSUC01P or APSUX02P/APSUC02P) is shown in Figure 46 on page 150. This format prints faster than the default separator pages on SNA-attached and TCP/IP-attached printers because it contains less data to be transmitted to the printer.

For better performance on the 4224 and 4234 printers, use the default separator pages (APSUX01/APSUC01 and APSUX02/APSUC02), shown in Figure 45 on page 149, rather than the optional separator pages (APSUX01P/APSUC01P and APSUX02P/APSUC02P).
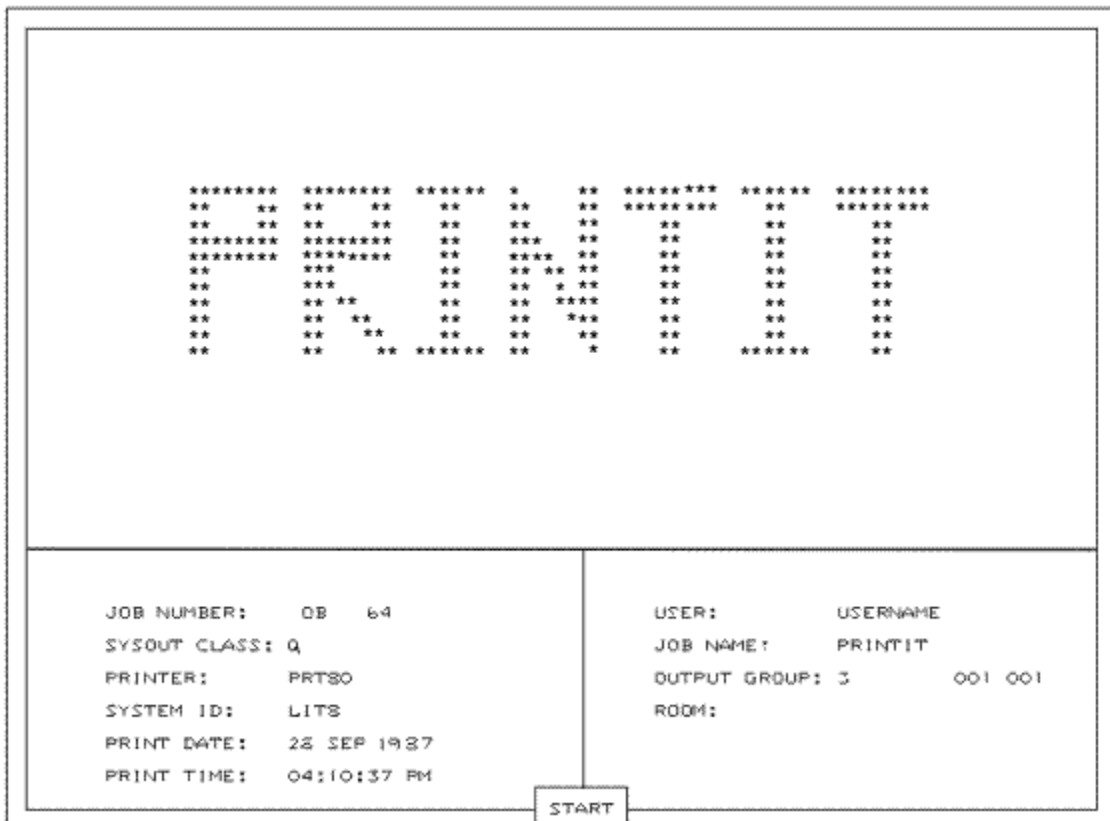
Figure 46. Example of a fast-printing job-header page for PSF (APSUX01P/APSUC01P)

In addition, PSF supplies source code for a version of Exit 1 and Exit 2 to create job separators that use guaranteed print labeling (APSUX01S/APSUC01S and APSUX02S/APSUC02S). For more information about guaranteed print labeling, see *PSF for z/OS: Security Guide*.

You can also create your own separator page exits. For more information, see "Changing exit routines" on page 262 and "Separator-page exits: Exit 1, Exit 2, and Exit 3" on page 276.

Additional distribution information

The separator page that is produced by the IBM-supplied exits might contain additional distribution information. This information is obtained from the OUTPUT JCL keywords for the print data set. The data from the following keywords can appear on the separator page:

- ADDRESS
- BUILDING
- DEPT
- NAME
- ROOM
- TITLE

In addition, if the data set is segmented, the segment ID is placed on the separator page. For more information about these keywords and how to specify them, see *PSF for z/OS: User's Guide* and the *z/OS MVS JCL Reference*.

The C exits always have more distribution information on the separator pages. However, the IBM-supplied assembler exits must be assembled with SYSPARM set to any value except null for additional distribution information to appear on the separator pages. The SYSPARM value causes the assembler exits to use the format that places the additional distribution information on the separator page. If SYSPARM is omitted, the assembler exits are assembled to produce the old format.

The default assembler exits (APSUX01 and APSUX02) are assembled in PSF with SYSPARM specified. The default assembler exits place the additional distribution information on the separator page. If the default assembler Exit 1 and Exit 2 are used without modification, no assembly is required to obtain the additional distribution information.

This example shows how to specify SYSPARM:

```
//ASM    EXEC asm_pgm,PARM=(SYSPARM(YES))
```

Figure 47 on page 151 shows a sample header page that contains the additional distribution information that is produced by the default APSUX01/APSUC01 exit that PSF supplies.

Figure 47. Sample header page with distribution information from the default APSUX01/APSUC01 exit

Figure 48 on page 152 shows a sample header page that contains the additional distribution information that is produced by the optional fast printing APSUX01P/APSUC01P exit.
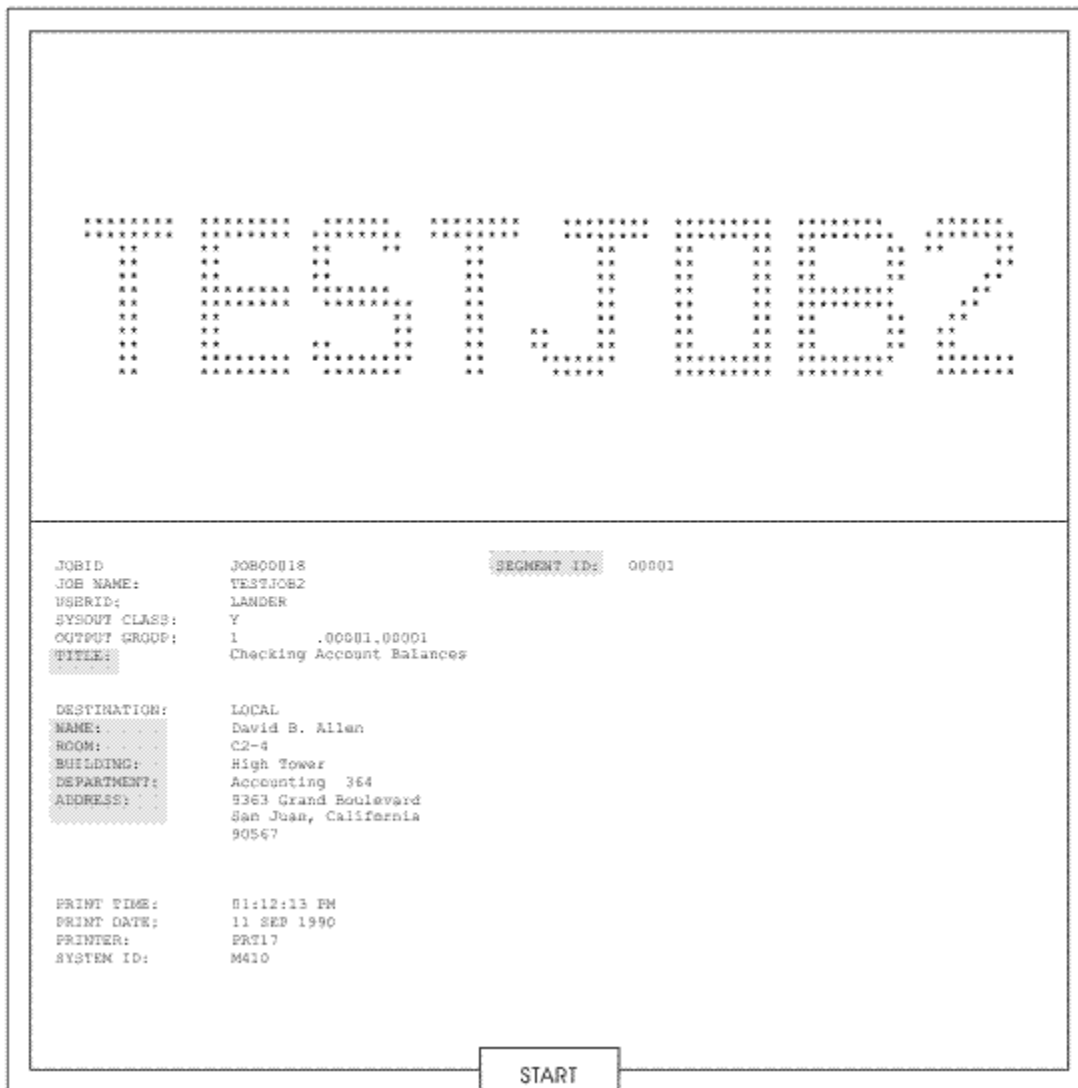
Figure 48. Sample header page with distribution information from the optional APSUX01P/APSUC01P exit

Separator page resources

The form definition, page definition, and font that are used for printing the separator pages are specified on OUTPUT statements in the PSF startup procedure. These output statements are referenced by the JOBHDR and JOBTRLR parameters of the PRINTDEV statement. Examples of separator page OUTPUT statements are shown in "Sample PSF startup procedures" on page 135.

In the separator page OUTPUT statement, specify a fixed-space font that prints at 12 or more characters per inch, and specify a page definition that prints in the print direction you want. Use a form definition suitable for your printer. For example, for a cut-sheet printer, you can specify a form definition that selects job-header pages from the alternative paper bin (in which you keep colored paper).

PSF does not use inline resources to process separator pages. PSF obtains all resources used for separator pages from the system library or the security library.

The page origin for the job-header or job-trailer page is the same as the default media origin. For information about the media origins a printer supports, see the documentation provided with the printer. For information about the coordinate system that is used to define page and media origins, see *Intelligent Printer Data Stream Reference*.

Printing copies of a separator page

To print more than one copy of a header or trailer separator page, use one of the following methods:

- If the header or trailer uses the default form definition, specify the number of copies in the COPIES parameter on the OUTPUT statement for the header or trailer page. For example, to print three copies of your trailer page, specify:

```
//JOBTRLR  OUTPUT PAGEDEF=V06483,CHARS=60D8,COPIES=(,3)
```

- If the header or trailer does not use the default form definition, specify the number of copies in the form definition used by the header or trailer. Specify the name of the form definition in the FORMDEF parameter on the OUTPUT statement for the header or trailer page. For example, to use form definition F1MYSEP for a trailer page, specify:

```
//JOBTRLR  OUTPUT PAGEDEF=V06483,CHARS=60D8,FORMDEF=MYSEP
```

Notes:

1. Include the OUTPUT statement for the header or trailer in the PSF startup procedure. Name the OUTPUT statement in the JOBHDR or JOBTRLR parameter of the PRINTDEV statement in the PSF startup procedure. See "Sample PSF startup procedures" on page 135 for sample PSF startup procedures.

2. Do not specify both the COPIES and FORMDEF parameters on the OUTPUT statement. If you do, PSF ignores the COPIES parameter and uses the number of copies specified in the form definition.

3. Do not use the COPIES=(*n*) form of the COPIES parameter. If you do, PSF ignores the parameter because this form of the COPIES parameter does not apply to separator pages. Always use the COPIES=(,*n*) form as shown in the example.

For additional information about the COPIES parameter, see *PSF for z/OS: User's Guide*.

If forms marking is specified, the 3800 printer automatically prints multiple copies of trailer pages. See "Forms marking" on page 153 .

Distributed print function (DPF) considerations

PSF identifies to DPF the data sets to be printed as a group. DPF keeps a data set group together as one print file on the spool.

PSF and DPF on a JES3 system

When you are using DPF with JES3, make sure that all data sets grouped into a single DPF print file have the same forms requirements.

To ensure that JES3 schedules data sets with similar forms requirements together, use the SETUP=FORMS parameter on the PRINTDEV statement. However, when you use this parameter, a change of forms typically causes an intervention message to be sent to the JES3 console. In this environment, the intervention message is not necessary or desirable. Because DPF is a spooling device, no forms need to be changed when the message is displayed. By using the JES3 initialization deck, you can suppress the intervention message for an individual printer. You can issue a JES3 operator command to change the setting in the deck.

You can produce output that consists of multiple print data sets with different print classes between job separators. If your installation is set up this way, all of the data sets between the job separators is grouped into one print file for DPF, but only the class of the first data set in the group is displayed by DPF.

Forms marking

Forms marking produces marks on the edges of the trailer pages, as Figures Figure 49 on page 154 and Figure 50 on page 154 show. If separator pages are turned off, the marks appear on the edges of a blank page that PSF inserted at the end of the job. If a job header page exists but no job trailer page, the marks appear on the edges of the header page.
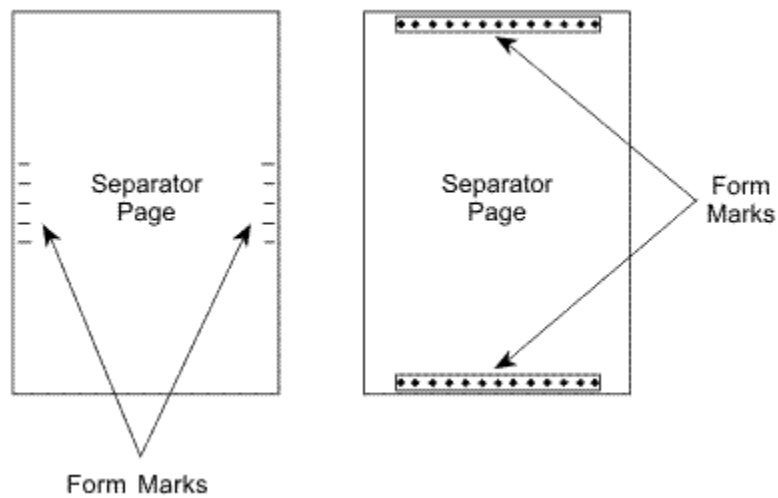
Figure 49. Example of forms marking for cut-sheet paper
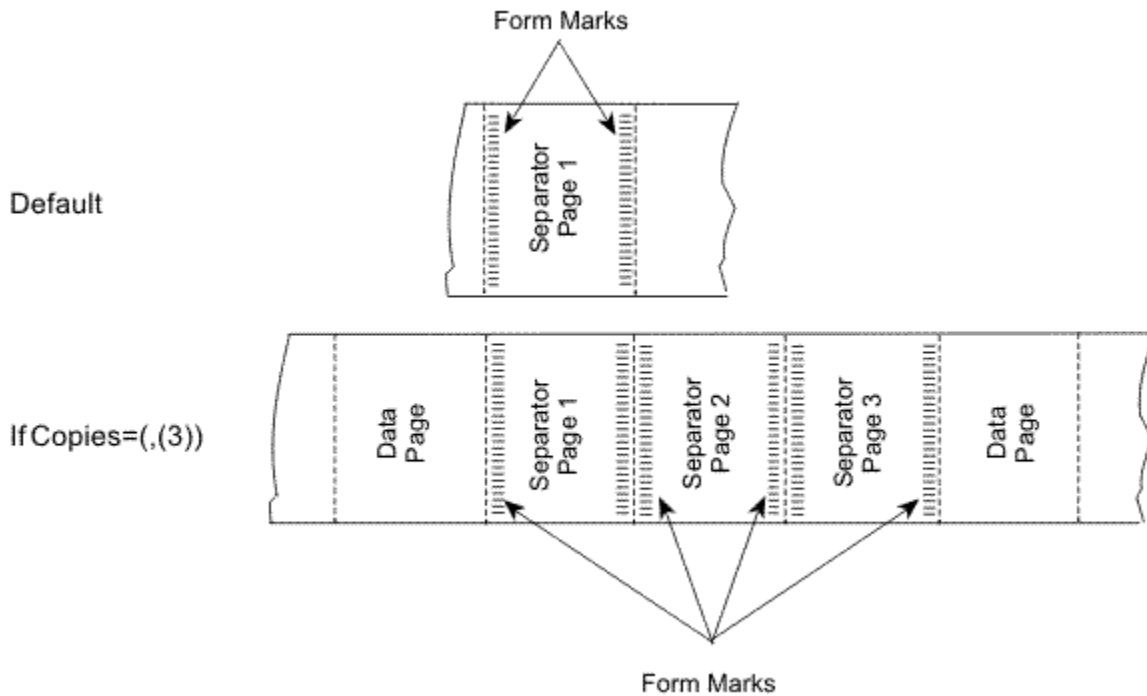
Figure 50. Forms marking for continuous forms

Forms marking is controlled by the following JES initialization options:

- For JES2, select the MARK option on the PRT*nnnn* statement.
- For JES3, select BURST=(YES,M) on the DEVICE statement.

For more information, see "Sample initialization statements for JES2" on page 124 and "Sample initialization statements for JES3" on page 129.

For a 3800 printer, forms marking extends across the perforation as shown in Figure 51 on page 155.
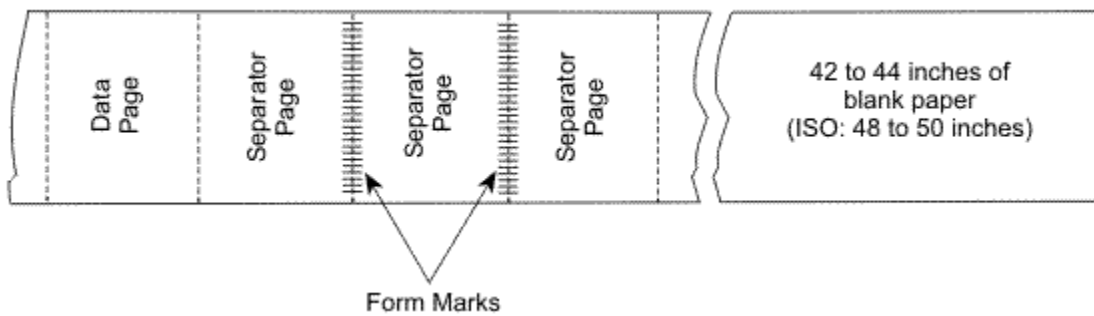
Figure 51. Example of forms marking

If the form length is 7 inches, 8.5 inches, or 11 inches, three copies of the trailer page are printed. For shorter forms, five separator pages are printed and marked. Because the marks extend across the perforation, a length of blank paper is issued after the marked trailer pages to prevent the markings from stopping under the fuser roll and transferring to other pages.

Forms marking of multiple-page job trailers

You can control the forms marking of trailer pages by using installation exits. You can mark particular pages of a multiple-page job trailer, or you can suppress marking of trailer pages altogether.

The ECAMFTRL flag in the Exit Communication Area (ECA) controls the forms marking of job trailer pages. Forms marking is done only if you set this field to 0 and specify forms marking in the JES initialization options. The default for the ECAMFTRL flag is 0. To suppress forms marking, set the ECAMFTRL flag to 1.

The ECAMFTRL flag can be set in any PSF exit except Exit 5. After it is set, the flag remains set until reset by an exit. However, PSF uses the flag only when trailer pages are being processed by Exit 2 (APSUX02/APSUC02).

When you are processing multiple-page job trailers in Exit 2, and you want only the last job trailer page to be marked, set the ECAMFTRL flag to 1 for all but the last trailer page. When ECAMFTRL is set to 1, PSF does not send the MARK FORM order to the printer during the processing of the job trailer. Exit 2 must set ECAMFTRL to 1 before it puts the first record of the first page of the multiple-page job trailer, and set ECAMFTRL to 0 before it puts the first record of the last page of the multiple-page job trailer.

Forms marking of the interrupt message page

You can control forms marking and offset stacking of the interrupt message page by using the Printer Inventory or Exit 7 to mark or offset-stack the interrupt message pages that PSF inserts into the printed output. For more information, see .

Carrier-strip marking

The 3800 printer also supports carrier-strip marking, diagrammed in .
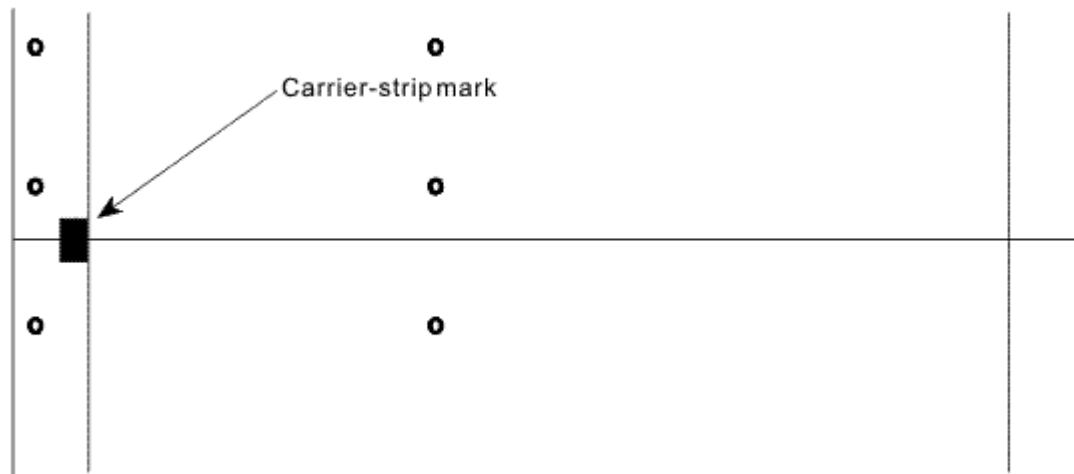
Figure 52. Example of carrier-strip marking on the 3800 printer

You can control carrier-strip marking by specifying JES initialization options or by using the PSF installation exits.

When carrier-strip marking is controlled by JES initialization options, the carrier-strip mark is printed on the job trailer of each job.

- For JES2, carrier-strip marking occurs on the 3800 printer if forms marking is turned off by use of the MARK=NO option of the PRT*nnnn* statement.
- For JES3, carrier-strip marking is specified in the DEVICE statement with STACKER=(YES,C).

For printers that support the mark form carrier strip (MFCS) command, you can use PSF installation exits to control carrier-strip marking. You can set field ECAMFCS to X'F5' to initiate marking, and to X'00' to disable marking. After ECAMFCS is set to X'F5', carrier-strip marking is d until it is reset to X'00', or until the end of the job. ECAMFCS can be set in any exit except Exit 5. For separator page exits, set the field value before the first record is put. For basic information about exits, see Chapter 25, "Using installation exits," on page 257.

Note: On the 3800 printer, if you request both carrier-strip marking (MFCS) and forms marking on the trailer page, three copies of the trailer page are printed if the form length is 7 inches, 8.5 inches, or 11 inches, and the carrier-strip marking is printed. The forms marking across the perforation is not printed. For shorter forms, the only difference is that five separator pages are printed.

On the 3800 printer, if carrier-strip marking is requested, and the page with the carrier-strip mark stops in the transfer station because no other pages are to be printed (or for any other reason), the carrier-strip mark is not printed. For more information about carrier-strip marking, see *Reference Manual for the IBM 3800 Printing Subsystem Models 3 and 6*.

Copy marking and offset stacking

You can use copy marking and offset stacking to separate data sets within jobs or to separate jobs. Copy marking is used for continuous-forms printers. Offset stacking is used for cut-sheet printers or for continuous forms sent to a burster-trimmer-stacker or equivalent post-processing device.

For continuous-form printers that support offset stacking, PSF issues the Alternate Offset Stacker (AOS) command when the printer's offsetter is set as installed/enabled and the printer's BTS/CTS is either set as installed/enabled or set as not installed/not enabled. For special considerations, see the BURST and COPYMARK statements in "Parameters for the JES2 initialization statement" on page 125 and see the COPYMARK and STACKER statements in "Parameters for the JES3 initialization statement" on page 130.

You can control offset stacking of the interrupt message page by using the Printer Inventory or Exit 7 to offset-stack the interrupt message pages that PSF inserts into the printed output. For more information, see Chapter 26, "Recovering from print errors," on page 325.

Copy marking is illustrated in Figure 53 on page 157.
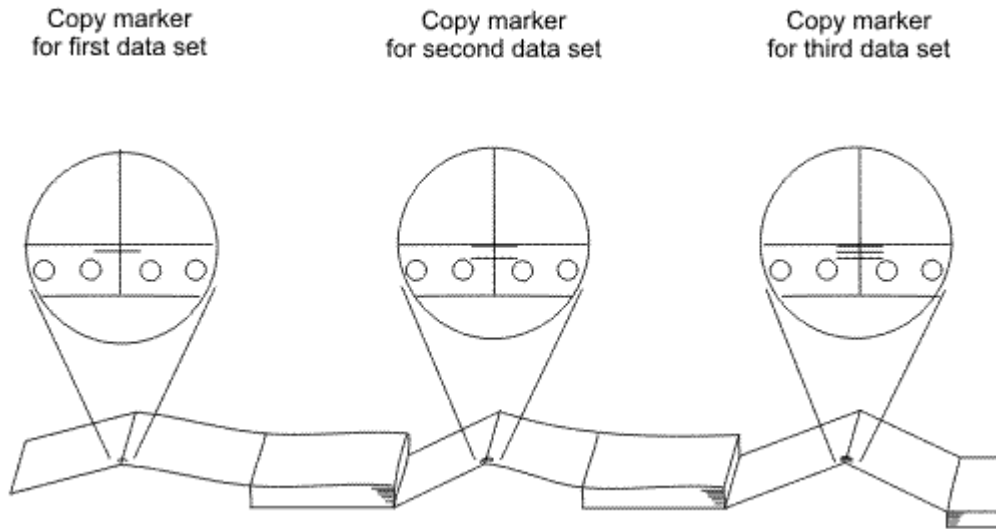
Figure 53. Example of copy marking to simulate offset stacking

The printer marks the carrier strip of each page of the data set with 1, 2, or 3 black marks, incrementing the count at the end of each data set or at the end of each job. If the same job is sent to a cut-sheet printer, instead of printing copy marks, the printer offsets each data set or job in the output stacker.

Copy marking and offset stacking are controlled by JES initialization parameters in the following list. To suppress copy marking, see "Suppressing copy marks" on page 157.

- For JES2, the COPYMARK parameter of the PRT*nnnn* statement specifies how the copy mark is to be used: incremented at the end of each data set, copy of a data set, or job; or no copy mark is to be used.

  For JES2, the COPYMARK parameter of the PRT*nnnn* statement specifies when the offset stacking is to occur: at the end of each data set, copy of a data set, or job, or not at all.

  For the syntax of the COPYMARK parameter, see "Parameters for the JES2 initialization statement" on page 125.

- For JES3, the COPYMARK parameter of the DEVICE statement specifies when the copy mark increment is to occur: at the end of each data set or copy of a data set, or at the end of each job.

  For JES3, the COPYMARK parameter of the DEVICE statement specifies when the offset stacking is to occur: at the end of each data set or copy of a data set, or at the end of each job, or not at all.

  For the syntax of the COPYMARK parameter, see "Parameters for the JES3 initialization statement" on page 130.

The form definition resource can also specify separation by copy group. Then, each time a new copy group is activated, either the pages are offset or the copy mark is incremented. For more information about the form definition, see *PSF for z/OS: User's Guide*.

Suppressing copy marks

If you do not want to use copy marks, you can suppress them in one of these ways:

- Specify COPYMARK=NONE on the JES2 initialization statement (see COPYMARK).
- Specify YES on the **Suppress copy marks** parameter in the Printer Inventory (see Suppress copy marks).
- Set XTP7SUCM in Exit 7 (see "XTP7SUCM" on page 436).

Separating copy marking and offset stacking

The JES COPYMARK parameter controls copy marking and offset stacking (jogging); however, you can override the JES COPYMARK setting for offset stacking with the **Offset stacking** parameter in the Printer Inventory (see Offset stacking) or XTP7JOG in Exit 7 (see "XTP7JOG" on page 435). If either of these fields is set, the JES COPYMARK parameter controls only copy marking and the Printer Inventory or Exit 7 controls offset stacking.

You can set offset stacking in the Printer Inventory or Exit 7 to NONE (no offset stacking is done), DATA SET (offset stacking is done on a data set boundary), or JOB (offset stacking is done on a job boundary).

If XTP7JOG is set to an incorrect value in Exit 7, PSF defaults to the value set by the JES COPYMARK parameter. You can set the XTP7JOG option in the Printer Inventory or during the initialization (INIT) call to Exit 7 (XTP7INIT), or during the begin data set call (BDSC) to Exit 7 (XTP7BDS). In the BDSC, XTP7JOG can be set ON only for the first data set of a job. If it is set ON for any other data set in the job, PSF ignores the setting. For more information, see "XTP7JOG" on page 435.

Notes:

1. If the **Suppress copy marks** parameter in the Printer Inventory (see Suppress copy marks) or the XTP7SUCM field in Exit 7 (see "XTP7SUCM" on page 436) is set, copy marks are suppressed, regardless of the JES COPYMARK setting.

2. If an Invoke Medium Map (IMM) structured field that references a copy group that is requesting offset stacking is present in the data stream, this offset stacking command overrides the **Offset stacking** parameter in the Printer Inventory and the XTP7JOG field in Exit 7. It also increments the copy mark count, unless copy marks are suppressed by setting the JES COPYMARK parameter to NONE, or by setting the **Suppress copy marks** parameter in the Printer Inventory or the XTP7SUCM field in Exit 7 to ON.

Separating data sets or copies of data sets

Data sets within a print job or multiple copies of the same data set within a print job can be separated by:

- Data set header pages
- Copy-marking or offset stacking

PSF creates and prints a data set header page before each data set and before each copy of a data set within a print job. Whether data set header pages are to be printed is specified by the following JES initialization statements:

- For JES2, the SEPDS parameter of the PRT*nnnn* statement.
- For JES3, the HEADER parameter of the DEVICE statement. This parameter controls both data set headers and job headers.

For more information, see "Parameters for the JES2 initialization statement" on page 125 and "Parameters for the JES3 initialization statement" on page 130.

The content of the data set header page is controlled by PSF installation Exit 3. PSF supplies a default version of this exit. You can also code your own version of the exit. For more information, see "Changing exit routines" on page 262 and "Separator-page exits: Exit 1, Exit 2, and Exit 3" on page 276.

The form definition, page definition, and font that are used for printing the data set header pages are specified on an OUTPUT statement in the PSF startup procedure, and are referenced in the DSHDR parameter of the PRINTDEV statement. For information about the OUTPUT statements that are used with separator pages, see "Job separator pages" on page 148.

Copy marking or offset stacking is also used to separate data sets and copies of data sets within a job. For a description of these options, see "Copy marking and offset stacking" on page 156.

Disabled mechanisms

Some printers can function even if a mechanism, such as a paper source, becomes disabled. When the mechanism first becomes disabled and a job is sent to the printer, a PSF abend occurs, as it does for any permanent printer error. However, the operator can choose to restart PSF for a disabled printer, and the mechanism is bypassed in one of these ways:

- If the duplex paper path is disabled, the printer prints in simplex.
- If a paper source is disabled, paper is selected from an alternative source.
- If the offset stacker is disabled, no offset stacking occurs between jobs or copy groups.

You might want to define separate print output classes to assist the operator in determining whether to restart PSF for a disabled printer. One class might be defined for print data sets that are tolerant of disabled mechanisms. This class would be defined for print-job submitters who, for example, accept simplex instead of duplex output. Another class might be defined for print data sets that are not tolerant of changes to the output caused by disabled mechanisms. If separate classes are defined in this way and a mechanism on a printer becomes disabled, the PSF operator can safely start a tolerant class of data sets on that printer and use the printer until the mechanism is enabled.

Note: When a duplex paper path or an offset stacker mechanism is disabled, the PSF FSA must be stopped and restarted to pick up a change in the mechanisms status. If the mechanism is disabled when the FSA is started, and is then enabled by the operator, the FSA must be restarted to pick up the change in mechanism status.

For information about the disabled mechanisms a printer supports, see the documentation provided with the printer.

System Management Facilities type 6 records

In deferred-printing mode, PSF creates a System Management Facilities (SMF) type 6 record for each data set that is processed. The information in the type 6 record depends on the device type.

- All printers
  - System ID (processor ID)
  - Logical output-device name
  - User ID
  - Job name
  - Job number
  - Step name
  - Procedure step name (procstepname)
  - Data set name (dsname)
  - Data definition name (DDNAME)
  - Process mode (PRMODE)
  - Time and date when PSF receives a data set from JES and when the SMF type 6 record is written
  - Data-input error indicator
  - Data set control indicators (data set ended, interrupted, restarted by JES operator)
  - Logical input-record count
  - Logical page count (accumulative number of logical pages per side)
  - Copy groups
  - SYSOUT class
  - Physical-page (forms) count (approximate number of physical sheets, including separator and message pages; the same whether you are printing simplex or duplex)

Note: The physical-page count (SMF6PGE) is only the same as the printer page counter if the print file pages are simplex and the printer is in simplex mode.

  – Impression (side) count (number of sides of sheets printed, including separator and message pages; blank Side 2 only counted if printing duplex)

    Note: The impression count (SMF6IMPS) is typically the same as the printer page counter.

  – Form name
  – CHARS fonts and UCS fonts
  – Number of fonts (each time a different rotation of an outline font is activated, the number of fonts is incremented by one in the SMF record; however, with raster fonts, a coded font is counted only once, no matter how many rotations of the font are activated)
  – Number of overlays and page segments loaded
  – Number of page definitions and form definitions loaded in virtual storage
  – Form definition names
  – Page definition names
  – Segment ID
  – Input bin information (maximum of six input bins), including sheet counts and paper length and width values for each bin used.
  – Scheduler work block text units (SWBTU)

    Note: The SWBTU contains the JCL values that are merged from the DD statement, the OUTPUT statement, and the JES defaults, according to the hierarchy in which the resources are arranged. You can use the SWBTUREQ macro to obtain the JCL parameters specified for the print data set from the SWBTU. The PSF-supplied separator exits (APSUX01/APSUC01, APSUX01P/APSUC01P, APSUX01S/APSUC01S, APSUX02/APSUC02, APSUX02P/APSUC02P, and APSUX02S/APSUC02S) contain an example of how to start the SWBTUREQ macro. For more information about the SWBTUREQ macro, see *z/OS MVS Programming: Authorized Assembler Services Reference SET-WTO*. For more information about SWBs and how to use the SWB token to retrieve keywords, see *z/OS MVS Using the Functional Subsystem Interface*.

• Printers using continuous forms

  – Paper-footage count
  – Forms-flash name (3800 printer only)
  – Flash count (3800 printer only)

• Printers using cut-sheet paper

  – Cut-sheet indicator
  – Bin-selection indicator

    Note: If the bin selection is changed while a page with multiple copies is being printed, the SMF6PGE count and the SMF6BNCN bin count might be incorrect by the number of copies of pages printed before the bin selection was changed.

  – Page count per bin
  – Duplex indicators

• Printers using the PSF print labeling function

  – For more information, see *PSF for z/OS: Security Guide*.

PSF supplies an installation exit point that you can use to provide an exit routine (Exit 5 module) to modify SMF type 6 records. If you did not provide this routine, PSF creates a standard type 6 record and writes it to the SMF data set.

For information about the Exit 5 module that is used to modify SMF type 6 records, see . For information about the mapping of fields in SMF type 6 records, see .

The SMF type 6 accounting records that are written at the host do not reflect any actions taken by the DPF operator, such as changing the copy count, deleting a print file without printing it, or reprinting data because of errors.

Structure of SMF type 6 records

PSF creates these sections in the SMF type 6 record:

- Base section
- First extension
- Common section
- Second extension
- Bin sections
- Enhanced SYSOUT section
- File transfer section

"Map of SMF type 6 records" on page 161 shows the map of SMF type 6 records. Values in parentheses are in hexadecimal.

Map of SMF type 6 records

```
==============================================================================
    RECORD TYPE 6 - SYSOUT WRITER
    IN JES2, WRITTEN FOR EACH JOB OUTPUT ELEMENT, WHICH
    REPRESENTS A GROUP OF DS DIFFERENTIATED BY PUNCH OR
    PRINTER SETUP & TYPE OF OUTPUT(EG HELD VS NON-HELD).
    IN JES3, WRITTEN FOR EACH COPY OF A DATA SET
==============================================================================
Offsets    Type      Length  Name          Description

  0   (0) STRUCTURE    94    SMFRCD06
  0   (0) SIGNED        2    SMF6LEN       RDW RECORD LENGTH
  2   (2) SIGNED        2    SMF6SEG       RDW SEGMENT DESCRIPTOR
  4   (4) BITSTRING     1    SMF6FLG       OPERATING ENVIRONMENT
  5   (5) UNSIGNED      1    SMF6RTY       RECORD TYPE 6
  6   (6) SIGNED        4    SMF6TME       TOD RECORD WRITTEN
 10   (A) CHARACTER     4    SMF6DTE       DATE RECORD WRITTEN
 14   (E) CHARACTER     4    SMF6SID       SYSTEM ID FROM INSTALLATION
 18  (12) CHARACTER     8    SMF6JBN       JOB NAME FROM JMR
 26  (1A) SIGNED        4    SMF6RST       READER START TIME (.01 SEC)
 30  (1E) CHARACTER     4    SMF6RSD       READER START DATE
 34  (22) CHARACTER     8    SMF6UIF       USER IDENTIFICATION FIELD
 42  (2A) CHARACTER     1    SMF6OWC       OUTPUT WRITER CLASS - BLANK FOR
                                           NON SYSOUT
 43  (2B) SIGNED        4    SMF6WST       TOD WRITER START
 47  (2F) CHARACTER     4    SMF6WSD       WRITER START DATE
 51  (33) SIGNED        4    SMF6NLR       # LOGICAL RECS WRITTEN
 55  (37) BITSTRING     1    SMF6IOE       IO ERROR INDICATOR
          1111  1...         *             RESERVED
          ....  .1..         SMF6DIE       DATA INPUT ERROR
          ....  ..1.         *             RESERVED
          ....  ...1         SMF6CBIE      CONTROL BLOCK INPUT
                                           ERROR@ZA05171
 56  (38) UNSIGNED      1    SMF6NDS       # DS PROCESSED BY WRITER
 57  (39) CHARACTER     4    SMF6FMN       FORM NUMBER
```

Figure 54. Map of SMF type 6 records (Part 1 of 6)

```
 61  (3D) BITSTRING    1  SMF6PAD1     INDICATORS - THE SECTIONS WILL
                                       BE IN THE ORDER AS LISTED
                                       BELOW WHEN THE BIT IS TURNED ON
        1...  ....         SMF6FEXT    FIRST EXTENSION PRESENT
        .1..  ....         SMF6REXT    COMMON SECTION PRESENT
        ..1.  ....         SMF6SEXT    SECOND EXTENSION PRESENT
        ...1  ....         SMF6ESS1    ENHANCED SYSOUT SEC PRESENT
        ....  1...         SMF6FTFR    FILE TRANSFER SEC PRESENT
        ....  .111         *           RESERVED
 62  (3E) SIGNED       2  SMF6SBS      SUBSYSTEM GENERATING ID
                                       EXTWTR=0, JES2-2, JES3-5,
                                       PSF=7, IP PrintWay = 9
 64  (40) SIGNED       2  SMF6LN1      LGTH OF SECTION INCLUDING SELF
 66  (42) BITSTRING    1  SMF6DCI      DATA SET CONTROL INDICATORS
        1...  ....         SMF6DCRV    RESERVED
        .1..  ....         SMF6SDS     SPUN OFF DATA SET
        ..1.  ....         SMF6OCN     TERMINATED BY OPERATOR
        ...1  ....         SMF6ORD     INTERRUPTED BY OPERATOR (JES2)
                                       OPERATOR RESTARTED DATA SET
                                       WITH DESTINATION(JES3)
        ....  1...         SMF6OR      RESTARTED BY OPERATOR
        ....  .1..         SMF6ROR     CONTINUATION OF INTERRUPTED
                                       GROUP(JES2) RECEIVED OP
                                       RESTARTED DS(JES3)
        ....  ..1.         SMF6OSS     CARRIAGE OVERRIDEN BY OPERATOR
                                       (JES2) OPERATOR STARTED WITH
                                       SINGLE SPACE(JES3)
        ....  ...1         SMF6INT     PUNCH WAS INTERRUPTED
 67  (43) BITSTRING    1  SMF6INDC     BIT INDICATORS. BITS 0-3 ARE
                                       RESERVED FOR FUTURE EXPANSION
                                       OF DATA SET CONTROL
                                       INDICATORS. BITS 4-7 ARE
                                       RECORD LEVEL INDICATORS TO BE
                                       SET IN BIT VALUE FORMAT.
                                       EXAMPLE: LEVEL 1=X'01', LEVEL
                                       12=X'0C'. THIS NUMBER WILL BE
                                       INCREMENTED BY 1 EACH TIME A
                                       NEW RELEASE CHANGES THE RECORD.
 68  (44) CHARACTER    4  SMF6JNM      WHEN SMF6INDC CONTAINS A X'1',
                                       THIS FIELD CONTAINS A
                                       FOUR-DIGIT EBCDIC JOB NUMBER
                                       WHEN SMF6INDC CONTAINS A X'3'
                                       OR GREATER, AND THE JOB NUMBER
                                       HAS MORE THAN 4 DIGITS, THIS
                                       FIELD CONTAINS ZEROS. IF THE
                                       JOB NUMBER IS < OR = TO 9999,
                                       THIS FIELD CONTAINS THE JOB
                                       NUMBER. FOR AN APPC
                                       TRANSACTION, THIS FIELD
                                       CONTAINS ZEROES. THE CORRECT
                                       JOB NUMBER OR APPC TRANSACTION
                                       ID IS FOUND IN SMF6JBID
 72  (48) CHARACTER    8  SMF6OUT      LOGICAL OUTPUT DEVICE NAME
                                       (FOR THE 3820, ACF/VTAM
                                       LOGICAL UNIT NAME)
 80  (50) CHARACTER    4  SMF6FCB      FCB ID
 84  (54) CHARACTER    4  SMF6UCS      UCS ID, END OF REC FOR EXT WTR
 88  (58) SIGNED       4  SMF6PGE      APPROXIMATE PHYSICAL PAGE
                                       COUNT
```

Figure 55. Map of SMF type 6 records (Part 2 of 6)

```
 92  (5C) CHARACTER    2  SMF6J2S      SPECIAL JES2 SECTION
 92  (5C) SIGNED       2  SMF6RTE      OUTPUT ROUTE CODE OR ZERO
 94  (5E) CHARACTER       SMF6END2     END OF JES2 BASE RECORD

 92  (5C) STRUCTURE   24  *
 92  (5C) CHARACTER   24  SMF6J3S      SPECIAL JES3 SECTION
 92  (5C) BITSTRING    2  SMF6DFE      DATA FORMAT ERROR INDICATORS
      1111  11..        *             0-5 RESERVED
      ....  ..1.       SMF6CCE       6-ONE OR MORE RECORDS
                                     CONTAINED INVALID 1ST CHAR
                                     CONTROL DATA, DEFAULT
                                     SUBSTITUTED
      ....  ...1       SMF6RBE       7-INVALID LGTH REC IN DATA,
                                     TRUNCATED OR PADDED
 93  (5D) BITSTRING    1  *           8-15 RESERVED
 94  (5E) UNSIGNED     2  SMF6OPR      OUTPUT PRIORITY
 96  (60) CHARACTER    8  SMF6GRP      LOG OUTPUT DEV GROUP NAME
104  (68) CHARACTER    8  SMF6RSVJ     RESERVED FOR JES3
112  (70) CHARACTER    4  SMF6RSVU     RESERVED FOR USER
116  (74) CHARACTER       SMF6END      END OF JES3 BASE RECORD

=============================================================================
     BEGINNING OF FIRST EXTENSION - THIS SECTION WILL ONLY BE
     PRESENT WHEN SMF6SBS IS SET TO 2, 5 OR 7 INDICATING THAT JES2
     JES3 OR PSF HAS GENERATED THIS RECORD.
=============================================================================
  0  (0) STRUCTURE   36  SMF6EXT1     FIRST EXTENSION - NON-IMPACT
                                     PRINTING SUBSYSTEM SECT
  0  (0) SIGNED       2  SMF6LN2      LENGTH OF FIRST EXTENSION
                                     INCLUDING THIS FIELD
  2  (2) CHARACTER    1  SMF6CPS(8)   COPIES DISTRIBUTION
 10  (A) CHARACTER   16  SMF6CHR      TRANSLATE TABLE NAMES
                                     FROM CHARS PARAMETER
 26  (1A) CHARACTER   4  SMF6MID      COPY MODIFICATION MODULE NAME
 30  (1E) CHARACTER   4  SMF6FLI      FLASH OVERLAY NAME
 34  (22) BITSTRING   1  SMF6FLC      NUMBER OF COPIES FLASHED
 35  (23) BITSTRING   1  SMF6BID      FLAG BYTE
      1...  ....       SMF6BTS       THE BTSS WAS USED FOR OUTPUT
      .1..  ....       SMF6OPJ       OPTCD=J WAS USED FOR OUTPUT
      ..1.  ....       SMF6CSP       CUT SHEET PRINTER
=============================================================================
     BEGINNING OF COMMON SECTION - THIS SECTION IS AN EXTENSION OF
     THE FIXED HEADER SECTION AND WILL BE WRITTEN BY ALL GENERATOR
     OF THE TYPE 6 RECORD. THIS WAS PREVIOUSLY CALLED THE ROUTING
     SECTION.
=============================================================================
  0  (0) STRUCTURE  162  SMF6RSEC     COMMON SECTION
  0  (0) SIGNED      2   SMF6LN3      LENGTH OF THIS SECTION
                                     INCLUDING THIS FIELD
  2  (2) CHARACTER   4   SMF6ROUT     OUTPUT ROUTE CODE
  6  (6) CHARACTER   8   SMF6EFMN     OUTPUT FORM NUMBER
 14  (E) CHARACTER  16   *            RESERVED
 30  (1E) CHARACTER  8   SMF6JBID     JOB ID
 38  (26) CHARACTER  8   SMF6STNM     STEP NAME
 46  (2E) CHARACTER  8   SMF6PRNM     PROCEDURE STEP NAME
 54  (36) CHARACTER  8   SMF6DDNM     DD NAME
 62  (3E) CHARACTER  8   SMF6USID     USER ID
 70  (46) CHARACTER  8   SMF6SECS     SECURITY LABEL (SECLABEL)
 78  (4E) CHARACTER  8   SMF6PRMD     PROCESSING MODE
 86  (56) CHARACTER  53  SMF6DSNM     DATA SET RESOURCE NAME
139  (8B) CHARACTER  3   *            RESERVED
142  (8E) CHARACTER  20  SMF6OTOK     OUTPUT GROUP TOKEN
=============================================================================
     BEGINNING OF SECOND EXTENSION - THIS SECTION WILL ONLY BE
     PRESENT WHEN SMF6SBS IS SET TO 7 INDICATING THAT PSF HAS
     GENERATED THIS RECORD.
=============================================================================
  0  (0) STRUCTURE  120  SMF6EXT2     SECOND EXTENSION - APA (ALL
                                     POINTS ADDRESSABLE) PRINTING
                                     SUBSYSTEM SECTION
  0  (0) SIGNED      2   SMF6LN4      LENGTH OF SECOND EXTENSION
                                     INCLUDING THIS FIELD
  2  (2) SIGNED      2   SMF6BNOF     OFFSET TO 'BINS' SECTION
  4  (4) SIGNED      4   SMF6FONT     NUMBER OF FONTS USED
  8  (8) SIGNED      4   SMF6LFNT     NUMBER OF FONTS LOADED
```

Figure 56. Map of SMF type 6 records (Part 3 of 6)

```
 12   (C) SIGNED       4  SMF6OVLY      NUMBER OF OVERLAYS USED
 16  (10) SIGNED       4  SMF6LOLY      NUMBER OF OVERLAYS LOADED
 20  (14) SIGNED       4  SMF6PGSG      NBR OF PAGE SEGMENTS USED
 24  (18) SIGNED       4  SMF6LPSG      NBR OF PAGE SEGMENTS LOADED
 28  (1C) SIGNED       4  SMF6IMPS      COUNT OF LOGICAL IMPRESSIONS
                                         (SIDES) PROCESSED
 32  (20) SIGNED       4  SMF6FEET      NUMBER OF FEET OF DOCUMENT
                                         PRINTED
 36  (24) SIGNED       4  SMF6PGDF      NBR OF PAGEDEFS USED
 40  (28) SIGNED       4  SMF6FMDF      NUMBER OF FORMDEFS USED
 44  (2C) BITSTRING    1  SMF6BIN       FLAG BYTE
          1...  ....      SMF6BIN1      BIN1 WAS USED FOR ANY PART OF
                                         THE DATA SET
          .1..  ....      SMF6BIN2      BIN2 WAS USED FOR ANY PART OF
                                         THE DATA SET
          ..1.  ....      SMF6BIN3      BIN3 WAS USED FOR ANY PART OF
                                         THE DATA SET
          ...1  ....      SMF6BIN4      BIN4 WAS USED FOR ANY PART OF
                                         THE DATA SET
 45  (2D) BITSTRING    1  SMF6PGOP      FLAG BYTE
          1...  ....      SMF6DUPS      STANDARD DUPLEX WAS USED FOR
                                         ANY PART OF THE DATA SET
          .1..  ....      SMF6DUPT      TUMBLE DUPLEX WAS USED FOR ANY
                                         PART OF THE DATA SET
          ..1.  ....      SMF6SYSA      KEYWORD SYSAREA=Y
          ...1  ....      SMF6DPGL      KEYWORD DPAGELBL=Y
          ....  1...      SMF6SUCC      PRINT OPERATION WAS SUCCESS
          ....  .1..      SMF6SPGL      KEYWORD SPAGELBL=Y
          ....  ..1.      SMF6SOER      ERROR OCCURRED PROCESSING
                                         SECURITY OVERLAY
          ....  ...1      SMF6IGER      IMAGE GENERATOR OVERRUN ERROR
                                         OCCURRED
 46  (2E) BITSTRING    1  SMF6FLG3      FLAG BYTE
          1...  ....      SMF6SLIG      SECURITY LABEL INTEGRITY
          .1..  ....      SMF6JHPP      JOB HEADER PAGE WAS PRINTED
          ..1.  ....      SMF6JTPP      JOB TRAILER PAGE WAS PRINTED
          ...1  ....      SMF6DPLS      DATA PAGE LABELING WAS SUPPRESSED
          ....  1...      SMF6UPAS      USER PRINTABLE AREA WAS SUPPRESSED
 47  (2F) UNSIGNED     1  SMF6APAL      LEVEL INDICATOR FOR APA SECTION
 48  (30) SIGNED       4  SMF6NSOL      NUMBER OF SECURITY OVERLAYS USED
 52  (34) SIGNED       4  SMF6NSFO      NUMBER OF SECURITY FONTS USED
 56  (38) SIGNED       4  SMF6NPS       NUMBER OF SECURITY PAGE
                                         SEGMENTS USED
 60  (3C) CHARACTER    8  SMF6FDNM      FORMDEF NAME
 68  (44) CHARACTER    8  SMF6PDNM      PAGEDEF NAME
 76  (4C) CHARACTER    8  SMF6PTDV      PRINTDEV NAME
 84  (54) CHARACTER   32  SMF6OCNM      OBJECT CONTAINER NAME(S)
 84  (54) CHARACTER    8  SMF6SETU      COMSETUP OBJECT CONTAINER NAME
 92  (5C) CHARACTER    8  *             RESERVED OBJECT CONTAINER NAME
100  (64) CHARACTER    8  *             RESERVED OBJECT CONTAINER NAME
108  (6C) CHARACTER    8  *             RESERVED OBJECT CONTAINER NAME
116  (74) SIGNED       4  SMF6LPGE      COUNT OF LOGICAL PAGES
120  (78) SIGNED       4  SMF6DSHT      COUNT OF DUPLEX SHEETS PROCESSED
124  (7C) SIGNED       4  SMF6SSHT      COUNT OF SIMPLEX SHEETS PROCESSED
128  (80) SIGNED       4  SMF6NUP1      COUNT OF N_UP 1 SHEETS PROCESSED
132  (84) SIGNED       4  SMF6NUP2      COUNT OF N_UP 2 SHEETS PROCESSED
136  (88) SIGNED       4  SMF6NUP3      COUNT OF N_UP 3 SHEETS PROCESSED
140  (8C) SIGNED       4  SMF6NUP4      COUNT OF N_UP 4 SHEETS PROCESSED
```

Figure 57. Map of SMF type 6 records (Part 4 of 6)

```
        ==============================================================================
          MULTI-BINS HEADER SECTION (OFFSET DEFINED BY SMF6BNOF)
        ==============================================================================
          0   (0) STRUCTURE    4  SMF6BNMI     MULTI BIN INPUT SECTION
          0   (0) SIGNED       2  SMF6BNLN     LENGTH OF SECTION INCLUDING
                                               BIN COUNTERS
          2   (2) SIGNED       2  SMF6BNUM     NUMBER OF COUNTER ENTRIES
          4   (4) CHARACTER       SMF6BNCS     START OF BIN COUNTERS
        ==============================================================================
          MULTI-BINS COUNTER SECTION
            - FOLLOWS "MULTI-BIN" HEADER SECTION
        ==============================================================================
          0   (0) STRUCTURE    8  SMF6BNCN(*)  BIN COUNTER
          0   (0) UNSIGNED     1  SMF6BNNO     BIN NUMBER
          1   (1) UNSIGNED     3  SMF6BNCT     BIN COUNTER
          4   (4) UNSIGNED     2  SMF6BNLE     PAPER LNGTH IN MILLIMETERS
          6   (6) UNSIGNED     2  SMF6BNWI     PAPER WIDTH IN MILLIMETERS


        ==============================================================================
          ENHANCED SYSOUT SECTION
        ==============================================================================
          0   (0) STRUCTURE    *  SMF6ESS      ENHANCED SYSOUT SECTION
          0   (0) SIGNED       2  SMF6LN5      LENGTH OF ESS SECTION
                                               INCLUDING THIS FIELD
          2   (2) SIGNED       4  SMF6SGID     SEGMENT IDENTIFIER
          6   (6) BITSTRING    1  SMF6IND      SECTION INDICATOR
                  1... ....       SMF6SJF      ERROR OBTAINING SWBTU - SWBTU
                                               DATA AREA NOT PRESENT
          7   (7) CHARACTER    1  SMF6RSV      RESERVED
          8   (8) CHARACTER    8  SMF6JDVT     JDVTNAME
         16  (10) SIGNED       2  SMF6TUL      SWBTU DATA AREA LENGTH
         18  (12) CHARACTER    *  SMF6TU       SWBTU DATA AREA - DATA AREA CAN BE
                                               PROCESSED USING SWBTUREQ MACRO
        ==============================================================================
          FILE TRANSFER SECTION
        ==============================================================================
          0   (0) STRUCTURE    *  SMF6FXFR     FILE TRANSFER SECTION
          0   (0) SIGNED       2  SMF6LN6      LENGTH OF FILE TRANSFER
                                               SECTION INCLUDING THIS FIELD
          2   (2) UNSIGNED     4  SMF6BYTE     TOTAL NUMBER OF BYTES SENT
          6   (6) UNSIGNED     4  SMF6IPV4     FORMATTED IPV4 TARGET ADDRESS
          6   (6) UNSIGNED     1   SMF6IP1     1ST SEGMENT OF FORMATTED IPV4 TARGET
                                               ADDRESS
          7   (7) UNSIGNED     1   SMF6IP2     2ND SEGMENT OF FORMATTED IPV4 TARGET
                                               ADDRESS
          8   (8) UNSIGNED     1   SMF6IP3     3RD SEGMENT OF FORMATTED IPV4 TARGET
                                               ADDRESS
          9   (9) UNSIGNED     1   SMF6IP4     4TH SEGMENT OF FORMATTED IPV4 TARGET
                                               ADDRESS
         10   (A) BITSTRING    1  SMF6FTL      FILE TRANSFER SECTION LEVEL INDICATOR
                  .... ...1       SMF6FTL1     z/OS V1R5
                  .... ..1.       SMF6FTL2     ADDED IPV6 SUPPORT
                  .... ..11       SMF6FTL3     ADDED TCPIP PORT SUPPORT
         11   (B) CHARACTER    1  SMF6FTL      RESERVED
         12   (C) UNSIGNED     2  SMF6PORT     TCPIP PORT NUMBER
         14   (E) CHARACTER    8               RESERVED
         22  (16) UNSIGNED     2  SMF6PQLN     LENGTH OF PRINT QUEUE NAME
         24  (18) CHARACTER   24  SMF6PRTQ     PRINT QUEUE NAME
         48  (30) UNSIGNED     8  SMF6BYTD     TOTAL BYTE COUNT THAT SUPPORTS >4 GB
         48  (30) UNSIGNED     4   SMF6HWRD    HIGH WORD OF TOTAL BYTE COUNT
         52  (34) UNSIGNED     4   SMF6LWRD    LOW WORD OF TOTAL BYTE COUNT
         56  (38) CHARACTER   16  SMF6IPV6     FORMATTED IPV6 TARGET ADDRESS
         72  (48) UNSIGNED     2  SMF6URIL     LENGTH OF HOST URI
         74  (4A) CHARACTER  255  SMF6URI      TARGET DEVICE URI
        329 (149) CHARACTER    3               RESERVED
        332 (14C) UNSIGNED     2  SMF6ACTL     LENGTH OF SMF6ACCT
        334 (14E) CHARACTER  143  SMF6ACCT     JOB ACCOUNT RECORD FROM JCL
        477 (1DD) CHARACTER    3               RESERVED
```

Figure 58. Map of SMF type 6 records (Part 5 of 6)

```
      =========================================================================
            CONSTANTS
      =========================================================================
      Length    Type      Value    Name          Description

          1    DECIMAL    1        SMF6LEV2      THIS VARIABLE IS FOR JES2 TO
                                                 SET LEVEL INDICATOR
          1    DECIMAL    3        SMF6J2L3      THIS VARIABLE IS FOR JES2 TO
                                                 SET LEVEL INDICATOR
          1    DECIMAL    4        SMF6J2L4      THIS VARIABLE IS FOR JES2 TO
                                                 SET LEVEL INDICATOR FOR
                                                 SECURITY SUPPORT

          1    DECIMAL    1        SMF6LEV3      THIS VARIABLE IS FOR JES3 TO
                                                 SET LEVEL INDICATOR
          1    DECIMAL    3        SMF6J3L3      THIS VARIABLE IS FOR JES3 TO
                                                 SET LEVEL INDICATOR
          1    DECIMAL    4        SMF6J3L4      THIS VARIABLE IS FOR JES3 TO
                                                 SET LEVEL INDICATOR FOR
                                                 SECURITY SUPPORT
          1    DECIMAL    5        SMF6LEV4      MVS/JES2 RELEASE 4.1.0
          1    DECIMAL    6        SMF6LEV6      PSF for OS/390 RELEASE 3.1.0
```

Figure 59. Map of SMF type 6 records (Part 6 of 6)

Chapter 10. Using direct-printing mode

This information provides information about how to attach an AFP printer in direct-printing mode, describes what must be done to define PSF as a z/OS subsystem, and provides examples of the JCL that must be coded when your application program is run.

PSF supports printers as system output devices for deferred printing under Job Entry Subsystem (JES), and for direct printing under Direct Printer Services Subsystem (DPSS). DPSS is the access means by which the application program sends records directly to an attached printer or directly to PSF, bypassing the JES spool. In direct-printing mode, an application program allocates a host-connected, channel-attached (non-SNA) printer for the program's exclusive use. When you use direct-printing mode, make sure that the printer is not allocated to JES, but is online to z/OS.

Notes:

1. See Chapter 3, "Using channel-attached printers with PSF," on page 9 if your printer is not yet defined to z/OS.

2. SNA-attached and TCP/IP-attached printers cannot use direct-printing mode.

3. Microfilm devices are not supported in direct-printing mode.

4. Direct-printing mode does not provide the following functions, which are provided only in deferred-printing mode:

   - User libraries
   - System-assisted restart
   - PSF repositioning

     Repositioning includes PAPER JAM recovery, PRINT ADJUST commands entered at the printer, and recovery for other intervention-required conditions at the printer.

   - Data-set checkpointing
   - Multiple data set processing
   - JES operator commands to control the printer
   - Job header and trailer pages
   - Data set header pages
   - Carrier-strip marking
   - Mandatory page labeling [3]
   - System Management Facilities (SMF) type-6 processing
   - Notification of job completion
   - Restartable abends
   - PSF installation Exits 1 - 7
   - Message data set redirection
   - Parameter specification in the Printer Inventory
   - AFPPARMS control statements

5. Because direct-printing mode does not support repositioning or the use of Print Adjust, check your printer documentation for the appropriate settings that are required to prevent the printer from reporting these types of errors to DPSS. Generally, settings such as DIRECT PRINT = YES|NO and JAM RECOVERY = ON|OFF must be set appropriately for direct printing to function correctly.

6. If PSF is attached to a printer that does not support the selective deletion of page segments and overlays, DPSS stops processing data sets if jobs require more than 127 page segments or more than

---

[3] For more information about page labeling, see *PSF for z/OS: Security Guide*.

127 overlays. If you have IBM printer microcode that supports extended overlays and page segments, the maximum number of overlays is extended to 32,511.

7. Information about the installation exit for direct-printing mode is given in "Direct-print exit: Exit 16" on page 318.

8. For forms marking, see the Exit 16 begin-data-set call (BDSC), described in "Direct-print exit: Exit 16" on page 318.

How to use direct-printing mode

The procedure for using PSF in direct-printing mode is:

1. Define DPSS to z/OS by using the procedures described in "Defining the DPSS" on page 168.

2. Code JCL to enable direct-printing mode by using "Application program JCL for direct-printing mode" on page 169 as a sample.

3. Allocate the printer by either:
   - Coding a DD statement in the JCL that is used to enable direct-printing mode
   - Coding a dynamic allocation invocation in the application program

4. In the application program, code an OPEN macro instruction for a data control block (DCB) that contains the DDNAME to be used to specify the printer in the JCL.

5. If you need more than one copy of a data set, follow the procedures in "Requesting multiple copies" on page 171.

6. Code a CLOSE macro instruction for the DCB of the DDNAME of direct-printing mode. PSF then detaches the FSA for the printer that is using direct-printing mode.

7. If the directly attached device is to be managed by PSF, DPSS Open establishes DPSS/PSF as the access method that the application program is to use. When the application closes the print data set, DPSS detaches the PSF subtask and returns to Data Management Close. Other job tasks that intervene by using exclusive ENQ/DEQ might result in system deadlock. You can prevent system deadlock by coding application level ENQ/DEQ around all system macros that might require ENQ on TIOT. Some examples of system macros that might require ENQ/DEQ are Open, Close, Dynalloc, and EOV. To avoid unexpected results or abend conditions, review the *Authorized Assembler Services Reference* for your operating system before you code the macro in your application.

Defining the DPSS

DPSS maintains communication between the application program and PSF. DPSS, in turn, uses the functional subsystem interface (FSI) to communicate with PSF and to deliver data to the printer.

To use direct-printing mode, you must first define DPSS as a z/OS subsystem by modifying the PARMLIB member IEFSSN00 or by defining a new IEFSSN*xx* member. IEFSSN00 (or the newly created member) must contain the following statement:

```
SUBSYS SUBNAME(DPSS) INITRTN(APSADPSS) INITPARM(n)
```

where

- DPSS is the subsystem name.
- APSADPSS is the program name.
- *n* is the maximum number of FSAs (printers) to be used per address space.

PSF supports a maximum of 128 FSAs. The actual number of FSAs depends on several factors, such as the number and size of resources used in the print jobs and the amount of virtual storage available.

Figure 60 on page 169 shows a job stream that can create a new IEFSSN*xx* member of SYS1.PARMLIB. Note that if you create a new IEFSSN*xx* member, you must update the IEASYS*xx* member to point to IEFSSN*xx*, and an IPL is required.

```
//IEFSSN JOB ...,MSGLEVEL=1,MSGCLASS=A,CLASS=A
//STEP1 EXEC PGM=IEBUPDTE,PARM=NEW,REGION=256K
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SYS1.PARMLIB,DISP=SHR
//SYSUT2 DD DSN=SYS1.PARMLIB,DISP=SHR
//SYSIN  DD  *
./ ADD NAME=IEFSSNBB,LIST=ALL
SUBSYS SUBNAME(DPSS) INITRTN(APSADPSS) INITPARM(1)
./ ENDUP
/*
```

Figure 60. Job stream for creating an IEFSSNBB member of SYS1.PARMLIB

Consider these:

- You can dynamically add the subsystem after the system IPL by using the following operator command:

  ```
  SETSSI ADD,SUBNAME=DPSS,INITRTN=APSADPSS,INITPARM=n
  ```

- PSF DPSS does not support any other dynamic subsystem functions.

Application program JCL for direct-printing mode

Figure 61 on page 169 shows sample JCL statements that enable direct-printing mode. JCL in this format can be incorporated into the application program job stream to start direct-printing mode. Be sure that you have allocated the space that is required in the system library for the libraries you refer to in your JCL.

```
//DIRECT   JOB...
⋮
//STEP01   EXEC PGM=program,REGION=1750K
⋮
//MSGDS    OUTPUT PAGEDEF=pdefname,FORMDEF=fdefname
//FONT01   DD    DSN=SYS1.FONTLIB,DISP=SHR
//TTFONT01 DD PATH='/usr/lpp/PSF/fonts/ttf/'
//         DD PATH='/usr/lpp/PSF/fonts/ttfa_g/'
//PSEG01   DD    DSN=SYS1.PSEGLIB,DISP=SHR
//OLAY01   DD    DSN=SYS1.OVERLIB,DISP=SHR
//PDEF01   DD    DSN=INST.PDEFLIB,DISP=SHR
//         DD    DSN=SYS1.PDEFLIB,DISP=SHR
//FDEF01   DD    DSN=INST.FDEFLIB,DISP=SHR
//         DD    DSN=SYS1.FDEFLIB,DISP=SHR
//DA1      CNTL
//DA1      PRINTDEV TRACE=YES,
//            FONTDD=*.FONT01,
//            FONTPATH=*.TTFONT01,
//            OVLYDD=*.OLAY01,
//            PSEGDD=*.PSEG01,
//            PDEFDD=*.PDEF01,
//            FDEFDD=*.FDEF01,
//            MESSAGE=*.MSGDS,
//            SETUP=FORMS,
//            BUFNO=5,
//            PAGEDEF=pdefname,
//            FORMDEF=fdefname,
//            CHARS=(fontname1,
//            fontname2,
//            fontname3,
//            fontname4),
//            PIMSG=(YES,16),
//            DATACK=BLOCK
//DA1      ENDCNTL
//OUT1     OUTPUT FORMDEF=fdefname
//name     DD    UNIT=180,CNTL=*.DA1,OUTPUT=*.OUT1,...
⋮
```

Figure 61. Sample JCL to enable direct printing

These are descriptions of the statements and parameters shown in Figure 61 on page 169:

EXEC
Specifies the name of the direct print application program and establishes the region size and the PSF PARM parameters used for trace specifications. For more information about the PSF PARM parameters, see "PARM parameters on the EXEC statement" on page 144. For more information about the REGION parameter, see "Selecting region sizes" on page 194.

MSGDS OUTPUT
Specifies the page definition and the form definition used to format and print messages.

DD
Identifies the system libraries that contain resources (fonts, page segments, overlays, page definitions, and form definitions) used for printing. For more information about libraries, see Chapter 6, "Working with libraries," on page 49.

CNTL
Precedes the PRINTDEV statement; ENDCNTL must follow the PRINTDEV statement. The CNTL and ENDCNTL statements have no parameters. The label on the CNTL statement must match the label on the corresponding ENDCNTL and PRINTDEV statements.

PRINTDEV
Specifies initialization parameters for a printer operating in direct-printing mode. Within the JCL for a single direct-printing job, you can code CNTL, PRINTDEV, and ENDCNTL statements for more than one printer. This parameter is required.

For a description of all the PRINTDEV parameters, see "Specifying PRINTDEV parameters" on page 102.

ENDCNTL
Specifies the end of a PRINTDEV statement.

OUTPUT
This optional statement specifies AFP parameters for PSF. See *PSF for z/OS: User's Guide* for more information about the AFP parameters that can be specified in the JCL.

***name* DD**
Defines the printer to be used and refers to the CNTL statement associated with this DD statement. The *name* specifies the DDNAME used by the application program that was specified in the EXEC statement (EXEC PGM=*program*). The UNIT parameter specifies the device address for the printer in direct-printing mode. The CNTL and ENDCNTL sequence that includes the PRINTDEV statement must be within the current step that contains the DD statement. This DD statement can also be allocated by using dynamic allocation instead of JCL. See the *z/OS MVS Programming: Authorized Assembler Services Guide* for information about how to use dynamic allocation. With dynamic allocation, the direct print application program can allocate the printer while the program is running, and thus can control how and when the printer is allocated. For instance, the application program can determine such things as:

- Which JCL CNTL statement to reference for the allocation
- Which printer to allocate
- The particular point in the application's processing at which the allocation is to occur

Dynamic allocation provides greater flexibility than JCL DD allocation provides, as Table 20 on page 170 shows.

Table 20. JCL DD allocation versus dynamic allocation

JCL DD Allocation	Dynamic Allocation
Must be defined before the job is submitted.	Can be defined while the application program is running.
Can be used only by authorized programs.	Can be used by both authorized and unauthorized programs.

Table 20. JCL DD allocation versus dynamic allocation (continued)	
JCL DD Allocation	**Dynamic Allocation**
Must be done during step allocation, just before the system passes control to the direct print allocation program.	Can be done at any point while the application program is running.

When you are using dynamic allocation, use the DALCNTL text unit key to refer to a CNTL statement in the JCL for the step.

Requesting multiple copies

Under JES, PSF retransmits a data set to provide more than one copy. In direct-printing mode, however, your application must retransmit the data set for each group value if either of the following applies:

- More than one copy is specified with the COPIES or COPYCNT parameter. For example, if you specify COPIES=14 in direct-printing mode, the number of copies defaults to 1. To get 14 copies, your application must retransmit the data set 14 times to PSF.

- More than one group value is specified in a COPIES JCL parameter. For example, if you specify COPIES=(,(1,3,2)) in a direct-printing mode, the total number of requested copies (6) is ignored. The first group value (1) determines the number of copies. If the first group value is 3, three copies of each page are printed consecutively.

For information about using the COPIES and COPYCNT parameters under JES, see *PSF for z/OS: User's Guide*.

3800 considerations

For the 3800 printer, you can use the SETPRT macro instruction to initially set or dynamically change the printer control information. For DPSS, the following parameters can be changed by use of the SETPRT macro:

- BURST
- CHARS
- COPIES
- FCB
- FLASH
- UCS

For additional information about how to use the SETPRT macro, see *z/OS DFSMS Macro Instructions for Data Sets*.

Chapter 11. Managing resources

Resources are collections of data and control information that PSF can use in printing a data set. PSF supports these resources:

- Fonts
- Form definitions
- Object containers
- Overlays
- Page definitions
- Page segments

Form definitions and page definitions are loaded into PSF storage; all other resources are downloaded to the printer. PSF determines when to load or download a resource.

After PSF processes a data set, it deletes all the page segments and overlays that were downloaded to printer storage, except for overlays specified in the Printer Inventory or PRINTDEV and any fonts, page segments, or both used by that overlay. The resource exit can be used to change this processing. Only after this deletion is done does PSF start to process another data set. For more information, see "Resource-deletion call (RDC)" on page 311. The most recently used fonts generally remain in the printer.

Resources can also be deleted from printer storage to allow space for a complex page and to maintain printer throughput. Complex pages typically contain many page segments, various fonts, several data object resources, or overlays.

Default resources can be defined when PSF is started. For instructions, see "PSF startup procedures for AFP printers" on page 135, and Chapter 10, "Using direct-printing mode," on page 167. Default resources are loaded into printer storage as needed and are deleted as necessary, either when space is needed for a complex page or when a device error occurs. If printer resources are deleted, PSF retransmits those resources to the printer as required.

Maintaining resource integrity

To maintain the integrity of resources, consider these:

- Page definitions, form definitions, overlays, page segments, fonts, and object containers can be placed inline in the print data set to ensure that the correct resources are used on the target system.
- You can update resource data sets, but you must make sure that PSF still has access to those data sets.
- You can use other AFP licensed programs (for example, Overlay Generation Language/370 and Page Printer Formatting Aid/370) to update resources in PSF libraries. The DSI and NODSI values in the SCHEDxx member of SYS1.PARMLIB are used to specify when resources can be updated. See "Default PPT entry" on page 6 for more information about specifying DSI or NODSI.
- If a resource in a library is modified after it is loaded into virtual storage or into printer storage,[4] PSF might not delete the unmodified resource from storage.

  To make sure that the modified resource is used, do one of these:

  – Put the resource in a user library.
  – Drain and then restart the FSA or the printer.
  – Change the name of the modified resource and use the new name.
  – Code the resource inline.

---

[4] Page definitions and form definitions are loaded into virtual storage; fonts, page segments, and overlays are loaded into printer storage.

– Use the resource exit.
- PSF deletes resources that were loaded from user libraries and inline resources when it finishes processing the data set for which they were loaded. PSF also tells the printer not to capture resources found inline or in user libraries. If you are connected through DPF or WPM, the Printer Inventory or Exit 7 can specify that inline resources are captured on DPF or WPM. See Capture inline resources for more information.
- Exclude PSF library data sets from products that compress or defragment the computer storage device while PSF is active.
- Exclude PSF library data sets from HSM migration.

How PSF selects resources

PSF resources can be specified in:

- Job Entry Subsystem (JES) initialization statements
- Printer Inventory
- PRINTDEV statement of PSF startup procedures
- User job control language (JCL) for the print job
- Print data set
- Form definitions and page definitions
- PSF resource Exit 7
- Printer defaults (fonts only)

PSF selects these processing parameter values in the following order:

1. In a form definition named on a OUTPUT JCL statement, including any inline form definitions named on the OUTPUT statement.
2. In the user JCL.
3. In JES initialization statements.
4. In the Printer Inventory or the PRINTDEV statement.
5. In a form definition named in the PRINTDEV statement, including any inline form definitions named on the PRINTDEV statement.

Selecting a forms flash for a 3800 printer

A forms flash is a way to print an overlay with a negative plate projected on a form. PSF selects a forms flash for the IBM 3800 printer, based on the control statement hierarchy.

Hierarchy of flash selection in deferred-printing mode

1. In the user JCL:

    a. FLASH specified by FORMDEF= on OUTPUT statement (see Notes 1 and 2)

    b. FLASH= specified on DD statement (see Note 3)

    c. FLASH= specified on OUTPUT statement (see Note 3)

2. JES default in the current printer setup

3. FLASH specified by **Form definition** in Printer Inventory or FORMDEF= in PRINTDEV statement (see Notes 2 and 3)

Hierarchy of flash selection in direct-printing mode

In the user JCL:

1. FLASH specified by FORMDEF= on OUTPUT statement (see Notes 1 and 2)

2. FLASH= specified on DD statement (see Note 3)

3. FLASH= specified on OUTPUT statement (see Note 3)

4. FLASH specified by FORMDEF= on PRINTDEV statement (see Notes 2 and 3)

Notes:

1. If an OUTPUT statement in the user JCL names a form definition, PSF uses the forms flash specified in that form definition. If no forms flash is specified in the form definition, the user JCL can specify the forms flash on the FLASH parameter.

2. If the user JCL does not include a FLASH parameter, the operator is told (in a SETUP message) to load the default JES forms flash, if one exists. If no default JES forms flash exists (and therefore no SETUP message is issued), PSF uses any forms flash that is currently loaded in the printer.

3. If the forms flash is specified only in the form definition from the Printer Inventory or the PRINTDEV statement, the user must specify a forms flash with the FLASH parameter in the USER JCL. Otherwise, PSF does not activate the forms flash; nor is the forms flash that is currently loaded in the printer used.

Selecting a copy group

A copy group (also called a medium map) is an object within a form definition or a print data set. A copy group within a print data set is called an internal copy group (or an internal medium map). Copy groups control such items as data suppression, forms flash, overlays, media source, modifications to a form, and the number of copies of each page.

The way PSF selects a copy group is based on specific criteria. PSF uses an internal copy group only if it is immediately followed by an Invoke Medium Map (IMM) structured field that references the name of that copy group. An IMM structured field that does not immediately follow an internal copy group cannot reference an internal copy group elsewhere in the document; therefore, PSF assumes that the IMM references a copy group in a form definition. If an IMM is not present, and for any pages in the data set that are encountered before an IMM, PSF uses the first or only copy group in the form definition that is used to print that data set.

PSF selects a copy group based on whether an IMM is present:

- If an IMM is present, PSF selects a copy group in this order:

  1. PSF uses an internal copy group if it is immediately followed by the IMM structured field that references it.

  2. PSF uses the first copy group the IMM references in the form definition that is used to print the data set.

- If an IMM is not present, PSF uses the first copy group in the form definition that is used to print the data set.

See "Form definitions in deferred-printing mode" on page 176 and "Form definitions in direct-printing mode" on page 176 for the hierarchies that PSF follows to select the form definition that is used to print the data set.

Selecting form definitions, page definitions, and fonts

Hierarchy lists in "Form definitions in deferred-printing mode" on page 176 through "TrueType and OpenType fonts in deferred-printing mode" on page 179 show how the form definition, page definition, and fonts for a print job are selected. PSF takes the user's JCL parameters from the DD statement and the first OUTPUT statement found; the search proceeds in the following order:

1. Explicitly referred to

2. Implicitly referred to at the step level

3. Implicitly referred to at the job level

If the same parameter appears both on the DD statement and on the OUTPUT statement, PSF uses the parameter from the DD statement.

An *explicitly referred to* OUTPUT statement is one named by the OUTPUT parameter of a DD statement. An *implicitly referred to* OUTPUT statement must contain a DEFAULT=YES parameter and must not be overridden by any OUTPUT keyword on a DD statement[5]. In the following example, OUT1 is an OUTPUT statement that is explicitly referred to:

```
//JOB01    JOB     MSGCLASS=A
//STEP01   EXEC    PGM=ENDMONTH
//OUT1     OUTPUT  CHARS=60D8
//DD1      DD      SYSOUT=A,OUTPUT=*.OUT1
```

In the following example, OUTA is an OUTPUT statement that is implicitly referred to:

```
//JOB02    JOB     MSGCLASS=A
//OUTA     OUTPUT  DEFAULT=YES,CHARS=60D8
//STEP01   EXEC    PGM=ENDMONTH
//DD1      DD      SYSOUT=A
```

For more information about the OUTPUT statement, see the *z/OS MVS JCL Reference*.

Form definitions in deferred-printing mode

This is the hierarchy of form definition selection in deferred-printing mode:

1. Internal copy group immediately followed by an IMM structured field in the print data set (see Note 1)
2. FORMDEF= specified on OUTPUT statement of the user JCL:

    a. Explicitly referred to (see Notes 2 and 3)

    b. Implicitly referred to step-level

    c. Implicitly referred to job-level

3. First inline form definition (see Note 4)
4. **Form definition** specified in Printer Inventory or FORMDEF= specified on PRINTDEV statement in the PSF startup procedure

Notes:

1. Internal copy groups immediately followed by an IMM structured field in the print data set are used before copy groups specified by a form definition.

    An IMM structured field that does not immediately follow an internal copy group cannot reference an internal copy group elsewhere in the document; therefore, PSF assumes that the IMM references a copy group in a form definition.

2. This includes inline form definitions, which are specified on the OUTPUT statement either by name or with the name DUMMY. If multiple inline fpr, definitions are specified on the OUTPUT statement with the name DUMMY, PSF uses the last inline form definition in the print data set.

3. This includes a form definition in a user library, which must be specified by name on the OUTPUT statement.

4. If the user JCL does not specify the form definition parameter and inline form definitions exist, the first inline form definition in the print data set is selected.

Form definitions in direct-printing mode

This is the hierarchy of form definition selection in direct-printing mode:

1. Internal copy group immediately preceding the IMM record in the print data set (see Note 1)
2. FORMDEF= specified on explicitly referred to OUTPUT statement of the user JCL (see Note 2)
3. First inline form definition (see Note 3)
4. FORMDEF= specified on PRINTDEV statement of the user JCL

---

[5] You cannot use an implicitly referred to OUTPUT statement on a printer in direct-printing mode.

Notes:

1. An IMM structured field that does not immediately follow an internal copy group cannot reference an internal copy group elsewhere in the document; therefore, PSF assumes that the IMM references a copy group in a form definition.

2. This includes inline form definitions, which are specified on the OUTPUT statement either by name or with the name DUMMY. If multiple inline form definitions are specified on the OUTPUT statement with the name DUMMY, PSF uses the last inline form definition in the print data set.

3. If the user JCL does not specify the form definition parameter and inline form definitions exist, the first inline form definition in the print data set is selected.

4. PSF does not use user libraries in direct-printing mode.

Page definitions in deferred-printing mode

This is the hierarchy of page definition selection in deferred-printing mode:

1. In the user JCL:

    a. PAGEDEF= specified on OUTPUT statement (see Note 1)

    b. FCB= specified on DD statement

    c. FCB= specified on OUTPUT statement (see Note 1)

2. JES default in the current printer setup (see Notes 2 and 3)

3. First inline page definition (see Notes 4 and 5)

4. **Page definition** specified in Printer Inventory or PAGEDEF= specified on PRINTDEV statement in the PSF startup procedure (see Note 3)

Notes:

1. The OUTPUT statement that is used is the first one found from explicit, step-level implicit, and job-level implicit.

2. Under JES2, the JES default PAGEDEF can be specified by the FCB parameter in the PRT*nnnn* statement or, if this parameter is not present, by the NIFCB parameter in the PRINTDEF statement.

3. Under JES3, a page definition specified in the Printer Inventory or the PRINTDEV statement is only selected if PDEFAULT=FCB and WS=C are specified on the DEVICE statement. Otherwise, if PDEFAULT=NONE or PDEFAULT is not specified, a JES3 default page definition is taken from the CARRIAGE parameter on the DEVICE statement or, if this parameter is not present, from the CARRIAGE parameter on the OUTSERVE statement. If the OUTSERV statement has no CARRIAGE parameter, a default of 6 lines per inch (lpi) is used, which selects page definition P16. To use inline page definitions in JES3, specify the name of the page definition in the PAGEDEF or FCB of the user JCL.

4. For JES2, if the user JCL does not specify the PAGEDEF or FCB parameter and inline page definitions exist, the first inline page definition in the print data set is selected unless JES sets a default.

5. If an inline page definition is included in the print data set, and either its name matches the name that is selected or the name from the JCL is DUMMY, the inline page definition is used. If multiple inline page definitions are specified as PAGEDEF=DUMMY on the OUTPUT statement, PSF uses the last inline page definition in the print data set.

6. A page definition from a user library is used if its name matches the name selected.

Page definitions in direct-printing mode

This is the hierarchy of page definition selection in direct-printing mode:

1. In the user JCL:

    a. PAGEDEF= specified on explicitly referred to OUTPUT statement of the user JCL (see Note 1)

    b. FCB= specified on DD statement (see Note 1)

c. FCB= specified on explicitly referred to OUTPUT statement of the user JCL (see Note 1)

2. First inline page definition (see Note 2)

3. PAGEDEF= specified on PRINTDEV statement of the user JCL

Notes:

1. This includes inline page definitions, which are specified by name or with the name DUMMY, on the OUTPUT or DD statement. If multiple inline page definitions are specified as PAGEDEF=DUMMY on the OUTPUT statement, PSF uses the last inline page definition in the print data set.

2. If the user JCL does not specify the PAGEDEF or FCB parameter and inline page definitions exist, the first inline page definition in the print data set is used.

3. PSF does not use user libraries in direct-printing mode.

FOCA fonts in deferred-printing mode

When PSF is printing line-format data, this is the hierarchy of FOCA fonts selection in deferred-printing mode:

1. Fonts that are specified by the current page definition, including a JES default page definition (see Note 1)

2. In the user JCL:

   a. CHARS= specified on DD statement

   b. CHARS= specified on OUTPUT statement (see Note 2)

   c. UCS= specified on DD statement

   d. UCS= specified on OUTPUT statement (see Note 2)

3. JES default font in the current printer setup (see Notes 3 and 4)

4. **`Character sets`** specified in Printer Inventory or CHARS= specified on PRINTDEV statement in the PSF startup procedure

5. If the Printer Inventory or the PRINTDEV statement of the PSF startup procedure names the active page definition, any fonts that are specified in that page definition are loaded

6. Hardware default

Notes:

1. The current page definition does not include a page definition specified by the Printer Inventory or the PRINTDEV statement of the PSF startup procedure.

2. The OUTPUT statement that is used is the first one found from explicit, step-level implicit, and job-level implicit.

3. Under JES2, you can select fonts with a SET command, the UCS parameter of the PRT*nnnn* statement, or the NIUCS parameter of the JES2 PRINTDEF statement. To prevent JES2 from sending a default font name to PSF, specify UCS=0 in the PRT*nnnn* statement. For more information about selecting fonts, see *z/OS JES2 Initialization and Tuning Reference* or *PSF for z/OS: User's Guide*.

4. Under JES3, you can select fonts with the CH parameter in the *START, *RESTART, and *CALL commands. You can also use the CHARS parameter in the DEVICE, OUTSERV, and SYSOUT statements. If PDEFAULT=CHARS and WS=U are specified in the JES3 DEVICE statement, hardware default fonts or fonts that are specified in the Printer Inventory or the PRINTDEV statement are selected; otherwise, hardware default fonts or fonts that are specified in the Printer Inventory or the PRINTDEV statement are never selected. For more information about selecting fonts, see *z/OS JES3 Initialization and Tuning Reference* or *PSF for z/OS: User's Guide*.

FOCA fonts in direct-printing mode

When PSF is printing line-format data, this is the hierarchy of FOCA fonts selection in direct-printing mode:

1. Fonts that are specified by the current page definition (unless the current page definition is only specified by the PRINTDEV statement)
2. In the user JCL:
   a. CHARS= specified on DD statement
   b. CHARS= specified on explicitly referred to OUTPUT statement
   c. UCS= specified on DD statement
   d. UCS= specified on explicitly referred to OUTPUT statement
   e. CHARS= specified on PRINTDEV statement
3. If the PRINTDEV statement names the active page definition, any fonts that are specified in that page definition are loaded
4. Hardware default

TrueType and OpenType fonts in deferred-printing mode

When PSF is printing line-format data, this is the hierarchy of TrueType and OpenType fonts selection in deferred-printing mode:

1. Fonts that are specified by the current page definition (the current page definition does not include a page definition specified by the Printer Inventory or the PRINTDEV statement of the PSF startup procedure)
2. If the Printer Inventory or the PRINTDEV statement of the PSF startup procedure names the active page definition, any fonts that are specified in that page definition are loaded

TrueType and OpenType fonts in direct-printing mode

When PSF is printing line-format data, this is the hierarchy of TrueType and OpenType fonts selection in direct-printing mode:

1. Fonts that are specified by the current page definition (unless the current page definition is only specified by the PRINTDEV statement)
2. If the PRINTDEV statement names the active page definition, any fonts that are specified in that page definition are loaded

Searching for resources specified by a print job

The locations where resources are stored depend on the resource type. Resource types are stored in these locations:

Inline in the print data set
    All resource types

System and user path libraries

    Extended code pages
    Object containers (also called data object resources)
    TrueType and OpenType fonts

System and user PDS or PDSE libraries

    FOCA fonts
    Form definitions
    Object containers
    Overlays
    Page definitions
    Page segments

See for examples of resource types.

When a resource is referenced by a print data set, PSF only searches the locations where that resource type can be stored. PSF searches for the resource in this order:

1. If a resource can be activated by PSF, resources that are resident in the printer.
2. If PSF finds an Invoke Medium Map (IMM), internal copy groups immediately preceding IMM records in the print data set.
3. Inline resources in the print data set.
4. Resources in user path libraries.
5. Resources in user PDS or PDSE libraries in the order they are specified.
6. Resources in system path libraries.
7. Resources in system PDS or PDSE libraries in the order they are concatenated.

   If multiple system libraries have been defined for different resolutions, see "Format resolution order" on page 217 for more information about how PSF chooses which system library to use.

Notes:

1. PSF truncates the resource name to eight characters when searching PDS or PDSE libraries.
2. Only the security libraries (if PSFMPL is active) and the system libraries are searched for resources to be used on auxiliary data sets. For B1 security information, see *PSF for z/OS: Security Guide*.
3. An overlay called out in the Printer Inventory or the PRINTDEV is always loaded from the system libraries.
4. For soft IOCA resources, which are those not specified in the data stream with the MDR structured field, PSF searches the system PDS or PDSE libraries before searching the system path libraries.

Using form definitions for basic N_UP printing

PSF supports basic N_UP printing, which enables the job submitter to place up to four pages in partitions on a sheet in simplex mode, and up to eight pages on a sheet in duplex mode. The term N_UP comes from the Page Printer Formatting Aid (PPFA) N_UP subcommand, which the job submitter specifies when a form definition is created.

N_UP printing is supported by some, but not all, AFP printers. To see whether your printer supports N_UP printing, see "Displaying printer information" on page 201 or the documentation provided with the printer. To find the microcode level necessary for N_UP printing, check the documentation for your printer, or consult your IBM marketing representative.

When it processes a form definition containing the basic N_UP subcommand, the PSF-supported printer divides a side of a sheet into one to four equal, fixed partitions. In basic N_UP printing (that is, printing in which the form definition does not include the explicit placement of pages in the partitions), the printer places page *n* in the first partition, page *n+1* in the second partition, page *n+2* in the third partition, and page *n+3* in the fourth partition. PSF processes the pages sequentially, in the order in which they occur in the data stream. PSF processes N_UP commands in both simplex and duplex printing mode, and applies both overlays and page segments to the pages.

Keep these considerations in mind when you are using N_UP printing:

- To create form definitions that contain the N_UP subcommand, you must have PPFA.
- Because the job submitter is placing, in effect, more pages of data on a sheet, the submitter must select appropriately small fonts.
- N_UP with guaranteed print labeling is not supported.
- PSF counts multiple pages on a single sheet and reports accurate accounting information.
- Although N_UP does not adversely affect the performance of PSF, the printer uses more buffer space when it accumulates data for multiple pages than when it accumulates data for a single page.
- PSF supports full error recovery and repositioning.

- With N_UP printing, PSF processes data suppression as before: PSF suppresses data for the entire side of a sheet, not for the individual pages in the partitions.
- The user can specify more page eject actions when a medium map is invoked.

Using form definitions for enhanced N_UP printing

You can use N_UP printing to place the pages at any location on either side of the sheet. With enhanced N_UP printing on the 3900-0W1, 3900 Duplex, 3130, and 3935 printers, users can:

- Place a page relative to any partition origin on either side of the sheet, in any orientation, and of any size that fits on the sheet. Users can place multiple pages relative to the same origin, when the total number of pages does not exceed the N_UP limit for that sheet.
- Place overlays relative to any partition origin, or page origin, with or without variable page data from the application program.
- Specify a different rotation for each page.
- Specify one or more different overlays for each page.
- Specify a different offset for each page.
- Place pages of any size that fit the physical sheet. Page size is not limited by the partition size.

Using PSF, the interface attachment feature, and the MICR postprocessor

The MICR postprocessor enables magnetic ink character recognition (MICR) printing if:

- The device supports MICR printing.
- MICR printing is enabled.
- The font is a MICR font.

Because every host MICR font contains a list to indicate to PSF that the font is a MICR font, you do not have to do anything to prepare the system or PSF for MICR printing. However, be aware that if the user requests a MICR font for a device that either disabled MICR support or does not support MICR printing, PSF issues a message and does not print the page.

Consider using a RACF-controlled user library to manage the MICR fonts. For more information, see "User libraries in deferred-printing mode" on page 51.

For information about the MICR printing support for a printer, see "Displaying printer information" on page 201 or the documentation provided with the printer.

Obtaining data about print file resources

An AFP Statistics (AFPSTATS) report is available in PSF to produce summary data about the resources that are used to print a document. The data in the report identifies resources that are used, describes how a resource is specified to PSF, and indicates the resource library (also called the repository) from which the resource is obtained.

The AFPSTATS report summarizes these resource types:

- Character set
- Coded font
- Code page
- Form definition
- Object container
- Overlay
- Page definition

- Page segment
- TrueType and OpenType fonts

The AFPSTATS report provides print file processing details so that you can:

- Determine in which resource libraries PSF found each resource.
- Diagnose some resource selection problems.
- Obtain statistical data about how a print file is printed, such as the total number of pages, the number of times a specific resource was referenced, and the number of significant events.
- Diagnose some print file printing performance problems.

To generate an AFPSTATS report for a print file, you must first define an AFPSTATS repository, which is a partitioned data set where AFPSTATS reports are written. You define the repository with the AFPSTATS DD statement in the PSF startup procedure and then allocate the data set. See "Specifying the AFPSTATS repository" on page 182.

After the AFPSTATS repository is defined and allocated, an AFPSTATS report can be requested for a print file by:

- Specifying the **AFPSTATS** keyword on the OUTPUT JCL statement. See *PSF for z/OS: User's Guide*.
- Using the XTP7ASAP flag in Exit 7. See "XTP7ASAP" on page 438.

The print file owner can use the OUTPUT JCL statement to request an AFPSTATS report for any PSF print file that is owned; however, if XTP7ASAP is changed by the installation exit, it overrides the OUTPUT statement.

When an AFPSTATS report is written to the AFPSTATS repository, you can view it as a softcopy report or format it as a hardcopy report and print it.

Note: When the AFPSTATS report is active, PSF is collecting data and writing it to the AFPSTATS repository. These extra activities during PSF processing might cause degraded performance. Whenever you experience a problem with performance, first ensure that all extra activity, such as PSF traces and reports, are disabled before you confirm the performance problem.

For more information about the AFPSTATS report, including the format of a softcopy or hardcopy AFPSTATS report, see *PSF for z/OS: User's Guide*.

Specifying the AFPSTATS repository

Requested AFPSTATS reports are written to the AFPSTATS repository, which is an existing PDSE data set. For every request it receives to generate an AFPSTATS report, PSF adds a member to the repository.

To specify the AFPSTATS repository where the AFPSTATS reports are written, define a DD statement in the PSF startup procedure. The AFPSTATS repository must be an existing partitioned data set allocated as a PDSE. You must allocate the PDS as PDSE to reduce problems with insufficient directory blocks and storage space and to allow multiple FSAs to open the data set without a system abend. IBM recommends that the AFPSTATS repository contains the attributes specified in Table 21 on page 182.

Table 21. Recommended AFPSTATS repository attributes

Attribute	Value	Type	Description
DCB=DSORG=	PO	Required	PDS data set organization
DSNTYPE=	LIBRARY	Required	Data set defined as PDSE
DCB=RECFM=	VB	Required	Variable blocked records
DCB=LRECL=	512	Required	Maximum bytes in record
DISP=	SHR	Required	Data set can be used by multiple print files

Table 21. Recommended AFPSTATS repository attributes (continued)			
Attribute	**Value**	**Type**	**Description**

SPACE=	(TRK,(30,5,5))	Required	The size of the AFPSTATS repository depends on several report factors:
			
			Frequency
			    How often an AFPSTATS report is requested.
			
			Size
			    The number of resources used by the print file that is requesting an AFPSTATS report.
			
			Retention
			    How long AFPSTATS reports are required to exist in the AFPSTATS repository.
			
			The SPACE parameter is a recommendation; you might need to change it depending on the report factors.

shows the format of the DD statement you use to specify the AFPSTATS repository in the PSF startup procedure.

```
//AFPSTATS    DD    DSN=dsname,DISP=SHR
```

Figure 62. Specifying the AFPSTATS repository

The DD name for the AFPSTATS repository must always be AFPSTATS. Also, use DISP=SHR so that multiple PSF FSAs can simultaneously write AFPSTATS reports to the repository.

The AFPSTATS repository must be shareable by all print file owners so they can:

- Request that an AFPSTATS report is produced.
- View members in the AFPSTATS repository.
- Delete members from the AFPSTATS repository when they are no longer needed.

Chapter 12. Understanding AFP performance

This information presents the concepts of Advanced Function Presentation (AFP) performance. You can use this information in tuning the software and hardware to maximize the printer throughput with minimum effect on other applications that are running on the host system.

For information about printer throughput rates (given in pages per minute, lines per minute, and impressions per minute), see the documentation provided with the printer.

AFP is an integration of high-function printers with printer software. PSF is the software that runs on the host, preparing data and sending it to a printer, managing AFP resources both within the PSF address space and in the printer, and doing error handling and error recovery. To understand AFP performance, consider these:

- Characteristics of the printer and the total number of printers installed.
- Density and complexity of the print job or data stream sent to the printer.
- Data rate at which the attachment hardware can send data to the printer.
- Ability of the host processor to process and send data to the printer.

The printer

Printer performance, or throughput of the printer, is typically stated in impressions per minute (ipm). It depends on several factors, including:

- Mechanical speed
- Internal processing speed
- Attachment bandwidth
- Paper size
- Printer features

Duplexed pages contain two impressions, one on each side of a sheet of paper. N_UP printing places up to eight impressions on a sheet in duplex mode. For example, N_UP 4 duplex places four impressions on each side of a sheet of paper.

Mechanical speed

The mechanical speed of the printer is defined as the speed at which paper moves through the paper path. Mechanical speed, typically measured in impressions per minute (ipm), determines the rated (maximum) speed of the printer. The mechanical speed of a continuous-forms printer depends in part on the size of the paper that is used.

Internal processing speed

A printer receives the data stream from the host processor. The internal processing speed of the control unit in the printer limits the amount of data the printer can process per unit of time. If the processing requirements of the print-job data stream are greater than the capabilities of the control unit in the printer, the printer throughput is slower than its mechanical speed. If this happens on a cut-sheet printer, the printer stops picking input sheets for one or more cycles until it is ready to send another page to its image generator. If this happens on a continuous-forms printer, the printer stops moving the form for a number of cycles until it is ready to send another page to its image generator. In either case, the printer is running below its mechanical speed. See "Data density and complexity" on page 186.

Attachment bandwidth

As with internal processing speed, the usable bandwidth of the attachment between the host system and the printer limits the amount of data the printer can process per unit of time. When the type of attachment or competing uses for an attachment limit data rates that are below what the control unit in the printer requires, the printer throughput is slower than its mechanical speed.

Paper size

Throughput in impressions per minute (ipm) can be faster for smaller paper sizes than for larger paper sizes. For example, the throughput of a 3835 printer is 88 ipm for 12 x 8.5 inch pages, but is 68 ipm for 14.875 x 11 inch pages.

Printer features

In case print jobs have unusual performance requirements, familiarize yourself with printer features that might affect throughput. For example:

Raster storage

If a print job contains large amounts of image data, or uses large or multiple fonts, it can cause a shortage of raster pattern storage that would affect printer throughput.

Enhancements and RPQs

Some printers can be equipped with performance enhancements, custom features, or request for price quotations (RPQs) that improve the throughput for dense or complex print jobs.

Duplex printing

Some printers can print on both sides of the paper. Duplex printing does not necessarily affect the throughput of the printer. However, frequent changes from simplex to duplex or from duplex to simplex can noticeably reduce the throughput of cut-sheet printers.

For more information about RPQs and performance, see the documentation provided with the printer. For information about the IPDS features of your printer, see "Displaying printer information" on page 201.

The print job

Printer performance can be limited by such characteristics of the print job as:

- Data density and complexity
- Characteristics of the print data sets
- Number and type of resources
- Size of the print job
- Use of FOCA outline fonts

These characteristics affect the use of host resources, storage, DASD, device controller use, channel use, and the host processor.

Data density and complexity

The data stream that is sent to the printer consists of:

- Print data—text and images that print on the page
- Control information—the data that defines the placement and appearance of the printed output

Printer throughput is affected by the amount of print data and control information. In general, bytes of print data require less processing than bytes of control information, and bytes of image data can be processed faster than bytes of text data; however, image data contains far more bytes than text data.

PSF generally uses more processor cycles and more DASD activity as the density and complexity of the print job increase. This increase continues until the limit of the control unit in the printer is reached, at which time the processor use becomes constant.

Characteristics of print data sets

Print data sets can contain line data, XML data, data that is already composed into pages (MO:DCA-P data), image data, graphics data, and bar code data. The data can be simple enough to require little processing by PSF or the printer, or complex enough to require more processing.

The characteristics of the print job can affect printer throughput. If the data streams for jobs have different proportions of control bytes to data bytes, a specific printer handles them at different maximum rates. The following descriptions of the characteristics of print data sets indicate how the characteristics of print jobs can affect host-processor usage, attachment limitations, and printer throughput.

Formatted traditional line data
Uses few fonts, and formatting is done with a page definition. An example is a one-up or multiple-up computer listing. With the same number of characters per page, formatted line data requires more processing by PSF than does simple MO:DCA-P data because each line is an independent element that must be positioned.

Unformatted traditional line data
Uses several fonts, and formatting is done with a more complex page definition. An example is a billing application. With the same number of characters on a page, unformatted line data requires more processing by PSF than formatted line data because parts of each line are independent elements that must be positioned.

Record format line data
Uses any number of fonts, and formatting is done with a more complex page definition. An example is a bank statement application. With the same number of characters on a page, record format line data requires more processing by PSF than traditional line data because more functions, such as header and trailer processing, are available.

XML data
Uses any number of fonts and formatting is done with a more complex page definition. With the same number of characters on a page, XML data requires more processing by PSF than record format line data because each character must be looked at to parse the XML data, such as start tags, end tags, and entity references. The functions available to XML are similar to the functions available to record format line data.

Simple MO:DCA-P data
Uses few fonts, no justification, no tabulation, and no formatting. An example is a simple business letter. Simple MO:DCA-P data requires less processor usage than line data, if both have the same number of characters per page. Little processing by the printer is required unless the pages contain large numbers of characters.

Complex MO:DCA-P data
Uses formatting, line and column justification, and multiple fonts. An example is in-house publishing with illustrations and tables that consist of formatted text and require several control bytes per text character. Image data is likely to be included. Because of the many controls in the data stream, complex MO:DCA-P data requires more processing by the printer, and is more likely to affect printer throughput, than simple MO:DCA-P data.

Simple image data
Uses a single raster pattern to define a simple image. An example is an uncompressed scanned photograph or a bank check. Simple images require less processing by the printer and PSF for each raster byte than complex images. Because simple image data consists of 7200 bytes for each square inch of image at 240-pel density or 11250 bytes for each square inch of image at 300-pel density, higher data rates and processor usage might be required. Special images, such as compressed images or images with large amounts of shading and color, require more processing by the printer and can affect printer throughput.

Complex image
> Data consists of several independent raster patterns, or single raster patterns that are compressed. Examples are graphic objects that are converted to images, and compressed scanned images. Defining each raster pattern requires control bytes that must be processed by PSF or by the printer.

Number and type of resources

PSF manages the resources for the print data set. For example, PSF loads fonts, page segments, and overlays during the processing of a print data set, and deletes page segments and overlays when the processing is completed. PSF retains a number of fonts in the printer between print data sets, depending upon the printer model.

Some print data sets require many resources that need resource-management processing. Management of the resources can increase processor usage and decrease printer throughput. For example, if PSF tries to load resources, and the printer runs out of raster pattern storage, PSF deletes all resources that are not required to print the current sheet and tries loading the new resources. If this action does not free enough raster pattern storage, PSF deletes all the resources in the printer and stops processing the data set.

Resource management can reduce printer throughput in two ways: by using time during which no printing can be done, and by increasing the amount of processing required to print a data set. Resource management can also increase printer throughput. For example, if the print job contains large resources that are used on multiple pages, such as color EPS objects or IOCA FS45 images, PSF can manage these resources if they are identified as hard resources with the Map Data Resource (MDR) structured field. PSF sends hard resources to the printer once instead of each time they are included in the print job.

An AFP Statistics (AFPSTATS) report is available in PSF to produce summary data about the resources that are used to print a job. The data in the report identifies resources that are used, describes how a resource is specified to the print server, and indicates the repository from which the resource is obtained. For more information about the AFPSTATS report, see "Obtaining data about print file resources" on page 181 or see *PSF for z/OS: User's Guide*.

Inline resources

The use of inline resources might adversely affect performance when an error makes it necessary for PSF to reposition because the inline resources must be stored in a cache again. This adverse effect might arise more frequently when the print job contains many inline resources, when the resources are large, or when PSF repositions frequently.

To alleviate region area storage constraints, inline resources can be stored above the 2 GB bar (64-bit storage) by specifying `above-the-bar-storage=inline-resources` on the AFPPARMS control statement or `Above the bar storage=Inline resources` in the Printer Inventory. Because some PSF components cannot address 64-bit storage, an inline resource stored above the bar must be temporarily copied to 31-bit storage before PSF can use it. Depending on the number and size of the inline resources, storing them above the bar can adversely affect performance.

Resident fonts

Some printers support resident fonts. Selecting resident fonts can improve some aspects of performance. For example, printer throughput can be increased if fonts do not have to be sent to the printer. For information about the resident fonts a printer supports, see the documentation provided with the printer.

Font pruning of raster fonts

PSF optionally prunes raster fonts to reduce the amount of font data sent to the printer. This means that PSF does not send all the characters in a character set when it downloads a font. Instead, PSF matches the characters in the character set to the code points in the code page, and sends to the printer only the characters that are referenced in the code page. Font pruning can save time that is needed to send fonts to a printer, and can reduce the amount of raster pattern storage used in the printer.

PSF does not prune outline fonts.

Font pruning is especially useful when the number of characters in the character set exceeds the number of code points in the code page. However, font pruning can increase processor use.

Sometimes font pruning does not improve performance—for example, when the font character sets match the code pages that are used.

Note: You can deactivate the pruning of double-byte fonts without deactivating the pruning of single-byte fonts.

For information about font pruning, see Prune double-byte fonts and Prune single-byte fonts.

Captured fonts

Some printers, such as the 3130 printer, can capture downloaded fonts. Captured fonts automatically become new temporary printer-resident fonts. On printers with lower-speed attachments, captured fonts improve performance on future jobs that use the same fonts; on printers with higher-speed attachments, captured fonts can decrease performance. For more information, see Chapter 20, "Marking fonts for font capture," on page 231.

Expediting resource deletion

At the end of data set processing for a job, PSF deletes user library and inline resources from the printer, and then uses reasonable resource loaded values (RRLVs) to manage whether more resources are deleted (see "Resource-deletion call (RDC)" on page 311). This deletion process can take time and might affect performance.

To expedite resource deletion and improve performance, you can set the **resource-deletion-ratio** parameter in the AFPPARMS control statement or the **Resource deletion ratio** parameter in the Printer Inventory. The value for the parameter is a number that determines whether PSF deletes all fonts, overlays, object containers, and page segments at the end of data set processing, without considering RRLV settings. If the ratio of inline and user resources to the total number of loaded resources (expressed as a percentage) is greater than or equal to the parameter value, PSF expedites resource deletion. If the ratio percentage is less than the parameter value, PSF's standard methods are used to delete the resources. The parameter value is 0 - 100, with a default of 90. If the value is 0, PSF does not expedite resource deletion.

When resource deletion is expedited, subsequent print jobs can start sooner. However, any resources that are needed for the next print job, including overlays that might be specified in the Printer Inventory or PRINTDEV statement, are retransmitted to the printer, which might negatively affect performance. Consider these guidelines when you are deciding whether to expedite resource deletion:

- Expedited resource deletion occurs more frequently with a lower parameter value and less frequently with a higher parameter value.
- Set the parameter value to a low number if resources are not typically used by subsequent jobs or if resources for most jobs are inline.
- Set the parameter value to a high number if delays are caused by reloading system resources between jobs.
- Resource deletion is not expedited if XTP7NDEL calls Exit 7 to delete or retain resources at data set end (DSE). For more information, see "Using RDC to control resources" on page 312.

Size of print jobs

Printing several short print jobs requires that job-to-job transition processing is run frequently. Therefore, short jobs require more processor power than longer jobs, and can affect printer throughput.

Using outline fonts

PSF supports both FOCA outline fonts and TrueType and OpenType outline fonts. The benefits of using outline fonts are:

- The character set for an outline font can be smaller than its equivalent for a raster font, thereby requiring less time to download data.
- Outline fonts of different point sizes can share the same character set in the printer, which reduces the time that is needed to download.
- Code pages can be shared, which also reduces the time that is needed to download.

When the time required to download raster fonts would affect printer throughput, the use of outline fonts might improve printer performance. For information about how to use the Printer Inventory, the PRINTDEV statement, or Exit 7 to automatically map raster fonts to FOCA outline fonts (for printers that support outline fonts), see "Mapping to outline fonts" on page 236 and "Considerations related to FOCA outline fonts" on page 300. This mapping slightly increases PSF's use of the central processor.

When you request that a FOCA outline font is reloaded, you might experience some performance degradation if several coded fonts are using the same code page or character set. All the fonts that are sharing the code page or character set are deactivated and then reactivated, all from the same reload request. (Deactivating and reactivating an outline coded font does not make it necessary to send all the font data to the printer again; only the commands necessary to associate the code page and the character set are sent.)

Transmission and compression of data

When PSF is driving TCP/IP-attached printers or SNA-attached printers that are communication-attached, PSF can compress line data to use communication lines more efficiently. You can activate line-data compression by specifying the optional **Blank compression** parameter in the Printer Inventory (see Blank compression) or the COMPRESS keyword in the PRINTDEV statement (see COMPRESS).

PSF itself can do line-data compression, which is activated by specifying the optional **Blank compression** parameter in the Printer Inventory or the COMPRESS keyword in the PRINTDEV statement.

Line-data compression might improve performance because PSF can compress continuous blanks and trailing blanks and can replace strings that contain more than five blanks with Repeat String presentation text controls. Line-data compression is most likely to improve printer throughput when the following conditions are met:

- The printer is attached with a low-speed attachment.
- Printer throughput is limited by the attachment.
- The print job contains line data. PSF does not compress print jobs that contain other kinds of data.
- The line data contains a significant number of blank strings or trailing blanks that contain more than five blanks.

Note: If the line data does not have these characteristics, the use of data compression might degrade performance. For more on blank compression, see "Blank compression for line data" on page 196.

The attachment hardware

If you install a printer with an attachment that sends data at a rate below the maximum rate of the printer, the printer can print more slowly than its mechanical speed. The attachment methods for printers include:

- A non-SNA printer can be host-connected and channel-attached.
- A printer can be local-attached.
- A printer can be communication-attached.
- A printer can be TCP/IP-attached.

Host-connected, channel-attached printer

Channels transfer data fast enough that printer throughput is not typically limited. When you configure a system, consider the total data rate requirements of all the devices attached to a channel.

Specify that the channel adapter is to be set in high-speed data-streaming mode if possible. To set this mode, see the appropriate processor and printer publications. The use of other modes can affect printer throughput, in ways that depend on the printer, the application, and the length of the printer channel cable.

An ESCON channel can send and receive data faster than a parallel channel. With the ESCON channel adapter installed in a printer, the printer can be attached to the host with an ESCON channel or a FICON Bridge channel. Therefore, IBM ESCON channel-attached printers that have an ESCON PCI channel adapter with Release 9.6 microcode and the 64 KB I/O buffer microcode enhancement installed might provide more performance improvements. For information about the ESCON channel adapter support for a printer, see the documentation provided with the printer; for more information about the FICON Bridge channel, see the FICON Bridge feature in the appropriate ESCON Director publication.

Local-attached printer

For a description of attachment methods, see "Local-attached devices" on page 17.

The capacity of a control unit can limit the throughput of the attached printers. When PSF uses VTAM, more processor usage is required than with channel attachment.

Communication-attached printer

For a description of attachment methods, see "Communication-attached devices" on page 18.

The speed of the communication line can be the most important performance factor for a communication-attached printer. If multiple printers are attached to the same communication line or network path, the total capability of the network is shared and can limit printer throughput.

When PSF uses VTAM, more processor usage is required than with channel attachment.

Table 22 on page 191 shows the approximate maximum data rates that can be reached with the attachments listed. The rates that are shown are not always achievable. The results depend on network and system loading, tuning parameters, and the characters of the printer data stream. The attachment can limit printer throughput.

Table 22. Approximate maximum attachment data rates

Communication line speed	9600 bits per second	19200 bits per second	56000 bits per second
Maximum printer data rate	1000 bytes per second	2200 bytes per second	6700 bytes per second

PSF direct or distributed print function (DPF)

Printers can be remotely attached with PSF direct or a distributed print function (DPF).

PSF direct

In a PSF direct configuration, PSF sends the print data set over an SDLC line, local token ring, or remote token ring to PSF direct, which passes the data set to a printer attached to a print server.

Note: PSF direct does not support remote resource caching.

When you are installing and configuring PSF direct, consider these:

- Data rate capability of the printer
- Data rate capability of the host attachment
- Data rate requirements of the print data set

Note: The use of data stream transforms, such as AFP to PCL, can decrease throughput.

DPF

DPF works with Resource Saving either OFF or ON.

In a DPF configuration, PSF sends the print data set over an SDLC line, local token ring, or remote token ring to DPF, to be spooled and later printed. DPF requires only one print server, which spools the print data sets in one stage and sends them to be printed in another stage.

The maximum data rates in DPF can be lower than data rates with PSF direct, depending on attachment speeds.

Size of the VTAM request unit (RU)

A VTAM request unit (RU) contains system data information for a printer. An increase in VTAM RU size might increase printing rate because processing larger (and therefore fewer) RUs contributes to higher efficiency.

However, the maximum size of the RU is limited and dependent upon the limitations of hardware and software. For example, the RU size is limited by the communications controller or by the maximum buffer size that is allowed by the printer, whichever is smaller. Generally, use the largest RU size suitable for the printer or attachment that is used.

VTAM PSNDPAC and DELAY parameters

VTAM includes a coat-tailing algorithm that is designed to minimize the number of buffers VTAM processes. In intermediate-speed and high-speed communication configurations, this algorithm can affect the performance both of communication-attached printers and of DPF or PSF direct.

- In intermediate-speed communication configurations, you can reduce or eliminate coat-tailing delays by specifying a large PSNDPAC pacing value in your VTAM logon-mode table entry.
- In high-speed communication configurations, you must eliminate coat-tailing delays to achieve full use of high-speed communication links. VTAM provides a DELAY parameter for this purpose; specify DELAY=0.

For more information about coat-tailing, see *z/OS Communications Server: SNA Customization*.

NCP MAXOUT value

MAXOUT (maximum output) is an operand in the physical unit (PU) macro instruction for the Network Control Program (NCP). The value that is specified for MAXOUT defines the number of units of data that the host system can send to the PU before the PU must respond. You must specify a value 3 - 7. A value less than the maximum supported by the PU can cause delays in the communication line, and thus degrade performance. For more information, see *NCP: System Support* for your system.

TCP/IP-attached printer

See Chapter 5, "Connecting a TCP/IP-attached IPDS printer," on page 41 for more information about TCP/IP-attached printers.

The throughput of TCP/IP-attached printers can be limited by the following factors:

- The configuration of your Internet Protocol networks.

  The default maximum transmission unit (MTU) size, which is specified on the GATEWAY statement in the TCP/IP profile on the z/OS system, might slow the transmission of data to the printer. For high-speed printers on a token-ring network, IBM recommends an MTU size of 4096 bytes and a dedicated LAN. Otherwise, for printers on an Ethernet network, the MTU size must be below 1500 bytes. For more information, see "GATEWAY" on page 45.

- A large amount of network traffic.
- Communication-line speed, if data is transmitted over telephone lines.

Table 22 on page 191 shows the approximate maximum data rates that can be reached on telephone lines with different line speeds. The rates that are shown are not always achievable; they depend on network and system loading, tuning parameters, and the characteristics of the printer data stream.

- A buffer size of less than 32768 specified in the TCP/IP profile.

  See "Understanding performance issues " on page 48 for more information about setting TCPSENDBFRSIZE to the optimum buffer size.

Also, when PSF sends data to TCP/IP-attached printers, more processor usage is required than for channel-attached printers.

The host system

The following factors that are related to the host system can affect printer throughput:

- Processing capability
- Amount of real storage available
- Capacity of the I/O subsystem
- System tuning

Processing capability

The capability of the host processor can affect printer throughput. The greater the density or complexity of a print job on higher-speed printers, the greater the processing capability required. A configuration with inadequate processor power, or with a large load other than printing, can limit throughput for dense or complex applications that are printed on a high-speed printer.

For print jobs, the processor requires time to do each of the following tasks:

- Read the data from auxiliary storage.
- Convert the data to a printer data stream.
- Send the data to a printer.
- Manage resources.
- Manage checkpointing.
- Prepare for error recovery.
- Manage VTAM attachment protocol.

The speed of the printer and the characteristics of the print data set affect processing. High-speed, channel-attached printers place a greater demand on the processor than low-speed printers.

User applications other than PSF (for example, NJE) can affect the processor capacity available for printing and can limit printer throughput.

Amount of real storage available

Real storage holds PSF code, working storage used by PSF, and I/O buffers. A host-system processor with limited real storage can cause contention for real storage and DASD that can decrease printer throughput.

Real-storage requirements might be lower for low-speed printers than for high-speed printers.

Capacity of the I/O subsystem

The data rate capacity and speed of the DASD devices can affect printer performance. Some print data sets require more DASD I/O from the spool or the PSF resource libraries than other print data sets, depending on the density or complexity of the print data.

Some DASD devices, such as the 3390, can handle large amounts of image data while a cache controller can be used to hold frequently accessed resources.

If the print data set references large resource objects (such as color EPS objects or IOCA FS45 images), fast DASD (such as an Enterprise Storage Server®) can be used to improve performance.

Selecting region sizes

The region-size requirements for PSF depend on:

- Installation options in the PSF startup procedure
- The number of printers the region supports
- Requirements of the individual print job

During the initialization of the FSS and the first printer FSA, PSF code, control blocks, and various buffers must be stored. Each additional FSA requires storage for control blocks and buffers. Storage that is acquired during initialization remains static and is not released until the FSA is drained.

More storage is required for certain FSA control blocks while print jobs are being processed. This storage is dynamic, is acquired as needed, and is released after the print job is processed. Some print job characteristics require more storage and need to be considered when you calculate the region-size requirements. For example, if a print file contains a large amount of inline resources or uses color management, you might need to increase the region size.

Table 23 on page 194 shows the minimum storage required for PSF. The storage includes all subpools associated with each of the PSF-related task control blocks. In this table, the minimum storage above 16 MB for channel-attached FSAs is based on a buffer parameter value of 4. See "Buffers" on page 197.

Table 23. Minimum storage required for FSSs and FSAs. MB = megabyte.		
FSS or FSA	**Channel-attached**	**SNA- or TCP/IP-attached**
FSS and the first FSA	0.6 MB below 16 MB 5.7 MB above 16 MB	0.5 MB below 16 MB 7.0 MB above 16 MB
Each additional FSA in the FSS	0.3 MB below 16 MB 1.6 MB above 16 MB	0.4 MB below 16 MB 2.9 MB above 16 MB

In addition, the operating system allocates and page-fixes at least 4 KB of storage below the 16 MB line for each FSA to do I/O to the DASD associated with the PSF resource libraries. If you have a large number of printers that are inactive much of the time, see Close libraries when idle for more information.

Installation options

The following installation options can affect storage requirements during the initialization of PSF:

- Size of the PSF internal wrap trace table
  - PSF default is 32 (128 KB) per FSA.
  - For more information about tracing, see *PSF for z/OS: Diagnosis*.
- Requirements for the exit program:
  - Size of all installation exits (same in LINKLIB)
  - Per FSA:

    Size of the exit work area as defined in APSXUCOM
    Size of GETMAINS coded by the installation exits
- Resource buffer pool
  - The buffer pool is calculated by taking the smaller of one of the following values and multiplying that value by the number of FSAs:
    - Largest block size of all resource libraries multiplied by 4
    - Largest block size of all resource libraries multiplied by 2 and added to the next largest block size multiplied by 4

- – DCB=(BUFNO=*nnnnn*) can be coded on the DD statement in the resource library. This causes the allocation of an additional amount of virtual storage, which is calculated as:

  ```
  (BUFNO parameter) * resource library block size
  ```

- I/O buffer area

  The printer I/O buffer is calculated with the following formula:

  ```
  (buffer parameter) * 32 KB per FSA
  ```

  The buffer parameter value is obtained from the Printer Inventory (**Channel buffer count**) or the PRINTDEV statement (BUFNO); see Channel buffer count for more information. It specifies a factor for allocating real, fixed storage used in printer I/O operations. (The buffer parameter value is multiplied by 32 KB (32768 bytes) to determine the total amount of reserved storage.) The value can range from 1–10000.

  Below-the-line storage for all channel-attached printers, except the 3800 printer, can be 6–200 KB per FSA with buffer parameter values 1–10000.

  Below-the-line storage for the 3800 printer can be 38 - 486 KB per FSA with buffer parameter values 1–15. For 3800 printers, PSF uses the maximum value of 15 if a buffer parameter value greater than 15 is specified.

  Above-the-line storage for all channel-attached printers, except the 3800 printer, can be 32 KB–320 MB per FSA with buffer parameter values 1–10000. No above-the-line storage exists for the 3800 printer.

  Note: The buffer parameter is ignored for TCP/IP-attached printers and for SNA printers that are using VTAM.

  For more information, see "Buffers" on page 197.

Requirements for each FSA

These are some characteristics of print jobs that place additional storage requirements on each FSA:

- Number of resources that are required to print the document: 0.5 KB for each resource
- Number of error messages generated while formatting and printing the document: 0.2 KB for each message group
- Size of installation-generated page definitions and installation-generated form definitions: approximate size of the resource
- Size of the conditional processing page buffer: maximum number of text characters on a page, multiplied by 1.5

Selecting MEMLIMIT sizes

The MEMLIMIT parameter specifies the total amount of usable virtual storage that is above the bar in a single address space.

PSF uses storage above the bar only if `above-the-bar-storage=inline-resources` is specified on the AFPPARMS control statement or `Above the bar storage=Inline resources` is specified in the Printer Inventory. Storage that is above the bar is used only for inline resources. The amount of storage required depends on the number and size of inline resources in a job.

For more information about MEMLIMIT, see *z/OS MVS JCL Reference*.

System tuning

A host system that is running many applications and driving printers with PSF must be tuned correctly, or balanced, to satisfy the data-rate requirements of the printers.

Setting Workload Manager goals

With the Workload Manager (WLM) component of z/OS, you can manage workload distribution, balance workload, and distribute resources to competing workloads. To define WLM goals for PSF, put the PSF started tasks in a high-priority started task (subsystem type STC) service class.

For more information about Workload Manager, see *z/OS MVS Planning: Workload Management*.

Blank compression for line data

The **Blank compression** parameter in the Printer Inventory (see Blank compression) or the COMPRESS parameter in the PRINTDEV statement (see COMPRESS) provides a data compression function for line data. In line data that contains more than five contiguous blanks, PSF compresses the blanks to a single blank, plus control information for the printer to use in extracting the blanks.

Blank compression is most likely to improve printer throughput when a printer and the jobs that are printed have the following characteristics:

- The printer has a low-speed attachment.
- Printer throughput is limited by the attachment.
- The jobs are line data. Blank compression is not attempted for jobs of other types.
- The line data contains a significant number of interior strings of more than five blanks.

Blank compression can be used for host-connected, channel-attached printers, for SNA-attached printers, and for TCP/IP-attached printers. However, IBM does not recommend that you use it for host-connected, channel-attached printers, unless the attachment is the main performance bottleneck.

Note: Unless the printer and job characteristics are present, you might not see improvement in printer throughput. Increased processor cycles might be used to do the blank compression, but if the line data does not contain significant interior blanks, the volume of data that is generated might not be decreased enough to affect the overall performance of the attachment and the printer.

For unformatted line data that contains fields of data whose placement is specified in a page definition, for example, blank compression might give little or no improvement in throughput. The data sent to the printer might not contain interior blanks, although the text on the spool contains many blanks.

The same might be true of formatted line data that has no interior strings of consecutive blanks, when JES eliminated all trailing blanks because blank truncation was specified.

Because blank compression is an FSA parameter and affects all line data jobs sent to the printer, the system programmer must carefully check the characteristics of the jobs that are printed, and the listed characteristics for the attachment and printer, before the new parameter is implemented.

ACK per sheet

The Printer Inventory, Exit 7, or Exit 16 can be used to indicate to PSF to request an acknowledgement every sheet rather than an acknowledgement every page. This reduces the number of acknowledgements for jobs that are printing multiple pages per sheet (such as N_UP, duplex, and cut-sheet emulation), thereby improving the performance for these jobs.

ACK per sheet improves performance only when PSF is driving channel-attached printers. If you are using channel extenders, check with the vendor of your channel extender before you enable ACK per sheet. Many channel extenders do not work correctly when ACK per sheet is enabled.

For information about specifying ACK per sheet in the Printer Inventory, see Acknowledgement level.

Printer data rate requirements

The data rate to all-points-addressable (APA) printers increases with page density until the control unit in the printer is processing at its maximum rate. This data rate can be high, requiring substantial processor use.

The data rate capability of high-speed printers is typically greatest for uncompressed images. The data rates are high enough to warrant special attention when you configure and balance the host to the capabilities of the printers. Host processor use is related to the rate at which data is sent to the printers. For information about the characteristics of data, see "The print job" on page 186.

Data rates are highest for image data printed on high-speed printers. Image data can be dense, with hundreds of thousands of bytes per page, and requires high data rates. Data rates are lowest for text data printed on low-speed printers. Text data is not as dense as image data, and requires lower data rates.

Printer data checks

For normal operation, specify ALL for the **Print error reporting** parameter in the Printer Inventory (see Print error reporting) or use UNBLOCK in the DATACK parameter on the PRINTDEV statement in the PSF startup procedure (see DATACK). This causes PSF to report and correct errors that can affect printer throughput in some circumstances.

Buffers

The use of input (DASD) and output (printer) buffers can improve printer throughput.

The number and size of input buffers in the spooling subsystem[6] can affect the amount of resources available for a print job, and thus affect the printer throughput. As the number and size of buffers are increased to get more efficient processing, more real storage is required but less processor usage per page is used. As print rate increases, processor usage per unit of time increases.

For each printer started, PSF acquires a resource library input buffer pool. The size of the buffer pool is based on the printer model started and on the sizes of the resource libraries specified in the PSF startup procedure.

If a BUFNO parameter is specified on the DD statement of a resource library, an additional buffer pool is acquired exclusively for that resource library. This is generally not required for most printers and print jobs, but it can be considered for the page segment library when large, simple images are being printed on high-speed printers.

PSF ordinarily maintains a minimum of six page definitions and form definitions in virtual storage for each started printer between data sets. At maximum, PSF can maintain as many page definitions and form definitions as virtual storage can hold. You can use the resource-management Exit 7 to change the values that represent the page definitions and form definitions specified in the PSF startup procedure for the data set header and trailer pages, the messages data set, and the printer defaults. You can also change the specifications in the print data set, regardless of where they were first specified.

For channel-attached printers, the size of I/O buffers is controlled by the buffer parameter in the Printer Inventory (**Channel buffer count**) or on the PRINTDEV statement (BUFNO). For more information, see Channel buffer count.

For each printer not attached to VTAM or TCP/IP, PSF acquires an I/O buffer equal to 32 KB times the buffer parameter value in the Printer Inventory or the PRINTDEV statement. The I/O buffer pool is fixed in storage above the 16 MB line.

For 3800 printers, PSF uses the maximum buffer parameter value of 15 if a value greater than 15 is specified.

As a rule, the buffer parameter value is set to the number of 32 KB buffers needed to support channel printer I/O operations. Table 24 on page 198 recommends buffer parameter values based on the complexity of your page. If ACK per sheet is enabled for a job that is printing multiple pages per sheet (such as N_UP, duplex, or cut-sheet emulation), multiply the buffer parameter values listed in the table by the number of pages per sheet. For example, if a job is printing N_UP 4 and duplex (8 pages per sheet), multiply the buffer parameter value by 8.

---

[6] As specified in JES2 trackcell parameter, for example. A TRKCELL value of at least 5 is recommended.

Table 24. Recommended buffer parameter values	
Value	**Application Types**
1–2	Light and simple text; little image or graphics.
3–5	Dense complex text; moderately complex graphics; less than 5 square inches of 240-pel uncompressed images; highly compressed (20:1) full page 240-pel images.
6–20	All text; nearly all graphics; less than 15 square inches of 240-pel uncompressed images; moderately compressed full-page 240-pel, 300-pel, and 600-pel images.
21–100	All text; all graphics; full-page 240-pel uncompressed images; poorly compressed full-page 240-pel, 300-pel, and 600-pel images.
>100	Poorly compressed full-page 600-pel images; large complex 600-pel color images.

Block size

The physical block size of the spool files and resource libraries can affect the speed and efficiency with which the resources that are needed for a print data set can be accessed. Larger block sizes result in more efficient movement of data, which can reduce the load on the processor, the DASD, the device controllers, and the channels. However, larger blocks require more real storage.

Increasing the block size can increase processor usage by increasing the print rate. Increasing the block size for jobs that are printing at rated speed can decrease processor usage. Efficient removal of data from the spool is controlled by initialization parameters in the spooling system. For more information, see the publications for your system.

For JES2, set the TRKCELL value to at least 5. TRKCELL values of 10 or 12 might be needed for best performance with high-speed printers and applications with much data (full-track buffering on 3380 and 3390 devices). For information about enabling and setting TRKCELL, see the JES2 initialization and tuning publications for your operating system.

For JES3, the GRPSZ parameter is used to optimize spool performance. For information about GRPSZ, see the JES3 initialization and tuning publications for your operating system.

Data separation

To avoid contention for I/O, printer resources must be spread evenly across channels, volumes, and actuators. For example, the spool volume and the PSF resource library that holds page segments must be on different channels, and both must be separated from the printer channel.

Functional subsystem interface (FSI)

In deferred-printing mode, JES communicates with PSF through the functional subsystem interface (FSI). As an FSI system, PSF is marked pageable but not swappable. For efficient use of operating system paging and swapping facilities, PSF and TSO users must be assigned to different performance groups.

Checkpointing

A checkpoint provides the information that is needed to restart a print data set after an interruption or a recoverable error condition. When PSF restarts, it instructs the printer to discard all unprinted pages, reposition to the last checkpoint, and resume printing from that point. If no checkpoints are taken, PSF repositions to the beginning of the print data set.

The checkpoint interval can be specified in pages or in time. When you choose a value for it, you must bear in mind the speed of the printer, the characteristics of typical print data sets sent to the printer, and the following considerations:

- A checkpoint interval that is too small—that is, too frequent—can increase processor usage and decrease printer throughput.
- A checkpoint interval that is too large makes the recovery of a print job too lengthy. The reason is that, for SNA Logical Unit 1 (LU 1)-attached printers and TCP-IP-attached printers that can produce asynchronous data stream errors,[7] PSF issues a print buffered data command each time a checkpoint is taken. This command is required to make sure that the printer completely checked the data for errors. After PSF issues this data command, it must wait until all the pages that are sent to the printer are printed. If the checkpoint intervals are too frequent, the performance of these printers can be slowed.

---

[7] Some examples of TCP/IP-attached and LU 1-attached printers that produce asynchronous data stream errors are the 3812, 3816, 3912, 3916, 3930, and 4028 printers.

Chapter 13. Using PSF display functions

Instead of creating a PSF trace or memory dump, PSF display functions can be useful in these diagnostic situations:

- You need to know the status of the TCP/IP connection.
- You need to know the processing status of AFP Download Plus.
- You need to verify a printer's information that is available from the Sense Type and Model (STM) IPDS command and the Execute Order Homestate (XOH) Obtain Printer Characteristics (OPC) IPDS command.

This information describes the display functions used to obtain the status of the TCP/IP connection, the processing status of AFP Download Plus, and information about a printer.

Displaying TCP/IP status

If you need to know the status of a TCP/IP connection, you can do so without taking a PSF trace. By using the MVS MODIFY command as a PSF or AFP Download Plus operator interface, you can dynamically display TCP/IP status on the console and system log; however, it can be used only in deferred-printing mode. For more information about using the MODIFY command to display TCP/IP status, see *PSF for z/OS: Diagnosis* for the PSF operator interface and *PSF for z/OS: AFP Download Plus* for the AFP Download Plus operator interface. For an example of the TCP/IP status that is displayed on the console and system log for Download for z/OS, see *PSF for z/OS: Download for z/OS*.

Displaying AFP Download Plus status

A system programmer or operator who needs to know the processing status of AFP Download Plus can do so without taking a PSF trace. By using the MVS MODIFY command as an AFP Download Plus operator interface, you can dynamically display the AFP Download Plus status on the console and system log. Because AFP Download Plus transmits data through TCP/IP connections, the command can be used only in deferred-printing mode.

For more information about using the MODIFY command to display AFP Download Plus processing status, see *PSF for z/OS: AFP Download Plus*.

Displaying printer information

If you need to obtain the functional characteristics of a printer, you can do so without creating a PSF trace or memory dump. By using the display function in PSF, you can display the information in the system log or write it to a data set so you can view it or format and print it. The display function takes the printer information available with the IPDS STM and the XOH-OPC command responses and displays it to you.

These methods are used for displaying printer information:

- Activate the display printer information function in the PSF startup procedure. A DD statement in the PSF startup procedure defines the data set to which the printer information is sent. PSF always saves printer information for each printer defined in the startup procedure.
- Use the PSF operator interface to specify the display printer information parameter. This interface can dynamically display printer information in the system log. You can use the MVS MODIFY command to display the printer information in the system log; however, it can only be used in deferred-printing mode.

Requesting printer information with the PSF startup procedure

You use a DD statement in the PSF startup procedure to specify the data set where the printer characteristics information is written for each printer defined in the startup procedure. PSF creates a

member in the data set for each printer. The member name in the data set is the printer FSA name, such as PRT612. Before you specify the data set in the DD statement, you must allocate a partitioned data set (PDS). After information is saved to the data set, you can view and print the printer information report.

Allocating the partitioned data set

You must allocate the PDS as PDSE to reduce problems with insufficient directory blocks and storage space and to allow multiple FSAs to open the data set without a system abend. IBM recommends that the PDS have the attributes specified in Table 25 on page 202.

Attribute	Value	Type	Description
DCB=DSORG=	PO	Required	PDS data set organization
DSNTYPE=	LIBRARY	Required	Data set defined as PDSE
DCB=RECFM=	VB	Required	Variable blocked records
DCB=LRECL=	512	Required	Maximum bytes in record
DISP=	SHR	Required	Data set can be used by multiple print files
SPACE=	(CYL,(*nn*,1,10))	Required	DASD cylinders needed to process data **Note:** Space requirements are dependent on the capabilities of the printer and the number of printers for which you want information. A primary allocation of two cylinders of DASD space is recommended as a starting point. More space is required if you have more than 15–20 printers for which you want to run reports.

Table 25. Recommended PDS Attributes

Specifying the DD statement

Figure 63 on page 202 shows the format of the DD statement you use for saving printer information to a data set. The DD name for the PRTINFO data set must always be PRTINFO. The data set name can be anything you want; however, IBM suggests that you use the name of the startup procedure and the PRTINFO qualifier to accurately identify the information that the data set contains. You should also use DISP=SHR to let multiple PSF FSAs simultaneously write printer information to the data set.

```
//PRTINFO    DD   DSNAME=fssname.prtinfo,DISP=SHR
```

Figure 63. Saving Printer Information to a Data Set

Figure 64 on page 202 shows the DD statement used to display printer information for a printer when the FSS name is APSWPROT.

```
//PRTINFO    DD DSNAME=APSWPROT.PRTINFO,DISP=SHR
```

Figure 64. Saving Printer Information to a Data Set (Sample)

Viewing and printing the report

After the printer information is saved to the data set, you can view the softcopy report (see "Softcopy report" on page 373 for an example of the source data). You can also format the softcopy report and print it as a hardcopy report.

To generate a hardcopy printer information report, use these IBM-supplied page and form definitions to format the softcopy report:

Page definition
    P1DPI01
Form definition
    F1DPI01

The page definition uses PPFA record formatting to define the resulting printer information report. It also uses proportional spaced and monospaced, sans-serif fonts from AFP Font Collection Version 2. Because the report contains unprintable characters returned by the printer for some fields, IBM recommends that you specify DATACK=BLOCK in your print request to avoid messages about incorrect characters.

Figure 65 on page 203 shows an example of a job stream that formats the softcopy printer information report for printing as a hardcopy report.

```
//JOB1      JOB    ...
//STEP1     EXEC   PGM=IEBGENER
//SYSPRINT  DD     SYSOUT=*
//SYSIN     DD     DUMMY
//OUTRL     OUTPUT PAGEDEF=DPI01,FORMDEF=DPI01,DATACK=BLOCK
//SYSUT2    DD     SYSOUT=*, OUTPUT=*.OUTRL
//SYSUT1    DD     DSN=APSWPROT.PRTINFO(PRT612),DISP=SHR
```

Figure 65. Example of the Job Stream for Formatting the Printer Information Report

For an example of a hardcopy printer information report, see "Hardcopy report" on page 359.

Requesting printer information with the MODIFY command

A system programmer or operator who needs to obtain the functional characteristics of a printer can do so by specifying the display function with the PSF operator interface. The display function takes the printer information available with the IPDS STM command and the XOH-OPC command responses and dynamically displays it in the system log.

The MVS MODIFY command is used to display the printer information in the system log; however, it can only be used in deferred-printing mode, PSF must be connected to the printer, and the printer must be in ready status. PSF is connected to the printer when APS933I has been issued and a disconnect message such as APS929I has not been issued.

Figure 66 on page 203 shows the format of the MODIFY command used for displaying printer information.

```
{MODIFY | F}  fss_name,DISPLAY,fsa_name,DATA=PRTINFO
```

Figure 66. MODIFY Command Format for Displaying Printer Information

To display printer information, the PSF operator enters the MODIFY (or F) command with the DISPLAY and DATA=PRTINFO parameters.

Note: The AFP Download Plus feature of PSF does not support the DATA=PRTINFO parameter.

The parameters used with the MODIFY command to display printer information are:

fss_name
    Specifies the name of the FSS for which information is displayed. This parameter is required and must match the FSS parameter for the JES2 FSS statement or the FSSNAME parameter for the JES3 FSSDEF statement.
DISPLAY
    Specifies that information is displayed in the system log. This parameter is required.
fsa_name
    Specifies the name of the printer for which information is displayed. This parameter is required.
DATA=PRTINFO
    Specifies that information from the STM and OPC is displayed in the system log.

Figure 67 on page 204 shows the MODIFY command used to display printer information when the FSS name is APSWPROT and the printer name is PRT614.

```
MODIFY APSWPROT,DISPLAY,PRT614,DATA=PRTINFO
```
Figure 67. Displaying Printer Information (Sample)

For an example of printer information displayed in the system log, see .

Chapter 14. Finishing all pages in a job as a group

You can configure PSF so that an entire job (including the header, separator, message, and trailer pages) is finished together as a single group. This function is called print job finishing.

Enabling PSF for print job finishing

To request finishing, you specify the Medium Finishing Control (MFC) structured field in the Document Environment group of a form definition. See the *Mixed Object Document Content Architecture Reference* for more information.

Header pages, separator pages, user's pages, message pages, and trailer pages are all considered separate data sets by PSF. The print job finishing function tells PSF to keep all of the data sets together in one finishing group. To use the print job finishing function to include the entire job in the finished group of output, you must specify the MFC structured field in the form definition used by the header page. The MFC structured field must specify a scope value of print job (X'06').

You specify the form definition used by the header page in the JOBHDR DD of the PSF startup procedure. Therefore, all printers that are defined to use this header form definition and all jobs sent to these printers have the print job finishing applied.

This example shows a form definition that is used by the header page that requests that the entire job is stapled in the default corner:

```
FORMDEF    STPHDR
/***********************************************************/
/*  FINISH subcommand with PRINTJOB scope: corner staple    */
/***********************************************************/
    FINISH SCOPE PRINTJOB    OPERATION CORNER
         REPLACE yes;
```

For more information about coding print job finishing, see *Page Printer Formatting Aid: User's Guide*.

This example shows the JCL in the PSF startup procedure that is used to reference the form definition with the finishing function for the header page:

```
//JOBHDR3  OUTPUT PAGEDEF=V06483,
//         FORMDEF=STPHDR,CHARS=60D8
              :
              :
//PRT619    CNTL
//PRT619    PRINTDEV FONTDD=*.FONT03,
//         FONT300=*.FONT03,
//         FONT240=*.FONT02,
//         OVLYDD=*.OLAY01,
//         PSEGDD=*.PSEG01,
//         PDEFDD=*.PDEF01,
//         FDEFDD=*.FDEF01,
//         JOBHDR=*.JOBHDR3,
              :
              :
//PRT619    ENDCNTL
```

What if you don't want to finish all jobs sent to a printer

Because print job finishing is specified in the form definition for the header page, by default, all jobs sent to the printer that uses that header page are finished. If you need to print some jobs to this printer without print job finishing applied, you can do one of these:

- Define another logical printer in the PSF startup procedure for the same physical printer in this way:

  1. Use a form definition for the header page that does not request print job finishing.

2. Define that the logical printers share the physical printer (see "Using PSF parameters to attach and share printers" on page 118 for more information).

3. Set up a work-selection criteria to send the jobs you don't want finished to the newly defined logical printer.

- Set the XTP7PJO flag in Exit 7 to suppress print job finishing for certain jobs. See "XTP7PJO" on page 438 for more information.

Things to consider when using print job finishing

- Print job finishing adds a new outer level of finishing around the existing job. Any finishing operations that are currently specified by the data sets within the job become nested finishing operations. The printer and finisher must support this nesting or errors are generated and the existing finishing operations are not done. Some nested finishing operations are supported and others are not. See your printer and finisher documentation for the supported finishing operations.

- All pages within the print job are always included. It is not possible to selectively include some finished pages while others are excluded. For example, you cannot specify that the header is finished with the job, but the message pages are not finished.

- The form definition that is used by the header page initiates print job finishing; therefore, you must be running with header pages turned on.

- Suppressing interrupt message pages is suggested when PSF is attached to a printer with finisher capability; otherwise, interrupt message pages separate finished documents into two finished groups with the inserted page in the middle.

Chapter 15. Using cut-sheet emulation mode

Cut-sheet emulation (CSE) is the ability of a continuous-forms printer to provide output similar to output from a cut-sheet printer. (CSE mode is not supported when connected through a distributed print function (DPF) product or RPM3.)

Some continuous-forms printers[8] provide the option of printing in CSE mode. If the operator selects this option, the printer logically divides a continuous-forms sheet in half, parallel to the carrier strips. Then, PSF automatically prints jobs 2 up, treating each half of the sheet as if it were a single sheet of paper. If the printer is loaded with forms that measure 17 x 11 inches (not counting the carrier strips), PSF treats each 8.5 x 11 inch half-sheet the same as an 8.5 x 11 inch sheet on a cut-sheet printer. For a simplex job, page 1 is printed on the first half of the sheet, and page 2 on the second half. For a duplex job, page 1 is printed on the front of the first half, page 2 on the back of page 1, page 3 on the front of the second half, and page 4 on the back of page 3.[9]

When the output is slit and collated by post-processing equipment, it is identical to output printed on 8.5 x 11 inch cut-sheet forms.

How do I get cut-sheet emulation working

To use cut-sheet emulation mode, the operator must select the CSE function on the printer. When the CSE function is selected, PSF automatically ensures that all data sets are printed 2 up unless N_UP is specified in the active copy group of the form definition. The N_UP specification in the form definition overrides CSE because the purpose of N_UP is to give the job submitter control over how the printer places multiple logical pages on a sheet.

Do I have any control over using cut-sheet emulation

For all copy groups in the form definition that do not specify N_UP, PSF defaults to CSE when that mode is enabled at the printer. However, by using the Printer Inventory or installation Exit 7, you can request that PSF check page sizes to determine whether CSE is used. You have two options for checking page size: you can request that PSF check only the first printed page for which a new copy group is used, or you can request that PSF check each front side page. When either of these options is specified, PSF checks the Page Descriptor (PGD) structured field, either in the page definition or in the MO:DCA page, to see whether the page is small enough to fit 2 up on the sheet. If the page is not small enough, CSE is not used; the page is printed in normal 1_UP.

Note: Both options are used only for non-N_UP copy groups.

Will cut-sheet emulation work with 3800 Line-Mode Conversion

If you are using the 3800 Line-Mode Conversion function and you enabled CSE on the printer, PSF builds a page definition that takes advantage of CSE. However, PSF cannot determine whether your 3800 job is designed for wide or narrow paper. Installation Exit 4 has an option that you can use so that PSF knows which paper to emulate. For more information, see "Logical-record processing exit: Exit 4" on page 285.

---

[8] For the printers and microcode level to support CSE, consult the IBM service representative for your hardware.

[9] Some printers might offer options that enable the operator to control whether the first half of the sheet is the left or right side, and whether the front of the sheet is the top or bottom side of the physical form.

Things to consider when using cut-sheet emulation

- In CSE mode, PSF treats each half of a sheet of paper as a single physical sheet. An operation that causes a new copy group to be called, and therefore would ordinarily cause a sheet eject, now causes only a half-sheet eject. Numerous operations exist, including:

  - Change in medium overlay
  - Change to a different set of suppressions
  - Change in the number of copies of a page to be printed
  - Transition from simplex to duplex, or from offset to no offset

  Thus when two consecutive data sets are printed in CSE mode, the second starts on the next available half-sheet, without forcing a sheet eject. A single physical sheet of paper might contain the job trailer for job one and the job header for job two. If you need to prevent this, you can force the header page of a data set to eject to a new sheet by assigning it a form definition that specifies N_UP (N_UP 1 also works) or by setting the new sheet flag (XTP7NSF) in installation Exit 7.

- Form definition parameters PRESENT and DIRECTION, which PSF ordinarily ignores for cut-sheet printers, are accepted in CSE mode. CSE can accommodate a wide range of jobs in addition to those that are formatted specifically for cut-sheet paper, but it has an unfortunate side effect. Customers sometimes inadvertently code FORMDEF options other than the default PORTRAIT/ACROSS. Ordinarily, these options have no effect; but in CSE mode they might suddenly take effect, with unexpected and unwanted results.

- When PSF is running in CSE mode, the SMF type 6 records reflect the actual number of physical sheets of paper moved through the printer for the SMF6FEET and SMF6PGE fields. SMF6IMPS reflects the number of sides of sheets.

- If your application relies on PSF to place pages on the same half-sheet after a reposition or an operator command, PSF forces the number of interrupt message pages to be an even number of copies. You must still consider one of these options for the beginning of the data set:

  - Run with job headers active (Exit 1) and assign a form definition that specifies N_UP on the JOBHDR statement. If data set headers are required, you must also assign a form definition that specifies N_UP on the DSHDR statement.

  - Without job headers and data set headers active, set the XTP7NSF field on in installation Exit 7 (**CSE sheet eject** parameter in the Printer Inventory). XTP7NSF causes PSF to force each new data set to a new physical sheet.

  - With job headers, data set headers, or both active, set the XTP7NSF and XTP7CPPP fields on in installation Exit 7 (**CSE sheet eject** and **CSE preserve page position** parameters in the Printer Inventory). XTP7NSF causes PSF to force each new data set to a new physical sheet. XTP7CPPP causes PSF to preserve page placement by forcing the number of job header and data set header pages to an even number of copies.

- If your printer has an attached post-processor that can do left/right offset stacking, the offset stack (jog) command that is sent by the printer to the post-processor indicates which half-sheet is to be jogged. The printer determines which half-sheet is to be jogged by recognizing which sheet-half was being processed at the time when a jog was requested. If your post-processor cannot handle left/right offset stacking, the printer combines the jog commands so that if either half-sheet is to be jogged, the entire sheet is jogged. However, by using installation Exit 7 (XTP7NSF flag) or the Printer Inventory (**CSE sheet eject** parameter), you can force a sheet eject if your post-processor does not handle left/right offset stacking. This causes PSF to eject to a new sheet for the first page of a data set and for a copy group that contains a jog request that is called by an IMM in the data stream.

Chapter 16. Using Side and Edge Sensitive Paper Handling

You can use the Side and Edge Sensitive Paper Handling function in PSF to support side and edge sensitive paper on printers that do not include side and edge sensitive paper support. This function also provides improved performance for printers that do have this support.

This information describes side and edge sensitive paper, the existing support for side and edge sensitive paper, and how you can use Exit 7 or the Printer Inventory to select input paper bins for simplex and duplex.

Understanding side and edge sensitive paper

Side sensitive paper (such as letterhead) has a defined front and back. Edge sensitive paper (such as prepunched) has a defined edge where the holes are located.

For side sensitive paper, applications are coded with respect to the front and back of the form. For edge sensitive paper, applications are coded with respect to a particular edge.

Existing support for side sensitive paper

Many cut-sheet printers require that side sensitive paper is loaded one way for simplex, typically front-side down, and another way for duplex, typically front-side up. This requires either:

- Manual operator intervention to reload the bin (also called tray) between simplex and duplex jobs.
- Two bins that are loaded, one for simplex and the other for duplex. The application is coded to select the proper bin.

More sophisticated cut-sheet printers have support for side sensitive paper. The paper bins have a configuration setting for side sensitive paper, and the output prints properly for both simplex and duplex. However, the output prints by using the duplex paper path for both simplex output and duplex output, which cuts simplex throughput in half.

Existing support for edge sensitive paper

In addition to the requirements for side sensitive paper, many cut-sheet printers also require that edge sensitive paper is loaded differently for simplex and duplex. This is common for printers where the holes are on the leading or trailing edge as the paper moves through the printer. The paper is loaded with the holes either on the leading or trailing edge for simplex and rotated 180 degrees for duplex. This requires either:

- Manual operator intervention to reload the bin between simplex and duplex jobs.
- Two bins that are loaded, one for simplex and the other for duplex. The application is coded to select the proper bin.

More sophisticated cut-sheet printers have support for edge sensitive paper. The paper bins have a configuration setting for edge sensitive paper, and the output prints properly for both simplex and duplex. Also, the holes are stacked on the same side in the stacker. However, the output prints by using the duplex paper path for both simplex output and duplex output, which cuts simplex throughput in half.

Selecting input paper bins for simplex and duplex printing

When applications that are printing on printers with side and edge sensitive support are moved to printers that do not have this support, you encounter a migration problem. For example, the printer you are migrating from has prepunched paper loaded into Bin 1 and some applications printed simplex and others

duplex. For the new printer, the only automatic way to provide simplex and duplex support is to use two bins, where Bin 1 is loaded for simplex and Bin 2 for duplex. Unfortunately, all of the duplex applications use form definitions that are coded for Bin 1. You can modify the form definitions to use Bin 2, but this might not be practical if many form definitions are affected.

and describe how you can use PSF Exit 7 or the Printer Inventory to select input paper bins for simplex and duplex printing.

Using Exit 7

PSF is able to select different input paper bins for simplex and duplex printing, regardless of the bin specified in the form definition or in the JCL on the OUTPUT statement (INTRAY=). This function is activated by setting the XTP7SENS flag in installation Exit 7. Up to four different application bin values can be mapped to real printer bins for simplex and duplex.

The print application specifies a form definition that contains one or more copy groups. In each copy group, an input bin number is specified that identifies which input paper bin is to be used.

Exit 7 has fields that map the input bin number to two real printer bins, one for simplex and one for duplex. The input bin number is coded in field XTP7BINA, the real simplex bin is coded in field XTP7SIMA, and the real duplex bin is coded in field XTP7DUPA.

For example, if an application is coded to use edge sensitive paper from Bin 1, you can use the Side and Edge Sensitive Paper Handling function to use real printer Bin 1 for simplex and real printer Bin 2 for duplex. You would then code this in installation Exit 7:

- XTP7BINA = 1 (application or form definition bin number)
- XTP7SIMA = 1 (real bin number for simplex)
- XTP7DUPA = 2 (real bin number for duplex)

When the application requests simplex printing from Bin 1, PSF tells the printer to use real Bin 1. When the application requests duplex printing from Bin 1, PSF tells the printer to use real Bin 2.

Up to four logical bins can be remapped with these fields:

```
XPT7BINA    XPT7BINB    XTP7BINC    XTP7BIND
XPT7SIMA    XPT7SIMB    XTP7SIMC    XTP7SIMD
XPT7DUPA    XPT7DUPB    XTP7DUPC    XTP7DUPD
```

The logical sets are identified with a common suffix letter. Sample installation exits APSUX07V and APSUC07V can be modified to set any of these fields.

For printers that include side and edge sensitive paper support, you need to consider using the support in PSF to achieve better performance, if the printer has enough bins.

Using the Printer Inventory

PSF is able to select different input paper bins for simplex and duplex by using the Printer Inventory. This function is activated when bin values are specified in the Printer Inventory. Up to four different application bin values can be mapped to real printer bins for simplex and duplex.

The print application specifies a form definition that contains one or more copy groups. In each copy group, an input bin number is specified that identifies which input paper bin is to be used.

The Printer Inventory has parameters that map the input bin number (source tray) to two real printer bins (substitute trays), one for simplex and one for duplex. You can specify the source tray number, the simplex substitute tray, and the duplex substitute tray in the Printer Inventory by using the Infoprint Server ISPF panels or the Printer Inventory Definition Utility (PIDU).

To specify the tray values with the ISPF Printer Inventory panels, you use the **Input Tray Substitution** parameter. To specify the tray values with the PIDU statement, you use the **tray_substitutions** parameter. For example, if an application is coded to use edge sensitive paper from Source Tray 1, you can use the Side and Edge Sensitive Paper Handling function to use Substitute Tray 1 for simplex and Substitute Tray 2 for duplex. To do this, use one of these parameters:

• PIDU:

```
tray_substitutions={1->{1 2}}
```

Note: Braces must be entered as shown and do not indicate an alternative value.

• ISPF panel:

```
Input Tray Substitution:
   Source tray:  Substitute trays:
                 Simplex  Duplex
   _1_           _1_      _2_
```

When the application requests simplex printing from Source Tray 1, PSF tells the printer to use Substitute Tray 1. When the application requests duplex printing from Source Tray 1, PSF tells the printer to use Substitute Tray 2.

To specify multiple tray values, use one of these parameters:

• PIDU:

```
tray_substitutions={1->{1 2}2->{3 4}3->{1 4}}
```

Note: Braces must be entered as shown and do not indicate an alternative value.

• ISPF panel:

```
Input Tray Substitution:
   Source tray:  Substitute trays:
   tray:         Simplex  Duplex
   _1_           _1_      _2_
   _2_           _3_      _4_
   _3_           _1_      _4_
```

These examples map Source Tray 1 to Substitute Tray 1 for simplex and Substitute Tray 2 for duplex. Source Tray 2 is mapped to Substitute Tray 3 for simplex and Substitute Tray 4 for duplex. Source Tray 3 is mapped to Substitute Tray 1 for simplex and Substitute Tray 4 for duplex.

For printers that include side and edge sensitive paper support, you need to consider using the support in PSF to achieve better performance, if the printer has enough trays.

Chapter 17. Handling job streams that are formatted at different resolutions

Typically, production of printed data is a multi-step process. An application first formats and spools data for printing. Later the data is passed to a printer for presentation on paper.

In the past, printers printed at a single resolution (pel density) and all the printers in the enterprise printed at the same resolution. Jobs were formatted with resource libraries that contain resources at the same resolution as the printer.

Today, an enterprise might have printers with varied resolution. Some printers print with a resolution of 240-pels per inch, some printers print with a resolution of 300-pels per inch, and some printers can switch between resolutions. In addition, new resolutions for printers are being developed.

Furthermore, some jobs are dependent upon the resolution of the resource libraries in use at the time the job was formatted. If a job is formatted with resources of one resolution and printed with resources of another resolution, fidelity imperfections can occur.

In many cases, fidelity imperfections are not important. In these cases, formatting with resources of one resolution and printing with resources of a different resolution might be acceptable. In other cases, perfect fidelity might be required.

Some of the types of fidelity imperfections that can occur from failing to format and print with the same resolution resources are:

- Data no longer fits exactly in a box on a form.
- Columns for tabular data overlap or are spaced wider than intended.
- Right-aligned data might show a ragged right margin.
- Minor typeface differences might be observed.

In the past, to deal with these complications in an enterprise, printers were grouped by print resolution. By using routing options and print queues that are specified in the application, the application output got routed to a compatible printer. The PSF PRINTDEV JCL statement for the printer had to be set up to use resource libraries to print the application data that matched the resolution of the resources that were available when the application was run.

A better solution is provided with PSF where a single FSA can:

- Have access to multiple resource libraries of the same resource type but with different resolutions.
- Accept information from the application that indicates the resolution that is used to format the application output.
- Select the resource libraries with the correct resolution to print the data.
- Be given resource libraries to be used as default resource libraries when the application does not indicate what resolution was used to format the application data.
- Accept information from the data stream that indicates how to handle any mismatches between the resolution that is specified when formatting and the resolution of the resource found when printing.
- Use printers that accept print data formatted at multiple resolutions.
- Use printers that allow the manual switching of print resolution, for example the InfoPrint 4000.

The following information introduces the key concepts related to printing at multiple resolutions and refers to more detailed information:

- Multiple resolution printers
- Resource libraries segmented for multiple resolution
- Format resolution specified from the application

- Multiple resolution and installation exits
- Format resolution order

Multiple resolution printers

Two multiple resolution modes are now supported by PSF. This support is for printers that:

- Can be manually switched from one resolution to another (switchable resolution mode).
- Accept print data even though it was not formatted for the particular resolution of the printer (automatic or auto-resolution mode).

Printers with switchable resolution mode

PSF has support for printers that can be manually switched from one resolution to another.

Before this support, when PSF was attached to a printer that might be switched from one resolution to another, the following procedure was used:

1. Stop the FSA for the printer.
2. Switch the printer to the new resolution.
3. Start a new FSA that points to resource libraries compatible with the printer's new resolution.

This procedure required coordination for the stopping and starting of PSF and for the switching of the printer resolution.

Now with this support, a single FSA can be defined with multiple specifications for resource libraries for fonts, page segments, and overlays. Each of these resource types can contain a 240-pel JCL DD statement, a 300-pel JCL DD statement, a default JCL DD statement, or all three statements.

When a printer is manually switched from one resolution to another, PSF automatically starts using the resource libraries specified for the new printer resolution.

Printers operating in switchable resolution mode report to PSF the resolution that they are currently switched to. If a printer is switched to 300-pel resolution, it reports an IPDS resolution of 300-pel to PSF. PSF uses this resolution when it selects resource libraries for this printer.

For more information about how to define these JCL DD statements, see "Specifying PRINTDEV parameters" on page 102.

For an example startup procedure set up for multiple resolution libraries, see "Sample PSF startup procedure for four printers" on page 137.

Printers with automatic or auto-resolution mode

PSF now supports printers that operate in automatic or auto-resolution mode, unless they are connected through a distributed print function (DPF) product or RPM3.

Printers with automatic mode print at a particular pel density (for example, 600-pels per inch), but accept resources at different resolutions, such as 240-pels and 300-pels per inch. The printer then scales the resources appropriately based on their resolution and metric technology.

Printers operating in the auto-resolution mode report to PSF the resolution of the print head (for example, 600-pel). However, they also tell PSF that they accept other resolutions.

To use this feature, the PSF printer definition must be set up with access to all the different resolution resource libraries for fonts, overlays, and page segments. Additionally, PSF must receive an indication of the intended resolution when the print data was formatted. If no intended resolution is given, PSF uses the default JCL DD statement for that resource type.

For more information about how to define these JCL DD statements, see "Specifying PRINTDEV parameters" on page 102.

For an example startup procedure that is set up for multiple resolution libraries, see "Sample PSF startup procedure for four printers" on page 137.

For more information about format resolution, see "Format resolution specified from the application" on page 216.

Resource libraries segmented for multiple resolution

Resource libraries are used during formatting of print data and during the printing of the data. It might be important that the resource libraries that are specified to the print data formatting application match the resource libraries that are specified to PSF when the data is being printed. Failure to match the libraries might result in skewing of column alignment and of line endings when the data is printed.

The following AFP resource types are affected by print resolution:

Fonts
Raster fonts *must* match the printer resolution, unless your printer has automatic resolution mode. Even with automatic mode, it is preferable to use a font that matches the resolution at which the document was formatted. Outline fonts can print at any resolution, but it might be preferable to use a raster font if the document was formatted with raster fonts.

Overlays
Overlays contain text that is formatted with fonts and images that are created at specific resolutions. For this reason, an overlay might print better at a specific printer resolution.

Page segments
Most printers can scale page segments to print at the resolution of the printer. However, the image in a page segment prints best at the resolution in which it was created.

Some enterprises might already have printers with different print resolutions attached to PSF. Multiple JCL DD statements might be present in the startup procedure that define particular sets of resources based on resolution. For example:

```
//FONT01  DD DSN=SYS1.FONT3800            /* 3800 font library    */
//*
//FONT02  DD DSN=SYS1.FONTLIBB            /* 240 pel raster fonts */
//       DD DSN=SYS1.SFNTILIB,DISP=SHR   /* System outline fonts */
//*
//FONT03  DD DSN=SYS1.FONT300             /* 300 pel raster fonts */
//       DD DSN=SYS1.SFNTILIB,DISP=SHR   /* System outline fonts */
```

In this case, the FONT02 font library is used when formatting data for and printing data on a 240-pel resolution printer. Likewise, the FONT03 font library is used for a 300-pel resolution printer.

Notice that a JCL DD statement for the same outline font data set appears in the font library for 240-pel resolution fonts and for 300-pel resolution fonts. This is because the outline fonts are resolution independent and can be used by a printer (that supports outline fonts) without regard for the resolution of that printer.

To use the multiple resolution capabilities of PSF, it is necessary to separate font libraries into separate font libraries (JCL DD statements): one for the 240-pel raster fonts and another for the 300-pel raster fonts. Outline fonts can be specified in both libraries.

The same is true for overlay and page segment libraries. If overlays and page segments exist that print better on 240-pel printers, separate them into a 240-pel overlay or page segment library. Similarly, if some overlays and page segments print better on 300-pel printers, separate them into a 300-pel overlay or page segment library.

By separating the resource libraries by resolution, you can define them to PSF through the PRINTDEV JCL statement that indicates the resolution of the resources in each particular library.

For more information about how to define these JCL DD statements, see "Specifying PRINTDEV parameters" on page 102.

For an example of a startup procedure that is set up for multiple resolution libraries, see "Sample PSF startup procedure for four printers" on page 137.

Format resolution specified from the application

"Multiple resolution printers" on page 214 explains that multiple resolution printers can scale print data that is formatted for a particular resolution printer to another resolution if the printer understands the intended resolution for the print data. To do this, the print data must be tagged with the format resolution. The format resolution can be specified either in the JCL for the print data or directly in the print data stream.

Note also, a companion parameter (Font Fidelity) exists that PSF uses if a mismatch occurs between the resolution specified in the application and the resolution of the resources.

Specifying format resolution in JCL

The RESFMT JCL keyword on the OUTPUT statement indicates which resolution was used to format the job. For more information about the RESFMT keyword, see the *PSF for z/OS: User's Guide*.

Specifying format resolution in the print data stream

Applications that generate Map Coded Font Format 2 (MCF2) structured fields (such as DCF or OGL) can now add a font resolution and metric technology triplet (X'84') in the MCF2. This triplet tells PSF what resolution was used to format this print data. For more information about the triplet, see the *Mixed Object Document Content Architecture Reference*. For more information about the support of the triplet for your application, see the application's documentation.

Specifying font fidelity in the print data stream

MO:DCA-P defined a Font Fidelity triplet (X'78') in the Presentation Fidelity Control (PFC) structured field in the form definition. The scope of the PFC structured field is the document that is controlled by the form definition that contains the structured field.

The Font Fidelity triplet is used to specify the exception continuation rules for the font exceptions that are generated when the font that is referenced in the data stream is found but is not available to PSF at the specified resolution.

Note: The Font Fidelity triplet is not used with TrueType and OpenType fonts.

The triplet has two settings:

- Stop presentation at the point of the first font resolution exception and report the exception.
- Do not stop presentation because of font resolution exceptions.

PSF uses the Font Fidelity triplet to control two things:

- Whether the specified format resolution must match the actual font resolution.
- Whether it is acceptable to activate a resident outline font for a raster font.

If the Font Fidelity setting is set to STOP and the format resolution is specified, PSF checks to make sure the format resolution and the actual resolution of the font match. If they do not match, PSF stops processing the page and reports the exception.

If the Font Fidelity setting is set to DO NOT STOP, PSF does not require that the specified format resolution and the actual font resolution match. PSF also allows a resident outline font to be activated for a raster font.

If it is critical that your document prints with the same resolution resources that were used when it was formatted, specify STOP for the Font Fidelity in the form definition.

If it is acceptable to print the document at any resolution, either specify DO NOT STOP for the Font Fidelity in the form definition or do not specify a Font Fidelity because the default is DO NOT STOP.

Specifying format resolution

To override the format resolution that is specified in the data stream and on the user's JCL, you can specify the format resolution in several ways:

- The **Resolution** parameter in the Printer Inventory or XTP7FMT during the Exit 7 initialization (INIT) call can be set to 240-pels or 300-pels per inch resolution. This setting establishes a PSF default format resolution for all jobs and can be overridden only by the Exit 7 begin-data-set call (BDSC) or end-data-set call (EDSC).
- XTP7FMT in Exit 7 can be set during the BDSC or EDSC to 240-pels or 300-pels per inch resolution for any or all print data sets. You can use this setting to override the resolution on a data set by data set basis, which might be useful when you combine two job streams, one that is 240-pels and another that is 300-pels, and you are printing to a printer that accepts data streams at both resolutions. If each job stream has a unique characteristic, such as output class, Exit 7 can set the XTP7FMT value based on the characteristic for each data set.

Format resolution order

Format resolution can be specified in numerous places, including data stream and JCL. The following items show the order in which PSF looks for format resolution:

1. If the printer is NOT in automatic mode, which means it is accepting resources only at a single resolution, PSF uses the resolution reported to PSF by the printer to select the resolution for the resource libraries.

2. If the printer is in automatic mode, which means it can accept resources at multiple resolutions, PSF looks for the first specification of the format resolution in the following order:

   a. The PSF installation exit for Resource Management (Exit 7) begin-data-set call (BDSC) or end-data-set call (EDSC).

   b. The Printer Inventory or the PSF installation exit for Resource Management (Exit 7) initialization call.

   c. In the data stream in the MCF2 structured fields in the Font Resolution and Metric Technology triplet (X'84').

   d. In the OUTPUT JCL statement RESFMT keyword.

PSF uses the determined format resolution to choose the system libraries to use when it looks for resources. If inline resources, user libraries, or both are specified, PSF searches them first.

Font Fidelity controls the disposition of any errors in which the font found by PSF does not match the specified format resolution.

Chapter 18. Understanding PSF data protection and security

PSF and the printers it supports protect data in the following ways:

- Data cannot be read back into the host processor by using the available printer commands.
- Characters that are stored in the print buffer cannot be easily interpreted because they are translated from EBCDIC form.
- All raster character patterns that are stored in the printer (excluding the permanently resident fonts) can be erased. PSF can erase data that is remaining in the data buffers and in the accumulator, including page segments and electronic overlay data. PSF erases this data when a printer is started or stopped.
- No pages are left in the hardware buffer. PSF makes sure that all pages are in the stacker before it responds to a STOP WRITER command. When a printer that is using continuous forms is in direct-printing mode, the nonprocess runout (NPRO) parameter must be specified to ensure that all pages are stacked. If NPRO is not specified, the operator must do the NPRO.
- When a form change is requested, PSF provides a software NPRO to make sure that all printed pages are in the stacker before it allows the forms to be changed. For example, when a user specifies SYSOUT=(*class*[,, *formname*]) for printing on a continuous-forms printer, PSF causes the printer to do an NPRO on the last page of one form before JES issues the message that tells the operator to load the new form. This is done as a data security measure to make sure that all the pages of the first data set are successfully printed before the operator loads a new form. Because of this software NPRO, more blank pages can be printed to move the previously printed pages into the stacker.
- You have responsibility for specific protection for data sets in a shared environment, such as distributed print function (DPF). Sensitive material that is stored in the PSF resource library must be marked PRIVATE. For more information, see "Using APSRMARK to mark resources" on page 245.
- Resources that are stored in libraries that are defined in the PSF startup procedure are available to all print jobs. These libraries must not be used for sensitive resources or be protected by read passwords. Consider using RACF-controlled user libraries to manage sensitive resources. For more information, see "User libraries in deferred-printing mode" on page 51.
- PSF uses inline resources only for the data set with which they are sent inline; PSF ensures that the resources will not be used again after the processing of that data set is completed. PSF also tells the printer not to capture resources that are found inline or in user libraries. If the printer is attached through DPF or WPM, the Printer Inventory or Exit 7 can specify that inline resources are captured. See Capture inline resources for more information.

For specific printers, see the publications for those printers.

Security labeling

Consider using security labeling to protect printed output. Security labeling can be made either voluntary or mandatory. For more information, see *PSF for z/OS: Security Guide*.

Controlling print complete notification

PSF can notify as many as four users when a print job is complete. To request that such a message is sent, use the NOTIFY JCL keyword in the OUTPUT statement. To read more about specifying the NOTIFY keyword, see *PSF for z/OS: User's Guide*.

To control these notices, profiles must be created in the SMESSAGE class, and the users who are to send notification messages must be given authority for access to the profiles. To use the security that is provided by these profiles, activate both the SMESSAGE class and the SETROPTS RACLIST processing for

the SMESSAGE class. For more detailed information, see the *z/OS Security Server RACF Security Administrator's Guide*.

Chapter 19. Supporting resident fonts

Resident fonts are font resources that reside in a printer. Their use can significantly decrease the amount of data that PSF must send over slow-speed communication lines or congested LANs.

Not all printers support resident fonts, and of the ones that do support them, some support more than others. For information about the resident fonts your printer supports, see the documentation provided with the printer.

Resident fonts for different printers are put on different media. For most LAN-attached printers, resident fonts are included internal to the printer, installed and ready to use. For information about resident fonts for a particular printer, see the publications for that printer.

Figure 68 on page 221 is a diagram of a printer that supports resident fonts.
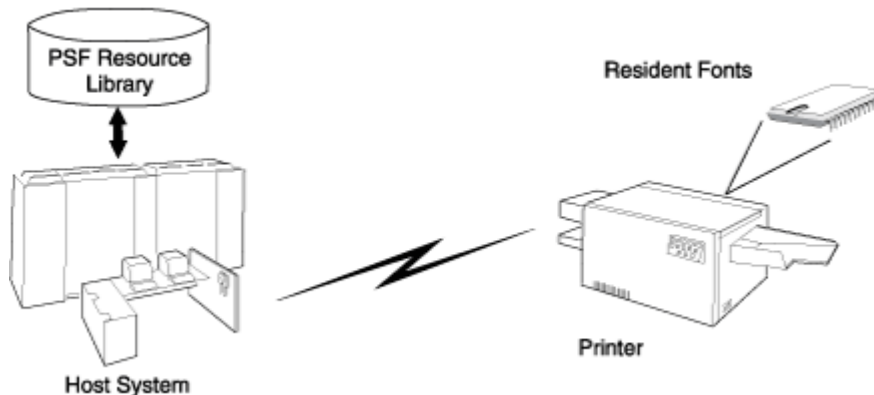
Figure 68. A printer that supports resident fonts

Notes:

1. The host version and the printer-resident version of a font cannot be identical, especially if your installation edited either version of the font. Also, printer-resident fonts and fonts that are stored in other libraries can differ, even though the fonts have the same name.

2. For printers attached through a workstation print manager (WPM) or a distributed print function (DPF), PSF does not support printer-resident fonts, even though the printer itself can support them. However, PSF direct does support printer-resident fonts.

3. Some printers, such as the 3130 printer and InfoPrint 3000 and 4000, can capture downloaded fonts. For more information, see Chapter 20, "Marking fonts for font capture," on page 231.

PSF provides two methods to access resident fonts:

- APSRFTBL (hardcoded table), used to identify resident fonts in printers that support only symbol sets—for example, the 4224, 4230, 4234, 4247, 6408, and 6412 printers.
- APSRMARK (utility), used for all other printers that support resident fonts.

PSF resident font table (APSRFTBL)

PSF uses the APSRFTBL module, which is supplied with PSF in the SYS1.SAMPLIB data set, to identify fonts resident in printers that support symbol sets, such as the 4224, 4230, 4234, 4247, 6408, and 6412 printers. PSF does not download fonts to these printers. APSRFTBL contains tables that associate the font character set and code page for a particular host font with the identifier of the resident symbol set. APSRFTBL does not contain code that can be run. You can change, assemble, and install it (or link-edit it with the **replace** option) to provide your own version of this module if you want to change the resident symbol sets to be used for particular code pages and character sets.

The sample APSRFTBL module is found in Appendix E, "FOCA font resource tables supplied by PSF," on page 411.

Changing APSRFTBL

You can add, delete, and change any entries in the APSRFTBL font resource table except the last line, which contains the characters 'FFFFFFFF' and indicates the end of that particular table; all code for the table must be placed before the 'FFFFFFFF' line.

Table 26 on page 222 shows the character set and code page tables that you can update, depending on the printer you are using or emulating.

Table 26. Tables to update in the APSRFTBL module based on printers

Printer[10]	Character Set Table	Code Page Table
4224	APSCS24V	APSCP24V
4234	APSCS34V	APSCP34V
4247	APSCS47V	APSCP47V
64xx (6400, 6408, 6412)	APSCS64X	APSCP64X
65xx (6408, 6412)	APSCS64X	APSCP64X

If the APSRFTBL resident-font tables do not contain an entry for a font requested in a print job, PSF issues an error message. Processing of the print job stops on the page where the font was requested, but continues on the succeeding pages. To prevent this error, you can edit the resident-font tables in APSRFTBL and associate the font character set or code page of any font referenced in a print job with the identifier of a resident font.

For example, suppose that a user specified Prestige 10-point font (character set C0S0PR10) for a print job that is sent to a 4224 printer, but that font is not included in APSCS24V, the character set table in the APSRFTBL module for 4224 font character sets. You can decide to map that font to the Gothic 10-point symbol set that is resident in the 4224 printer.

Note: For a list of resident symbol sets, see the publications for your printer.

To change the APSCS24V table in the APSRFTBL module for a 4224 printer:

1. Duplicate the entry line for the Gothic 10-point font.
2. Change the character set name in the duplicated line to the character set name of the font specified in the print job, as in this example:

```
DC      CL8'C0D0GT10',XL2'001A',XL2'0000'
DC      CL8'C0S0PR10',XL2'001A',XL2'0000'
```

3. Because the resident-font table is searched sequentially, put the fonts used most often at the beginning of the table.

   Note: Do not change or delete the last entry in the tables; the entry CL8'FFFFFFFF' marks the end of each table. Do not change the table labels.

4. After you change the table, assemble the APSRFTBL module and install (link-edit) APSRFTBL into the applicable LINKLIB.

---

[10] If a printer is run in emulation mode, the table for the emulation mode printer is used. For example, if the 4230 printer is run in 4224 emulation mode, you update the APSCS24V and APSCP24V tables.

PSF resident font utility—APSRMARK

Host-font resources—code pages and character sets—can reside in a printer. To identify such resources to PSF as potentially printer-resident, use the APSRMARK utility, which is provided by PSF.

APSRMARK is a batch program started with control statements. You can use it to do these:

- Mark all or individual host code pages and character sets of partitioned data sets, so that PSF knows that a printer-resident version of the resource exists.
- Report the marking status of all or individual members of partitioned data sets.
- Report the marking status of a font resource that is found in a sequential file that can later be moved into a font library.

Fonts that are updated by APSRMARK are called *marked* fonts. When you mark a font, you are essentially updating certain structured fields. For example, when you mark a code page, you are updating the Begin Code Page structured field.

To mark a font, you provide JCL statements to run the APSRMARK utility and identify fonts as either PUBLIC or PRIVATE. If the version of the font that is stored in the PSF resource libraries is marked PRIVATE, PSF downloads the font to the printer. A font that is stored in a PSF resource library must be marked PUBLIC so that the equivalent printer-resident font can be used; this prevents PSF from downloading the font to the printer. If a font is not marked, it is treated as if it were marked PRIVATE. To prevent unnecessary downloading of fonts, mark PUBLIC any font that has an equivalent printer-resident font. Some fonts in the AFP Font Collection, Infoprint Fonts, z/OS Font Collection, and 4028 Font Metrics are already marked PUBLIC. For more information about marking the fonts, see "Syntax of APSRMARK control statements" on page 224 and "APSRMARK control statement parameters for resident fonts" on page 225.

If someone updates a code page or character set on the host that was marked PUBLIC, you must run APSRMARK to change the marking of the host resources—for example, by entering new date and time values. Otherwise, the change is not reflected in the resources selected for printing a file. This is especially important when a resource is copied from a system library and modified for use in a private library. Unless you run APSRMARK on the modified resource, you might use the old resource by mistake.

Notes:

1. You cannot concatenate partitioned data sets as input to APSRMARK.
2. To identify 4224, 4230, 4234, 4247, 6408, or 6412 printer-resident font resources to PSF, see "PSF resident font table (APSRFTBL)" on page 221.
3. To read about the sample jobs that PSF supplies for marking resources, see Appendix C, "APSRMARK sample jobs," on page 399.
4. To use APSRMARK to identify resources for use by DPF, see Chapter 23, "Marking resources for distributed print function," on page 245.

   You can use APSRMARK to mark the same resources for DPF and for printer-resident use; the latter is described in "Using DPF fonts and printer-resident fonts with APSRMARK" on page 252. For printers that support resident outline fonts, you can use APSRMARK to mark host outline fonts. However, if the printer is supported through WPM or DPF, you cannot use outline fonts.

APSRMARK considerations

Before you mark fonts for resident use, consider these:

- Font character sets containing only font metrics are included with PSF as an aid in formatting documents that call for 4028 printer-resident fonts, for which PSF has no host equivalents. Because a 4028 metric-only font contains no raster data, PSF must use the printer-resident font that has the same name. Therefore, the 4028 metric-only fonts that are included with PSF are marked by use of the APSW4028 sample APSRMARK job. Code pages included with PSF are also marked PUBLIC.

  Do not mark metric-only font character sets PRIVATE.

- Some printers do not support all combinations of inline direction and character rotation for resident fonts. If users require combinations that are not available for a resident font, mark the host version of the affected font PRIVATE, so that PSF uses it instead of the printer-resident font. Fonts marked PRIVATE are sent to the printer as needed. For more information about the print directions and character rotations the printer supports, see the documentation provided with the printer.

- You can use APSRMARK to mark resources that can be used as inline resources or to mark user library resources. However, PSF tells the printer not to capture resources found inline or in user libraries. If you want the resource made resident, you need to install it directly on the printer or put it in a system library so that it can be captured. If the printer you are using is attached through DPF or WPM, you can use the Printer Inventory or Exit 7 to specify that inline resources are captured. See Capture inline resources for more information.

- When you are using resident fonts, you might see some of the following differences:
  - Different character raster patterns
  - Different font metrics, such as character positioning
  - Different characters included in the character set for the font

  If any of the differences occur, consider a couple of alternatives. One is to mark any altered fonts that are residing in the PSF resource library as PRIVATE so that the host version of the font is used instead of the printer-resident version (which is not altered). The other is to provide a host version of the font that is identical to the printer-resident font that is used.

- You can use a single font library for multiple 240-pel IPDS printers, even if different fonts are resident in each printer. You can mark the fonts resident in any printer as PUBLIC, and PSF queries the printer to determine whether the resident font is available. To determine which fonts your printer supports, see the documentation provided with the printer.

  Note: Do not use a 240-pel library for a 300-pel printer, even if the resident fonts in the 300-pel printer are equivalent (same names) to those fonts in the 240-pel library. Instead, create and mark a 300-pel version. For more information, see "Using APSRMARK" on page 224. The single 300-pel font library can be used for multiple 300-pel IPDS printers. Mark the fonts resident in any printer as PUBLIC.

Using APSRMARK

Before PSF can use a font that is resident in a particular printer, you must complete the following procedure:

1. Choose the printer-resident fonts that you want to use. For a list of the fonts resident on a printer, see the documentation provided with the printer.

2. Determine the member names for both the code page and the font character set of the corresponding font in the system library.

   Note: For 300-pel (300-dpi) printers such as the 3112, 3116, 3916, 3935, and 4028 printers, run the APSRCF30 utility to convert the 240-pel fixed fonts to 300-pel relative metric fonts before you use APSRMARK. APSRCF30 is a font-scaling utility; it is described in Appendix I, "Font conversion program, 240-pel to 300-pel (APSRCF30)," on page 457.

3. Use APSRMARK to mark the chosen code page and font character set. For more information about marking the fonts, see "Syntax of APSRMARK control statements" on page 224 and "APSRMARK control statement parameters for resident fonts" on page 225.

4. Use the member names that are marked in Step "3" on page 224 to make sure that the application that is generating the print job correctly references the font.

5. Make sure that the marked member is the first member with that name found by PSF, which selects resources according to hierarchy. See "Searching for resources specified by a print job" on page 179.

Syntax of APSRMARK control statements

The control statements that provide input to APSRMARK must be preceded by the SYSIN DD statement, and must use the following syntax:

- Each statement must be an 80-column record. The first 71 columns can contain valid APSRMARK parameters. APSRMARK ignores columns 73–80, which can contain a sequence number. A statement that does not fit on one line can be continued on the next line by entering a continuation character (any nonblank character) in column 72.
- Blank spaces are optional; they can be placed between parameters for readability.
- An asterisk (*) in column 1 denotes a comment statement.

If any syntax or I/O errors are found in the control statement parameters, APSRMARK reports them.

APSRMARK control statement parameters for resident fonts

To mark a host font to indicate that a printer-resident font must be used in its place, if one is available, use the following control statements:

RRID
Specifies a resident resource identifier of 8 hexadecimal digits.

Note: Sample jobs are included with PSF; see Appendix C, "APSRMARK sample jobs," on page 399. For specific values of RRID, use the following formulas:

- For the font character set:

```
RRID = FGID + FW
```

where:

```
     FGID = a font identifier of 4 hexadecimal digits
     +    = concatenation
     FW   = a font width of 4 hexadecimal digits
```

When RRID is specified, both FGID and FW must be specified, even if one or both fields contain only zeros or leading zeros.

- For the code page:

```
RRID = GCSGID + CPGID
```

where:

```
     GCSGID = a graphic character set identifier
              of 4 hexadecimal digits
     +      = concatenation
     CPGID  = a code page identifier
              of 4 hexadecimal digits
```

When RRID is specified, both GCSGID and CPGID must be specified, even if one or both fields contain only zeros or leading zeros.

If RRID is not specified, the FGID and FW values from inside the font character set or the GCSGID and CPGID values from inside the code page are used.

RRDATE
Consists of any value of 5, 6, or 7 decimal digits.

```
RRDATE = yyyyddd
```

where:

```
     yyyy = the year in 2, 3, or 4 digits. For example:
            89   =  1989
            02   =  1902
            002  =  2002
            1998 =  1998
            2004 =  2004
     ddd  = the 3 digit Julian date.
```

Note: For a 3820 printer, this value is defined in the printer publications. For all other printers with resident fonts, this value can specify the date on which the font was marked.

RRTIME
Consists of any value of 8 decimal digits.

```
RRTIME = hhmmsshh
```

where:

```
        hh = Hour
        mm = Minutes
        ss = Seconds
        hh = Hundredth of a second
```

Note: For a 3820 printer, this value is defined in the printer publications. For all other printers with resident fonts, this value can specify the time at which the font was marked.

{PUBLIC|PRIVATE}
Must be specified for both the code page and the character set. If a resource is marked PRIVATE, it is sent from the host to the printer when needed. If a resource is marked PUBLIC, and an equivalent resource is resident in the printer, PSF does not send the host resource from the host to the printer; PSF uses the printer-resident resource. shows an example of marking resources.

Do not mark metric-only font character sets PRIVATE.

Using control statements to mark a font

shows an example of how you can use JCL control statements to mark a font. In this example, SYS1.FONTLIBB (T1D0BASE) contains the code page, and SYS1.FONTLIBB (C0D0GT10) contains the character set. This JCL copies members T1D0BASE and C0D0GT10 from SYS1.FONTLIBB to SYS1.FONT3812. The marked copies are in SYS1.FONT3812.

```
//EX3812 JOB  (X,YY),MSGLEVEL=1,MSGCLASS=A,USER=SAM,PASSWORD=TODAY
//STEP1  EXEC  PGM=APSRMARK
//*
//SYSPRINT DD SYSOUT=A
//IN1    DD UNIT=SYSDA,DSN=SYS1.FONTLIBB,DISP=SHR,VOL=SER=USR000
//OUT1   DD UNIT=SYSDA,DSN=SYS1.FONT3812,DISP=SHR,VOL=SER=USR000
//*
//SYSIN  DD *
        INDD=IN1,OUTDD=OUT1,MEMBER=T1D0BASE,PUBLIC,RRID=000003EA,RRDATE=88044,RRTIME=00000000
        INDD=IN1,OUTDD=OUT1,MEMBER=C0D0GT10,PUBLIC,RRID=00280090,RRDATE=88044,RRTIME=00000000
/*
```

Note: Your syntax might differ; for syntax rules, see .

Figure 69. Sample JCL for marking resident fonts

APSRMARK control statement parameters (general)

INDD=*DDname*
Specifies the name of the input data set to be used when you mark or report resources. The INDD parameter can name either a sequential data set or a partitioned data set. This parameter is required.

MEMBER=*name*|ALL
This parameter is required for partitioned data sets. Do not specify this parameter for sequential data sets.

name
Specifies that resource marking or resource reporting is to be done for a single member (primary or alias) of a partitioned data set. When resources are marked, both the primary member and its alias are marked unless the primary is updated or deleted since the alias is generated; in that

case, the alias is not updated. An alias whose primary is updated or deleted can be marked only by specifying the name of the alias and specifying the NEWNAME keyword.

ALL
Specifies that resource marking or resource reporting is to be done for all members of a partitioned data set. When resources are marked, all primary members and aliases whose primaries are not updated or deleted are marked. Aliases whose primaries are updated or deleted are not marked. When resources are reported, only primary members are listed. If you specify MEMBER=ALL when you are marking multiple fonts for printer-resident use or printer capture, the RRID value is used from inside the font character set or the code page. You can also specify MEMBER=ALL with the REPORT parameter.

NEWNAME=*name*
Specifies a new name for a member of a partitioned data set. This parameter is optional and has meaning only when a specific member is specified by MEMBER=*name*. The newly named member is marked and becomes a primary member without an associated alias.

OUTDD=*DDname*
Specifies the DD name of the output data set in which the marked resources are to be placed. You can use the REPLACE parameter to mark resources (or change resource marking) and replace them in the same data set. This parameter is required.

REPLACE
Specifies that existing members of the specified output partitioned data set are to be overwritten (replaced). REPLACE must be specified to overwrite members. Use REPLACE to mark resources initially, and then replace them in the same data set. REPLACE is also used to change resource markings. This parameter is optional, and applies only to partitioned data sets.

REPORT
Specifies that a report is to be formatted and printed. The report lists how resources are marked. The INDD and MEMBER parameters determine which resources are reported.

- INDD specifies the name of the input data set.
- MEMBER=ALL specifies that resource reporting is to be done on all primary members of a partitioned data set.
- MEMBER=*name* specifies that resource reporting is to be done on a single primary or alias member of a partitioned data set.

SHORT|LONG
Specifies which report is formatted and printed when the REPORT parameter is specified:

- SHORT specifies that a short report is generated. See Figure 72 on page 229 for an example of a short report.
- LONG specifies that a long report is generated. This is the default report format. See Appendix D, "APSRMARK long report format," on page 401 for an example of a long report.

Sample control statements for APSRMARK

In the following examples, lowercase letters represent information that you specify to customize the JCL.

Marking a member with resident font information

Figure 70 on page 228 shows how to mark a member of a partitioned data set.

```
//example1 JOB  (X,YY),MSGLEVEL=1,MSGCLASS=A,USER=userid,PASSWORD=password
//*
//STEP1  EXEC  PGM=APSRMARK
//*
//SYSPRINT DD SYSOUT=class
//IN1    DD UNIT=dasd,DSN=dsname1,DISP=SHR,VOL=SER=volser
//OUT1   DD UNIT=dasd,DSN=dsname2,DISP=SHR,VOL=SER=volser
//SYSIN  DD *
     INDD=IN1,OUTDD=OUT1,MEMBER=codepage,PUBLIC,RRID=rrid,RRDATE=rrdate,RRTIME=rrtime
     INDD=IN1,OUTDD=OUT1,MEMBER=charset,PUBLIC,RRID=rrid,RRDATE=rrdate,RRTIME=rrtime
/*
```

Note: Your syntax might differ; for syntax rules, see "Syntax of APSRMARK control statements" on page 224.

Figure 70. Example of the use of JCL statements to mark a member of a data set

Figure 70 on page 228 contains the following JCL statements:

IN1 DD
Defines the input partitioned data set that contains the member to be marked by APSRMARK.

OUT1 DD
Defines the output partitioned data set that contains the member marked by APSRMARK.

Note: The data set attributes for OUT1 must match the attributes for IN1. The data set name for OUT1 can match the data set name for IN1.

SYSIN DD *
Contains the marked member name (resource name) and the three specific values for the RRID, RRDATE, and RRTIME parameters.

Generating a short report of resident font marking

The short report format lists resources but does not indicate whether a resource is correctly enabled for capture or for resident resource activation. The short report also contains question marks (?) to indicate that data was not present in a resource.

Figure 71 on page 228 shows how to use the SHORT parameter to generate a short report.

```
//EXAMPLE2 JOB  ,'IBM USER',MSGLEVEL=A
//STEP1    EXEC PSG=APSRMARK
//SYSPRINT DD   SYSOUT=A
//THISDD   DD   DSN=NEW.FONTLIB,DISP=SHR
//SYSIN    DD   *
     INDD=THISDD,MEMBER=ALL,REPORT,SHORT
/*
```

Note: Your syntax might differ; for syntax rules, see "Syntax of APSRMARK control statements" on page 224.

Figure 71. Example of the use of JCL statements in generating a short report

Figure 71 on page 228 contains the following JCL statements:

THISDD DD
Defines the partitioned data set that contains the member that is reported.

SYSIN DD
Precedes the control statements that specify the parameters for the report to be printed.

Figure 72 on page 229 is an example of a printed short report.

```
 APS580I      INDD=OUT2,MEMBER=ALL,REPORT
 APS580I  CZB50C  PUBLIC  SAVE   2004-02-29 14:32:56 9763 661200354381 PSFDVT
L1111111.L2222222.L3333333.L4444444.L5555555
 APS580I  C0MYNEW1 PUBLIC  SAVE   2004-02-29 14:32:56 9763 661200354381 PSFDVT
L1111111.L2222222.L3333333.L4444444.L5555555
 APS580I  +00280090  77060 12121212
 APS580I  C0FACTRY PUBLIC  SAVE                       PSFDVT
L1111111.L2222222.L3333333.L4444444.L5555555
 APS580I  C0NOMARK ??????? ?????? ????-??-?? ??:??:?? ???? ??????????? ?????? ?
 APS580I  F1NOBOX  ??????? ?????? ????-??-?? ??:??:?? ???? ??????????? ?????? ?
 APS580I  M1UBLPK  ??????? ?????? ????-??-?? ??:??:?? ???? ??????????? ??????? ?
 APS580I  O1ROI000 PUBLIC  SAVE   2004-02-29 14:32:56 CBE4 661200354381 PSFFDVT
LISZ.MKDRCS
 APS580I  P1SEC5   ??????? ?????? ????-??-?? ??:??:?? ???? ??????????? ?????? ?
 APS580I  S1SCHLAF PUBLIC  SAVE   2004-02-29 14:32:56 47A5 661200354381 PSFDVT
DOSCHE.PRIVATE.PSEGMENT.RESOURCE
 APS580I  T1L000XN PUBLIC  SAVE   2004-02-29 14:32:56 A335 661200354381 PSFDVT
LISZ.MKDRCS
 APS580I  T1MYNEW1 PUBLIC  SAVE   2004-02-29 14:32:56 A335 661200354381 PSFDVT
LISZ.MKDRCS
 APS580I  +01330129  77060 12121212
 APS580I  T1BASIC1 PUBLIC  SAVE                       PSFDVT
LISZ.MKDRCS
 APS580I  XZH0FN   ??????? ?????? ????-??-?? ??:??:?? ???? ??????????? ??????? ?
```

Figure 72. Printed output of printer-resident short report

Generating a long report of resident font marking

The long report format differs from the short report format in these ways:

- All data is identified and all unmarked resources are obvious.
- All data that affects resource capture is listed.
- All data that affects resident resource activation is listed.
- A statement indicates whether a specific resource can be captured, used for resident resource activation, or both.

Figure 73 on page 229 shows how to use the LONG parameter, which is the default, to generate a long report.

```
//EXAMPLE3 JOB  ,'IBM USER',MSGLEVEL=A
//STEP1    EXEC PSG=APSRMARK
//SYSPRINT DD   SYSOUT=A
//THISDD   DD   DSN=NEW.FONTLIB,DISP=SHR
//SYSIN    DD   *
     INDD=THISDD,MEMBER=ALL,REPORT,LONG
/*
```

Note: Your syntax might differ; for syntax rules, see "Syntax of APSRMARK control statements" on page 224.

Figure 73. Example of the use of JCL statements in generating a long report

Figure 73 on page 229 contains the following JCL statements:

THISDD DD
Defines the partitioned data set that contains the member that is reported.

SYSIN DD
Precedes the control statements that specify the parameters for the report to be printed.

See Appendix D, "APSRMARK long report format," on page 401 for a description of a long report format.

Diagnosing problems with printer-resident fonts

Table 27 on page 230 summarizes some of the problems that result from incorrect use of APSRMARK, and provides possible solutions.

Table 27. Resident font diagnosis	
Problem	**Possible solution**
PSF loads the host library version of the font into the printer and does not use the printer-resident font.	The code page, the font character set, or both are not marked correctly. See "Using APSRMARK" on page 224. If the resources are marked correctly, make sure that your MO:DCA-P document (or your page definition) is using the names of the resources that are marked as resident. Make sure that the marked resource is found by PSF before any unmarked version of the resource. For more information, see "Searching for resources specified by a print job" on page 179.
A printer-resident font is used, but not the one specified.	Some printer-resident fonts are simulated by the printer and might appear different from the version in the system library. For more information about font simulation that is done by your printer, see the publications for your printer. If a simulation is not the problem, make sure that the correct RRID values are specified. For more information about RRID values, see "APSRMARK control statement parameters for resident fonts" on page 225.
Printing results in messages APS818I and APS717I, where the sense bytes 0, 1, and 19 of message APS717I contain the values X'02', X'18', and X'02', respectively.	You might specify a character rotation or print direction that your printer does not support. For information about the character rotations and print directions your printer supports, see the documentation provided with the printer.
Printing results in message APS279I.	Font character sets containing only font metrics are provided as an aid in formatting documents that reference certain printer-resident fonts that have no equivalent downloadable host font. These metric-only font character sets must be marked by the APSRMARK utility with the PUBLIC keyword specified so that PSF can access the printer-resident font. Make sure that the font that is identified in the message is marked correctly. **Note:** APSW4028 is a sample APSRMARK job that contains control statements to correctly mark 4028 Font Metrics character sets, which can be resident in printers such as the 3116, 3916, and 4028 printers. If the 4028 Font Metrics character set is marked correctly, make sure that the printer contains the resident font that you want.
Printing results in message APS280I.	If your installation is using 240-pel fonts to access 300-pel resident fonts, make sure that the font cartridge that is installed in the printer contains the 300-pel resident font that you are trying to access with a marked 240-pel resident font. If the correct 300-pel resident font does not exist in the printer, PSF attempts to download the host font. APS280I is issued when PSF attempts to download a 240-pel host font to a 300-pel printer. Make sure that the host font library is a 300-pel version (contains 300-pel character sets). For more information about creating and marking a 300-pel font library, see "Using APSRMARK" on page 224.

Chapter 20. Marking fonts for font capture

Some printers, such as the 3130, InfoPrint 3000, and InfoPrint 4000 printers, can capture downloaded fonts. The concept is similar to *dynamic caching* in DPF. For more information about DPF dynamic caching, see Chapter 23, "Marking resources for distributed print function," on page 245.

Both raster fonts and outline fonts can be captured. Captured fonts automatically become new temporary printer-resident fonts. This improves performance on future jobs that use the same fonts. If the fonts are ever deleted from the printer, PSF automatically downloads the host version the next time it is needed. The printer manages the captured font library.

A font character set or code page can be captured by a printer only if it is marked PUBLIC, has a date and time stamp for resident fonts, and is loaded from a system library. If the printer you are using is attached through DPF or WPM, you can use the Printer Inventory or Exit 7 to specify that inline resources are captured. See Capture inline resources for more information. Fonts marked PRIVATE are always downloaded, never captured. You can use the APSRMARK utility to mark a resource.

Single-byte raster font capture

For raster fonts that were previously included as a part of the PSF product, run the sample APSRMARK jobs that were included with PSF. For more information, see Appendix C, "APSRMARK sample jobs," on page 399.

The single-byte raster fonts included in the AFP Font Collection (Program Number 5648-B33) and z/OS Font Collection (a base element of z/OS V2R1 or later, Program Number 5650-ZOS) are marked correctly for activating resident fonts. To capture fonts, you must run APSRMARK with the same syntax used to mark fonts for activating printer resident fonts. See "PSF resident font utility—APSRMARK" on page 223 for more information.

For single-byte raster fonts that you create or change, run APSRMARK to mark them PUBLIC, and specify the GRID part in RRID and the new date and time in RRDATE and RRTIME.

When a font is not to be captured, mark it PRIVATE. For example, if a copyright restriction exists against having a copy of a font on the printer hard disk drive, mark the font PRIVATE.

Single-byte outline font capture

All single-byte outline fonts included in AFP Font Collection (Program Number 5648-B33), Infoprint Fonts for z/OS (Program Number 5648-E76), and z/OS Font Collection (a base feature of z/OS, Program Number 5650-ZOS) are marked correctly for activating resident fonts and capturing fonts.

Any single-byte outline fonts created or changed by the IBM Type Transformer program might need to be marked PUBLIC, but they contain all the other information needed for font capture. You do not need to specify RRID, RRDATE, or RRTIME when you run APSRMARK. For information, see "PSF resident font utility—APSRMARK" on page 223.

When a font should not be captured, mark it PRIVATE. For example, if there is a copyright restriction against having a copy of a font on the printer hard disk drive, mark the font PRIVATE.

Double-byte raster font capture

The double-byte raster fonts included in the AFP Font Collection (Program Number 5648–B33) and z/OS Font Collection (a base element of z/OS V2R1 or later, Program Number 5650-ZOS) are marked correctly for activating resident fonts and capturing fonts. You do not need to run APSRMARK.

For new installations of DBCS font product IDs 5771-AGB, 5771-AEK, 5771-AFZ, 5771-AFW, or 5771-AEN, run the sample APSRMARK job included with the product.

For raster double-byte sections that you create or change, run APSRMARK to mark them PUBLIC, and specify the GRID part in RRID and the new date and time in RRDATE and RRTIME. For information about using APSRMARK, see "PSF resident font utility—APSRMARK" on page 223.

When a font should not be captured, mark it PRIVATE. For example, if there is a copyright restriction against having a copy of a font on the printer hard disk drive, mark the font PRIVATE.

Double-byte outline font capture

The double-byte fonts in *IBM AFP CJK Metric-Only Fonts,* RPQ 8A8080, are not suitable for capture, because they do not contain font data. However, they are correctly marked for activating the printer-resident version of the fonts. You do not need to run APSRMARK.

The double-byte outline fonts included in AFP Font Collection (Program Number 5648-B33), Infoprint Fonts for z/OS (Program Number 5648-E76), and z/OS Font Collection (a base feature of z/OS, Program Number 5650-ZOS) are marked correctly for activating resident fonts and capturing fonts.

When you use the IBM Type Transformer program to create downloadable DBCS outline fonts, you can specify CAPTURE=YES|NO. CAPTURE=YES corresponds to PUBLIC, and CAPTURE=NO to PRIVATE. You need to run APSRMARK only if you want to change the setting.

TrueType and OpenType font capture

TrueType and OpenType fonts are enabled for printer-resident activation and capture when you use the Font Installer for AFP Systems or WorldType Fonts for AFP Print Servers optional features of Infoprint Fonts for Multiplatforms (Program Number 5648–E77) or the z/OS Font Collection (a base element of z/OS V2R1 or later).

Chapter 21. Using FOCA outline fonts

This information explains why IBM supports printers that use FOCA outline fonts, how outline fonts and raster fonts differ, and how you can tailor PSF so that print-job submitters can use outline fonts.

IBM's font strategy

IBM's long-range strategy is for you to be able to print any kind of document on any printer in your enterprise. In a move toward fulfilling this strategy, IBM is supporting printers that can use outline font technology[11] because outline font technology is resolution-independent.

The AFP Font Collection (Program Number 5648-B33) and the z/OS Font Collection (a base feature of z/OS, Program Number 5650-ZOS) contain a selection of fonts in both outline and raster formats. Infoprint Fonts for z/OS (Program Number 5648-E76) contains a selection of outline fonts only. (For a list of publications describing IBM's fonts, see the "Bibliography" on page 535.) Say that a document for which an outline font is requested is sent to a printer on which the font libraries have been installed. If the printer supports outline fonts, the document is printed in the requested font. If the printer supports only raster fonts, the outline font is mapped to an equivalent raster font, and the document is printed in that raster font.

If a raster font is requested for a document, PSF handles the request in accordance with the specifications you make when you configure PSF. You can configure PSF to use the raster font or to map to an equivalent outline font. For information about how to use the Printer Inventory, the PRINTDEV statement, or Exit 7 to map to an equivalent outline font, see "Mapping to outline fonts" on page 236.

You can create fonts and add them to your font libraries by using the Type Transformer program, which you can obtain at no extra charge from IBM Type Transformer for Windows (www.ibm.com/support/docview.wss?uid=psd1P4000840). For information about creating new fonts, see *IBM AFP Fonts: Type Transformer User's Guide*.

Understanding outline fonts

For raster fonts, each character (bitmap) is made up of an arrangement of pels. Characters are stored in their full-size, bitmap forms, with the attendant characteristics of style, size, weight, height, width, and resolution.

For outline fonts, each character shape is represented by a mathematical expression, which, when combined with the font metrics, determines the style, shape, height, width, and space of the character. Outline fonts are stored in this format, along with vertical scaling information for determining the size of the characters, as well as information about the rotation of each character. When a printer that supports outline fonts needs to use font characters, it creates a rasterized bitmap for each character, at the printer's resolution, and rotates the characters as instructed.

Character sets and code pages for outline fonts are maintained as separate objects in the printer. Thus, outline font character sets can be shared among coded fonts. This allows an application that uses outline fonts to use less printer storage, because it can use a character set or a code page that is already available in the printer and scale it to a different size.

Outline fonts are resolution-independent; that is, they can be used by any printer that supports outline fonts. When a printer needs to use a given character (and not until then), it creates a rasterized bitmap from an outline font, scaling the character larger or smaller as needed (see Figure 74 on page 234). A raster font character, by contrast, is stored and used in its full, rasterized size.

---

[11] For example, the 3900-0W1, 3900 Duplex, 3900 wide/duplex, 3935, and 3130 printers.
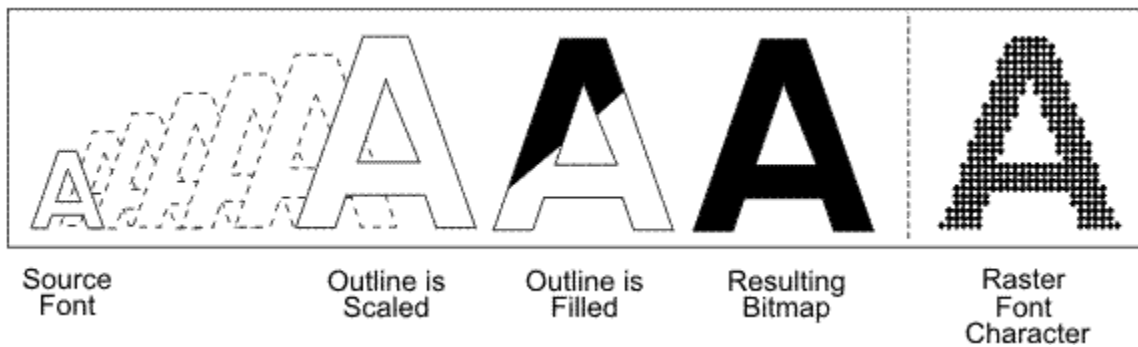
Figure 74. Comparison of an outline font character with a raster font character

Sending default characters in single-byte outline fonts

Because the Load Code Page Control (LCPC) IPDS command is used to download an outline code page to the printer, PSF can use this command to tell some printers to print the default character whenever a missing code point or missing character is encountered. For those printers that support this function, the default character for both single-byte and double-byte outline code pages is sent to the printer through the LCPC IPDS command.

Many printers do not support sending the default character to the printer with the LCPC IPDS command. For those printers, you can use the **Send default character** parameter in the Printer Inventory (see Send default character) or XTP7SDCH in Exit 7 (see "XTP7SDCH" on page 436) to request that PSF pass the default character information to the printer by fully populating the code page of an outline single-byte code page (this does not work for an outline double-byte code page). The Code Page Control (CPC) structured field in the code page identifies the ID of the character to use when no character is assigned to a code point. By fully populating the code page, any code points that are undefined are completed with the default character information from the CPC structured field. If the code page is not fully populated, the printers use a space if an undefined code point is used in the data.

Unlike raster fonts, PSF cannot verify that the default character selected is actually in the character set. Therefore, it is up to the user to ensure that the default character selected is actually in the character set. The printer uses a space as the default character if a code point calls out a character that does not exist in the font character set.

The **Send default character** parameter in the Printer Inventory or XTP7SDCH in Exit 7 is ignored if it is used with a printer that supports sending the default character with the LCPC IPDS command.

Concatenating outline fonts with other library resources

An outline font can be placed inline in the print data, in a user library, in a security library, or in a system library. Outline fonts can be stored in the same library as raster fonts, or they can be stored in a separate library. For more information, see "Concatenating defined libraries to system libraries" on page 51.

Creating fonts

You can create fonts by using, for example, Type Transformer. With Type Transformer, you can create or modify raster font character sets, outline font character sets, and code pages in FOCA format, and store them in host font libraries for use with PSF.

You can add entries to PSF's font mapping tables that identify new outline font character sets and their raster equivalents in different sizes. These fonts can then be specified in print jobs with either outline font member names, raster font member names, or GRIDs. Working from the mapping tables, PSF selects either the raster version or the outline version of your new fonts, whichever can be used with your printer.

To add a single-byte outline font:

1. Use Type Transformer to create or modify a FOCA outline font character set and coded font from an Adobe Type 1 font. If you do not have a code page that can be used with the new font character set, you also need to use Type Transformer to create or modify a code page.

2. If it is possible that the jobs that reference outline fonts need printing on a printer that supports raster font technology only, use Type Transformer to generate equivalent FOCA raster fonts in the sizes you need and in the resolution that your raster font printer supports (either 240-pel or 300-pel).

3. If the font is to be included as an inline font and it is possible that the job is routed to printers that do not support outline fonts, generate the FOCA raster equivalent of this font (by using Type Transformer) in the size and resolution you need and include it inline.

4. Move the new FOCA outline font and raster font character sets, code page, and coded font to the z/OS host font libraries.

5. Modify the Font-Name-to-GRID Mapping table (APSRNTGM) and the GRID-to-Font-Name Mapping table (APSRGTNM) to include the new outline font character set, the equivalent raster font character sets, and the new code page, as described in "Using the font mapping tables" on page 237.

6. Assemble the user mapping tables, and link-edit them with PSF.

7. Make sure that the z/OS host font libraries are concatenated in the PSF startup procedure.

Capturing new outline fonts

Some printers, such as the 3130 printer, can capture downloaded fonts. For more information, see Chapter 20, "Marking fonts for font capture," on page 231.

Font mapping

PSF sometimes needs to equate a font specified in a print job to another font. The process of equating fonts is called *font mapping*. The mapping process must not be confused with font substitution (which is done by use of Exit 7). The mapping process is used to verify that the specified font can be used in the current environment, or to find an equivalent font that can be used.

PSF maps fonts in any of three situations:

- The technology of the font is incompatible with the technology that the printer supports. For example, PSF maps fonts when an outline font is specified in a print job but the printer supports only raster fonts. This process is called a *technology-compatibility map*.

  Note: Double-byte fonts can be technology-mapped only at the coded font prefix level. They are not mapped through the user mapping tables.

- A GRID, rather than member names, is specified in the print data. PSF uses the mapping tables to find the corresponding code page and character set member names compatible with the technology of the attached printer and the setting of the **Map to outline fonts** parameter in the Printer Inventory, the MAP2OLN parameter on the PRINTDEV statement, or XTP7MTOF in Exit 7 (see the next bullet). This process is called a *GRID map*.

- The Printer Inventory, the PRINTDEV statement, or Exit 7 requests that PSF map fonts to outline fonts and the attached printer supports outline fonts. This process is called an *outline map*. See "Mapping to outline fonts" on page 236 for more information.

The user does not select technology-compatibility mapping or GRID mapping; PSF does these mappings automatically.

The different mapping functions are done at different times during the processing of the font. The choice of mapping functions to be done depends on how the font was specified in the print job. For example, a coded font is mapped by one process and a pair of code page and character set names by another process.

Mapping to outline fonts

If your printer supports outline fonts and you have existing applications that use raster fonts, and you want to use outline fonts without changing the applications, you can request that PSF map fonts to outline fonts.

The **Map to outline fonts** parameter in the Printer Inventory (see Map to outline fonts), the MAP2OLN parameter on the PRINTDEV statement (see MAP2OLN), or XTP7MTOF in Exit 7 (see "XTP7MTOF" on page 432) controls whether PSF maps fonts to outline fonts. This option defaults to NO, which specifies that PSF is not to map fonts to outline fonts. If the option is set to YES, PSF attempts to map the font to a corresponding outline font by using user and system mapping tables. For more information, see "Using the font mapping tables" on page 237.

Note: PSF does not map a font to an outline font when:

- The font is used on an auxiliary data set.
- The font is an inline font or a font from a user library.
- A double-byte raster font section is requested by character set and code page name. Double-byte raster coded fonts are prefix-mapped to equivalent outline coded fonts.
- PSF senses that the font is already an outline font.
- The target printer does not support outline fonts.

For performance considerations, see "Using outline fonts" on page 189.

Recovering from font not found

If your printer supports outline fonts and you requested that PSF map to outline fonts (see "Mapping to outline fonts" on page 236), but many pages in your jobs are not processed because the outline font identified through the mapping algorithms did not exist on the host, you might want to turn on the Recover-from-Font-Not-Found function.

The **Recover from font not found** parameter in the Printer Inventory (see Recover from font not found), the MAP2OLN parameter on the PRINTDEV statement (see MAP2OLN), or XTP7RFNF in Exit 7 (see "XTP7RFNF" on page 432) controls whether PSF makes sure that the mapped font exists before proceeding. This option defaults to NO, which specifies to PSF that it does not need to make sure that the mapped font exists before proceeding. If the mapped font does not exist, PSF issues an error message and stops processing the page.

To specify that PSF is to make sure that the outline font derived from the mapping algorithms (for example, the mapped font) exists before proceeding, set the option to YES. PSF then does library queries to determine whether the mapped font exists. If PSF determines that the mapped font exists, it proceeds to load the mapped font. If the mapped font is not found inline or in any specified user library or system library, PSF does not map the font, but instead loads the raster font.

Note: Because more library queries exist for every mapped font resource, setting this option to YES degrades the performance of PSF.

Timing of mapping operations

The technology-compatibility map and the GRID map functions are done before the resource-access call to Exit 7, if that call is enabled. This means that Exit 7 can be called with font names that are different from the names specified in the job.

The outline map function happens after the resource-access call to Exit 7 and before the resource-load call to Exit 7, if those calls are enabled.

Mapping algorithms

PSF uses different processes to map one technology to another; the process that is chosen depends on how the input font is specified:

- For coded font names, the mapping process consists of merely changing the prefix of the input font, X0 or X1–Xg, to another prefix, Xz; the rest of the name remains the same. This is called *prefix mapping.*
- For fonts specified as GRIDs, PSF uses the mapping tables to find the corresponding code page and character set names.
- For fonts specified as pairs of code page and character set names, PSF uses the mapping tables to find another pair of code page and character set names or to verify that the original names are of the technology that is wanted.

Using the font mapping tables

PSF has font mapping tables of two types: user tables that you can modify and default internal system tables. These tables are used to map single-byte fonts of one technology to single-byte fonts of another technology.

The user mapping tables can be used to:

- Map input single-byte font names to equivalent raster font or outline font names. For example, if you added outline fonts or raster fonts to your host system, you can add entries to the user mapping tables to ensure that you can use these fonts when printing on printers that support either font technology.
- Selectively disable the mapping of the internal system tables. For example, if you want to use a particular raster font in a printer that supports both raster fonts and outline fonts, and if **Map to outline fonts** in the Printer Inventory (see Map to outline fonts), the MAP2OLN parameter on the PRINTDEV statement (see MAP2OLN), or XTP7MTOF in Exit 7 (see "XTP7MTOF" on page 432) is set to YES, you can add an entry to the user mapping tables to disable the mapping of this particular font so that it must be printed as a raster font.

Appendix L, "Mapping tables for outline fonts," on page 467, contains two PSF font mapping tables:

- APSRGTNM, which maps global resource identifiers (GRIDs) to font names
- APSRNTGM, which maps font names to GRIDs

These tables are optional, in that they do not have to be link-edited with the rest of PSF. APSRNTGM is also used by the font library utility. For more information, see "Using the Font Library Update (FLU) utility" on page 348.

When PSF is doing the mapping functions, it searches the user tables first. A mapping in the user tables takes precedence over a similar mapping in the internal system tables.

Each time that you modify the mapping tables, you must compile them and link-edit them with PSF.

Adding a mapping of a raster font to an outline font

To add a mapping of a raster font name to an outline font name:

1. Set **Map to outline fonts** in the Printer Inventory, the MAP2OLN parameter on the PRINTDEV statement, or XTP7MTOF in Exit 7 to YES to make sure that outline mapping is requested.
2. Add an entry to the Font-Name-to-GRID-Mapping table (APSRNTGM) to associate the raster font character set with its GRID parts, font technology, and vertical font size. For example:

```
DC CL8'C0MYRAST'    CHARACTER SET NAME
DC XL1'00'          RESERVED
DC XL1'05'          TECHNOLOGY ID - RASTER
DC XL2'1111'        FGID
DC XL2'2222'        GCSGID
DC XL2'0000'        FONT WIDTH
DC XL2'00C8'        VERTICAL FONT SIZE
```

You can find the FGID, GCSGID, and font size for all IBM-supplied AFP fonts in *IBM AFP Fonts: Font Summary for AFP Font Collection* or *z/OS Font Collection*. For non-IBM supplied fonts or raster fonts that are created from OEM outlines by using the IBM Type Transformer product, either determine the values that were used when the raster fonts were created or assign values yourself by using FGID and GCSGID values in the customer-assigned range 65280 - 65534 (decimal).

Code the font technology for the raster font character set as XL1'05'.

Specify a size parameter so that the printer knows what point size to use to render the selected font. You can specify font width, vertical font size, or both. Because vertical font size is more easily understood, it is preferable to use this value. Vertical font size is the point size multiplied by 20 (decimal).

If you created a specific code page for use with this font character set, use the GCSGID associated with that code page. Otherwise, code the GCSGID entry with a value of XL2'FFFF'.

3. If you are using a standard code page with this character set, you need not create a code page entry; PSF uses the internal system tables. If you created a specific code page for use with your character set, add an entry to the Font-Name-to-GRID Mapping table (APSRNTGM) to associate that code page with its GRID parts. For example:

```
DC CL8'T1MYCDPG'    CODE PAGE NAME
DC XL2'1234'        CPGID
DC XL2'2222'        GCSGID
```

4. Add an entry to the GRID-to-Font-Name Mapping table (APSRGTNM) to associate the FGID and GCSGID of the corresponding raster font character set entry in the Font-Name-to-GRID Mapping table (APSRNTGM) with the name of the outline font character set. For example, an outline font character set that corresponds to the raster font entry in Step can be coded like this:

```
DC XL2'1111'        FGID
DC XL2'2222'        GCSGID
DC XL2'0000'        FONT WIDTH
DC XL1'00'          RESERVED
DC XL1'1F'          TECHNOLOGY ID - OUTLINE
DC XL2'0000'        VERTICAL FONT SIZE
DC CL8'CZMYOUTL'    CHARACTER SET NAME
```

Use the same FGID and GCSGID parameters that were assigned to the raster font. These values link the raster font to the outline font.

Code the font technology for the outline font character set as XL1'1F'. This outline font uses the same code page as the corresponding raster font character set entry in the Font-Name-to-GRID Mapping table (APSRNTGM). If you created a specific code page for use with this font character set, use the GCSGID associated with that code page. Otherwise, code the GCSGID entry with XL2'FFFF'.

This entry is also for GRID-to-outline font mapping.

5. If you are using a standard code page with this character set, you need not create a code page entry; PSF uses the internal system tables. If you created a specific code page for use with your character set, add an entry to the GRID-to-Font-Name Mapping table (APSRGTNM), to associate that code page with its GRID parts. For example:

```
DC XL2'1234'        CPGID
DC XL2'2222'        GCSGID
DC CL8'T1MYCDPG'    CODE PAGE NAME
```

Adding a mapping of an outline font to a raster font

To add a mapping of an outline font to a raster font:

1. Add an entry to the Font-Name-to-GRID Mapping table (APSRNTGM), to associate the outline font character set with its GRID parts, font technology, and vertical font size. For example:

```
DC CL8'CZMYOUTL'    CHARACTER SET NAME
DC XL1'00'          RESERVED
DC XL1'1F'          TECHNOLOGY ID - OUTLINE
DC XL2'1111'        FGID
DC XL2'2222'        GCSGID
DC XL2'0000'        FONT WIDTH
DC XL2'0000'        VERTICAL FONT SIZE
```

You can find the FGID, GCSGID, and font size for all IBM-supplied AFP fonts in *IBM AFP Fonts: Font Summary for AFP Font Collection* or *z/OS Font Collection*. For non-IBM supplied fonts or outline fonts that are created from OEM outlines by using the IBM Type Transformer product, either determine the values that were used when the raster fonts were created or assign values yourself by using FGID and GCSGID values in the customer-assigned range of 65280 to 65534 (decimal).

Code the font technology for the outline font character set as XL1'1F'.

If you created a specific code page for use with this font character set, use the GCSGID associated with that code page. Otherwise, code the GCSGID entry with a value of XL2'FFFF'.

2. If you are using a standard code page with this character set, you need not create a code page entry; PSF uses the internal system tables. If you created a specific code page for use with your character set, add an entry to the Font-Name-to-GRID Mapping table (APSRNTGM), to associate that code page with its GRID parts. For example:

```
DC CL8'T1MYCDPG'    CODE PAGE NAME
DC XL2'1234'        CPGID
DC XL2'2222'        GCSGID
```

3. Add an entry to the GRID-to-Font-Name Mapping table (APSRGTNM) to associate the FGID and GCSGID of the corresponding outline font character set entry in the Font-Name-to-GRID Mapping table (APSRNTGM) with the name of the raster font character set. For example:

```
DC XL2'1111'        FGID
DC XL2'2222'        GCSGID
DC XL2'0000'        FONT WIDTH
DC XL1'00'          RESERVED
DC XL1'05'          TECHNOLOGY ID - RASTER
DC XL2'00C8'        VERTICAL FONT SIZE
DC CL8'C0MYRAST'    CHARACTER SET NAME
```

Note: If you use the outline font at different point sizes, you need an entry in the GRID-to-Font-Name Mapping table (APSRGTNM) for the raster font in each size.

Use the same FGID and GCSGID parameters that were assigned to the outline font. These values link the outline font to the raster font.

Code the font technology for the raster font character set as XL1'05'.

Specify a size parameter so that the printer knows what point size to use to render the selected font. You can specify font width, vertical font size, or both. Because vertical font size is more easily understood, it is preferable to use this value. Vertical font size is the point size multiplied by 20 (decimal).

If you specify a nonzero font width, this entry is also used for GRID-to-raster font mapping.

This raster font uses the same code page as the corresponding raster font character set entry in the Font-Name-to-GRID Mapping table (APSRNTGM). If you created a specific code page for use with this font character set, use the GCSGID associated with that code page. Otherwise, code the GCSGID entry with XL2'FFFF'.

4. If you are using a standard code page with this character set, you need not create a code page entry; PSF uses the internal system tables. If you created a specific code page for use with your character set, add an entry to the GRID-to-Font-Name Mapping table (APSRGTNM), to associate that code page with its GRID parts. For example:

```
DC XL2'1234'        CPGID
DC XL2'2222'        GCSGID
DC CL8'T1MYCDPG'    CODE PAGE NAME
```

Disabling the mapping of a raster font to an outline font

To disable the mapping of a raster font to an outline font, add an entry to the Font-Name-to-GRID Mapping table (APSRNTGM), which associates the raster font character set name with an FGID of XL2'0000':

```
DC CL8'C0MYRAST'    CHARACTER SET NAME
DC XL1'00'          RESERVED
DC XL1'05'          TECHNOLOGY ID - RASTER
DC XL2'0000'        FGID
DC XL2'0000'        GCSGID
DC XL2'0000'        FONT WIDTH
DC XL2'0000'        VERTICAL FONT SIZE
```

Adding a mapping of a GRID to a raster font

If you added a mapping for an outline font to this raster font character set, you already added an entry to map the GRID to a raster font. To use the GRID-to-raster font mapping, be sure that the font width is coded with a non-zero value.

If the GRID-to-Font-Name Mapping table (APSRGTNM) does not contain an entry for your raster font character set, do these:

1. Add an entry to the GRID-to-Font-Name Mapping table (APSRGTNM) to associate the FGID, the GCSGID, and the size to the raster font character set name. For example:

   ```
   DC XL2'1111'    FGID
   DC XL2'2222'    GCSGID
   DC XL2'0080'    FONT WIDTH
   DC XL1'00'      RESERVED
   DC XL1'05'      TECHNOLOGY ID - RASTER
   DC XL2'0000'    VERTICAL FONT SIZE
   DC CL8'C0MYRAST'    CHARACTER SET NAME
   ```

   You can find the FGID, GCSGID, and font size for all IBM-supplied outline fonts in *IBM AFP Fonts: Font Summary for AFP Font Collection* or *z/OS Font Collection*. For non-IBM supplied fonts or outline fonts that are created from OEM outlines by using the IBM Type Transformer product, either determine the values that were used when the raster fonts were created or assign values yourself by using FGID and GCSGID values in the customer-assigned range of 65280 to 65534 (decimal).

   Code the font technology for the raster font character set as XL1'05'.

   For raster font character sets, specify a font width. You can also specify a vertical font size.

   If you created a specific code page for use with this font character set, use the GCSGID associated with that code page. Otherwise, code the GCSGID entry with XL2'FFFF'.

2. If you are using a standard code page with this character set, you need not create a code page entry; PSF uses the internal system tables. If you created a specific code page for use with your character set, add an entry to the GRID-to-Font-Name Mapping table (APSRGTNM), to associate that code page with its GRID parts. For example:

   ```
   DC XL2'1234'    CPGID
   DC XL2'2222'    GCSGID
   DC CL8'T1MYCDPG'    CODE PAGE NAME
   ```

Adding a mapping of a GRID to an outline font

If you added a mapping for a raster font to this outline font character set, you already added an entry to map the GRID to an outline font.

If the GRID-to-Font-Name Mapping table (APSRGTNM) does not contain an entry for your outline font character set, do these:

1. Add an entry to the GRID-to-Font-Name Mapping table (APSRGTNM) to associate the FGID and the GCSGID with the outline font character set name. For example:

```
DC XL2'1111'        FGID
DC XL2'2222'        GCSGID
DC XL2'0000'        FONT WIDTH
DC XL1'00'          RESERVED
DC XL1'1F'          TECHNOLOGY ID - OUTLINE
DC XL2'0000'        VERTICAL FONT SIZE
DC CL8'CZMYOUTL'    CHARACTER SET NAME
```

You can find the FGID, GCSGID, and font size for all IBM-supplied raster fonts in *IBM AFP Fonts: Font Summary for AFP Font Collection* or *z/OS Font Collection*. For non-IBM supplied fonts or raster fonts that are created from OEM outlines by using the IBM Type Transformer product, either determine the values that were used when the raster fonts were created or assign values yourself by using the FGID and GCSGID values in the customer-assigned range of 65280 to 65534 (decimal).

Code the font technology for the outline font character set as XL1'1F'.

For outline font character sets, specify font width and vertical font size with XL2'0000'.

If you created a specific code page for use with this font character set, use the GCSGID associated with that code page. Otherwise, code the GCSGID entry with XL2'FFFF'.

2. If you are using a standard code page with this character set, you need not create a code page entry; PSF uses the internal system tables. If you created a specific code page for use with your character set, add an entry to the GRID-to-Font-Name Mapping table (APSRGTNM), to associate the code page with its GRID parts. For example:

```
DC XL2'1234'        CPGID
DC XL2'2222'        GCSGID
DC CL8'T1MYCDPG'    CODE PAGE NAME
```

Chapter 22. Using TrueType and OpenType Fonts

TrueType and OpenType fonts are outline fonts that are not defined by Font Object Content Architecture (FOCA). PSF supports Unicode-enabled TrueType and OpenType fonts, which have tables that identify the formatting information that is used to support Unicode encoding. TrueType and OpenType fonts are stored in the printer, inline in the print data set, or in UNIX files (Hierarchical File System (HFS) or z/OS File System (zFS) files) in user path libraries or in system font path libraries. TrueType and OpenType fonts must be referenced in the print data set by using the Map Data Resource (MDR) structured field.

Note: MO:DCA IS/3 data streams support only TrueType and OpenType fonts.

For more information about TrueType and OpenType fonts, see *PSF for z/OS: User's Guide* and *Using OpenType Fonts in an AFP System*.

PSF limitations for TrueType and OpenType fonts

PSF has these TrueType and OpenType font support limitations:

- PSF does not manage TrueType and OpenType fonts with installation Exit 7.
- PSF supports only TrueType and OpenType fonts on printers that support Unicode.

Obtaining TrueType and OpenType fonts

A basic set of TrueType and OpenType fonts is provided with the WorldType Fonts for AFP Print Servers optional feature of Infoprint Fonts for Multiplatforms (Program Number 5648-E77) and the z/OS Font Collection, a base feature of z/OS. the z/OS Font Collection contains prebuilt resource access tables (RATs) for TrueType and OpenType fonts that contain formal font names, paths, file names, and font attributes. The TrueType and OpenType fonts are mapped from their formal names to a path and file name that PSF can access.

See *z/OS Font Collection* and *IBM Infoprint Fonts: Font Summary* for more information about TrueType and OpenType fonts.

Setting up PSF to use TrueType and OpenType fonts

To use TrueType and OpenType fonts with PSF, you must do these steps:

1. Identify the path libraries where the fonts are found:
   - You can identify the system font path libraries in the PSF startup procedure with the FONTPATH parameter on the PRINTDEV statement. See FONTPATH.
   - The user can identify user path libraries with the USERPATH parameter on the OUTPUT JCL statement. see *PSF for z/OS: User's Guide*.
2. Use z/OS UNIX System Services to set access permissions to the directories and files in the font path libraries. Read permissions must be set for each path directory and file PSF is accessing. If the permissions for each directory or file are not set correctly, PSF cannot access the path library. For more information about z/OS UNIX System Services, see *z/OS UNIX System Services Command Reference*.

Chapter 23. Marking resources for distributed print function

A distributed print function (DPF) program runs on a personal computer. DPF makes it possible to store PSF resources—fonts, overlays, and page segments—in the hard disk storage of the personal computer. As Figure 75 on page 245 shows, these stored resources can then be used by a printer that is attached to the personal computer.
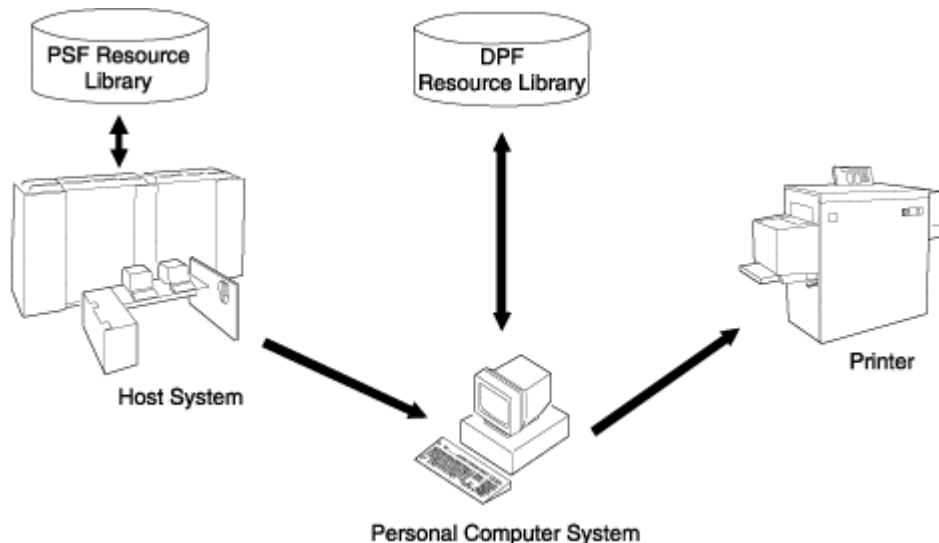
Figure 75. Access to resources by DPF

Note: A PSF direct program does not store resources in the intermediate device.

If requested, a DPF program might access these resources directly from the hard disk and load them into the printer at IBM Z channel-emulation speeds, thus reducing the amount of data that PSF sends over slow-speed communication lines.

PSF provides the APSRMARK utility, which specifies how DPF is to manage resources that are stored in the PSF resource library. This information describes the JCL statements that are used to start the APSRMARK utility.

Note: You can mark a resource only once and then use it with DPF. You do not need to mark separate resources for each program.

Using APSRMARK to mark resources

Before you can store a PSF resource in the DPF resource library, you must mark (identify) that resource by using the APSRMARK utility.

APSRMARK is a batch program started with control statements that mark:

- Sequential data sets
- Individual members of partitioned data sets
- All members of a partitioned data set

To mark a resource, you provide JCL statements to run the APSRMARK utility. Marking a resource updates certain structured fields for that resource. For example, marking an overlay updates the Begin Overlay structured field.

Notes:

1. You cannot concatenate partitioned data sets as input to APSRMARK.
2. You can also use APSRMARK to identify resident fonts. See Chapter 19, "Supporting resident fonts," on page 221.
3. You can use APSRMARK to identify the same resources for use by DPF and printer-resident fonts. However, with DPF, you cannot use printer-resident resources; you must use the resources in the DPF resource library. For more information, see "Using DPF fonts and printer-resident fonts with APSRMARK" on page 252.
4. DPF does not support outline fonts.
5. You can use APSRMARK to mark inline resources and use them on printers attached through DPF or printers with resident fonts. PSF uses these marked resources just as it uses other resources marked with APSRMARK.

Control-statement syntax for APSRMARK

A SYSIN DD statement precedes the control statements that provide input to APSRMARK. These statements use the following syntax:

- Control statements must be 80-column records.
  - The first 71 columns can contain valid APSRMARK parameters.
  - Statements that do not fit on one line can be continued on the next line by using any nonblank character in column 72.
  - APSRMARK ignores columns 73 - 80. If you want, you can put a sequence number in these columns.
- Blanks are optional and can be placed between parameters to make the statement more readable.
- An asterisk (*) in column 1 denotes a comment statement.

APSRMARK reports any syntax errors and I/O errors found in the parameters of the control statement.

Control-statement parameters for APSRMARK

Each APSRMARK control statement can contain the following parameters:

INDD=*DDname*
Specifies the name of the input data set you use when you mark or report resources. The INDD parameter can name either a sequential data set or a partitioned data set. This parameter is required.

MEMBER=*name*|All
This parameter is required for partitioned data sets. Do not specify this parameter for sequential data sets.

name
Specifies that resources are to be marked or reported for a single member (primary or alias) of a partitioned data set. When resources are marked, both the primary member and its aliases are marked unless the primary is updated or deleted after the alias is generated, in which case the alias is not updated. An alias whose primary is updated or deleted can be marked only by specifying the name of the alias and specifying the NEWNAME keyword.

ALL
Specifies that resources are to be marked or reported for all the members of a partitioned data set. When resources are marked, all primary members and aliases whose primaries are not updated or deleted are marked. Aliases whose primaries are updated or deleted are not marked. When resources are being reported, only primary members are listed. When you are marking multiple fonts for printer-resident use, do not specify MEMBER=ALL. Instead, specify each member name in multiple control statements. However, you can specify MEMBER=ALL when you specify the REPORT parameter.

NEWNAME=*name*
Specifies a new name for a member of a partitioned data set. This parameter is optional and has meaning only when a specific member is specified by MEMBER=*name*. The newly named member is marked and becomes a primary member without an associated alias.

OUTDD=*DDname*
Specifies the output data set where the marked resources are placed. You can use the REPLACE parameter to mark resources (or change resource marking) and replace them in the same data set. The OUTDD parameter is required.

{PRIVATE|PUBLIC}
Specifies the storage and use of PSF resources:

- Resources marked PRIVATE cannot be stored in the DPF resource library; instead, they are passed through to DPF for the one-time use of your printing application. (The default marking is PRIVATE, and PSF treats any unmarked resources as PRIVATE.)

- Resources marked PUBLIC are sent from the host system and stored in the DPF resource library only once; this eliminates the need for the host to send the resource to the printer each time it is printed.

You must code either PUBLIC or PRIVATE on each control statement.

REPLACE
Specifies that existing members of the specified output partitioned data set are to be overwritten (replaced). Use REPLACE either to mark resources initially and replace them in the same data set, or to change existing resource markings. This parameter is optional and pertains only to partitioned data sets.

Note: If you change a PSF resource, you must use APSRMARK to mark it again so that the changed version of the resource is managed correctly by DPF.

REPORT
Specifies that a report is to be formatted and printed. This report lists how resources are marked. The INDD and MEMBER parameters determine which resources are reported:

- INDD specifies the name of the input data set.

- MEMBER=ALL specifies that a report entry is made for all primary members of a partitioned data set.

- MEMBER=*name* specifies that a report entry is made for a single primary or alias member of a partitioned data set.

SHORT|LONG
Specifies which report is formatted and printed when the REPORT parameter is specified:

- SHORT specifies that a short report is generated. See Figure 80 on page 251 for an example of a short report.

- LONG specifies that a long report is generated. This is the default report format. See Appendix D, "APSRMARK long report format," on page 401 for an example of a long report.

Examples of control statements for APSRMARK

This information contains APSRMARK control statements for:

- Creating a PSF resource library with all members marked PUBLIC
- Creating a PSF resource library with some members marked PRIVATE
- Modifying an existing PSF resource library
- Printing a report of resource markings for a partitioned data set

In these four examples, lowercase letters represent JCL that you can customize for your organization.

PSF provides sample jobs used for marking resources. For more information, see Appendix C, "APSRMARK sample jobs," on page 399.

Creating a PSF resource library with all members marked PUBLIC

Figure 76 on page 248 shows the APSRMARK control statements for creating a copy of each member in a partitioned data set (for example, the system font library) and marking them PUBLIC.

```
//example1 JOB ,'IBM USER',MSGCLASS=A
//STEP1  EXEC PGM=APSRMARK
//*
//SYSPRINT DD SYSOUT=class
//IN1      DD UNIT=dasd,DISP=SHR,
//            DSN=dsname1,
//            VOL=SER=volser
//OUT1     DD UNIT=dasd,DISP=(NEW,KEEP),
//            DSN=dsname2,
//            SPACE=(same as IN1),
//            DCB=(same as IN1),
//            VOL=SER=volser
//SYSIN    DD *
  INDD=IN1,OUTDD=OUT1,PUBLIC,MEMBER=ALL
 /*
```

Figure 76. Sample JCL for creating a PSF resource library and marking all members PUBLIC

The sample JCL in Figure 76 on page 248 uses the following statements:

IN1 DD
　Defines the input partitioned data set that contains the members to be marked by APSRMARK.

OUT1 DD
　Creates an output partitioned data set that contains the PUBLIC members.

　Note: The data set attributes for OUT1 must match the attributes for IN1.

SYSIN DD *
　Precedes the control statements that specify that all members of IN1 are to be marked PUBLIC and placed in OUT1.

Creating a PSF resource library with some members marked PRIVATE

Figure 77 on page 248 shows the APSRMARK control statements for creating a partitioned data set that contains only PUBLIC members and for marking some members in the same data set as PRIVATE. Notice that this task is accomplished in two separate JCL steps. Members in an existing partitioned data set can be processed more than once in the same JCL step.

```
//example2 JOB ,'IBM USER',MSGCLASS=A
//STEP1  EXEC PGM=APSRMARK
//*
//SYSPRINT DD SYSOUT=class
//IN1      DD UNIT=dasd,DISP=SHR,
//            DSN=dsname1,
//            VOL=SER=volser
//INOUT1   DD UNIT=dasd,DISP=(NEW,KEEP),
//            DSN=dsname2,
//            SPACE=(same as IN1),
//            DCB=(same as IN1),
//            VOL=SER=volser
//SYSIN    DD *
  INDD=IN1,OUTDD=INOUT1,PUBLIC,MEMBER=ALL
 /*
//STEP2  EXEC PGM=APSRMARK
//*
//SYSPRINT DD SYSOUT=class
//INOUT1   DD UNIT=dasd,DISP=SHR,
//            DSN=dsname2,
//            VOL=SER=volser
//SYSIN    DD *
  INDD=INOUT1,OUTDD=INOUT1,PRIVATE,REPLACE,MEMBER=name1
  INDD=INOUT1,OUTDD=INOUT1,PRIVATE,REPLACE,MEMBER=name2
 /*
```

Figure 77. Sample JCL for creating a PSF resource library and marking some members PRIVATE

The sample JCL in Figure 77 on page 248 uses the following statements:

STEP1

　IN1 DD
　　Defines the input partitioned data set that contains the members to be marked by APSRMARK.

INOUT1 DD

Creates an output partitioned data set that contains the PUBLIC members.

Note: The data set attributes for INOUT1 must match the attributes for IN1.

SYSIN DD *

Precedes the control statements that specify that all members of IN1 are to be marked PUBLIC and placed in INOUT1.

STEP2

INOUT1 DD

Defines the existing partitioned data set that contains members to be changed from PUBLIC to PRIVATE.

SYSIN DD *

Precedes the control statements that specify that members *name1* and *name2* are marked PRIVATE and replaced in INOUT1.

Modifying an existing PSF resource library

Figure 78 on page 249 shows the APSRMARK control statements for changing one member in a partitioned data set from PRIVATE to PUBLIC. This example also shows how to take members from one partitioned data set and place them in a different partitioned data set.

```
//example3 JOB ,'IBM USER',MSGCLASS=A
//STEP1  EXEC PGM=APSRMARK
//*
//SYSPRINT DD SYSOUT=class
//INOUT1   DD UNIT=dasd,DISP=SHR,
//            DSN=dsname1,
//            VOL=SER=volser
//IN2      DD UNIT=dasd,DISP=SHR,
//            DSN=dsname2,
//            VOL=SER=volser
//OUT2     DD UNIT=dasd,DISP=SHR,
//            DSN=dsname3,
//            VOL=SER=volser
//SYSIN    DD *
  INDD=INOUT1,OUTDD=INOUT1,PUBLIC,REPLACE,MEMBER=name1
  INDD=IN2,OUTDD=OUT2,PUBLIC,MEMBER=name2
  INDD=IN2,OUTDD=OUT2,PUBLIC,MEMBER=name3
/*
```

Figure 78. Sample JCL for modifying an existing PSF resource library

The JCL in Figure 78 on page 249 uses the following statements:

INOUT1 DD

Defines the existing partitioned data set that contains a member to be changed from PRIVATE to PUBLIC.

IN2 DD

Defines the input data set that contains the members to be marked PUBLIC.

OUT2 DD

Defines the output partitioned data set that contains the members from IN2 marked PUBLIC.

SYSIN DD *

Precedes the control statements that specify how APSRMARK marks these members:

- The first INDD and OUTDD statements specify that member *name1* is to be marked PUBLIC and replaced in INOUT1.
- The second and third INDD and OUTDD statements specify that the members *name2* and *name3* in IN2 are to be marked PUBLIC and placed in OUT2.

Printing a report of resource markings in a partitioned data set

Figure 79 on page 250 shows the APSRMARK control statements for requesting a printed report on the resource-marking status of all members of a partitioned data set.

```
//example4 JOB ,'IBM USER',MSGCLASS=A
//step1   EXEC PGM=APSRMARK
//*
//SYSPRINT DD SYSOUT=class
//IN1      DD UNIT=dasd,DISP=SHR,DSN=dsname,VOL=SER=volser
//SYSIN    DD *
  INDD=IN1,MEMBER=ALL,REPORT,SHORT
/*
```

Figure 79. Sample JCL for reporting resource markings in a partitioned data set

The JCL in Figure 79 on page 250 uses the following statements:

IN1 DD
> Defines the partitioned data set that contains the members for which a resource-marking report is to be printed.

SYSIN DD *
> Precedes the control statements that specify the parameters for the report.

contains an example of a printed short report. For information about generating a long report, see "Generating a long report of resident font marking" on page 229.

```
APS580I  INDD=IN,REPORT,MEMBER=ALL
APS580I  C0D0GB10 PUBLIC  SAVE   89-07-21 12:50:56 95D6 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0D0GB12 PUBLIC  SAVE   89-07-21 12:50:56 3E0D 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0D0GI12 PUBLIC  SAVE   89-07-21 12:50:56 9CBF 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0D0GT10 PUBLIC  SAVE   89-07-21 12:50:56 BE89 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0D0GT12 PUBLIC  SAVE   89-07-21 12:50:56 32B5 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0D0GT15 PUBLIC  SAVE   89-07-21 12:50:56 4717 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0D0GT20 PUBLIC  SAVE   89-07-21 12:50:56 8C38 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0D0GT24 PUBLIC  SAVE   89-07-21 12:50:56 10A7 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0D0S012 ?????? ????   ??-??-?? ??:??:?? ???? ??????????? ?????? ??????.???????
APS580I  C0L0KATA PUBLIC  SAVE   89-07-21 12:50:56 C4B9 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0L00A0A PUBLIC  SAVE   89-07-21 12:50:56 10B6 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0L00B0A PUBLIC  SAVE   89-07-21 12:50:56 10A7 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0AE10 PUBLIC  SAVE   89-07-21 12:50:56 AB9E 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0AE20 PUBLIC  SAVE   89-07-21 12:50:56 9EDC 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0BITR PUBLIC  SAVE   89-07-21 12:50:56 EAEC 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0CB10 PUBLIC  SAVE   89-07-21 12:50:56 3F2F 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0CB12 PUBLIC  SAVE   89-07-21 12:50:56 C660 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0CE12 PUBLIC  SAVE   89-07-21 12:50:56 533B 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0CI10 PUBLIC  SAVE   89-07-21 12:50:56 071E 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0CR10 PUBLIC  SAVE   89-07-21 12:50:56 8F81 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0D0TR PUBLIC  SAVE   89-07-21 12:50:56 447F 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0D224 PUBLIC  SAVE   89-07-21 12:50:56 C59D 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0EBTR PUBLIC  SAVE   89-07-21 12:50:56 6BB3 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0EITR PUBLIC  SAVE   89-07-21 12:50:56 C660 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0ELTR PUBLIC  SAVE   89-07-21 12:50:56 D07A 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0ESTR PUBLIC  SAVE   89-07-21 12:50:56 5766 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0LB12 PUBLIC  SAVE   89-07-21 12:50:56 3901 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0LR12 PUBLIC  SAVE   89-07-21 12:50:56 6B64 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S00B10 PUBLIC  SAVE   89-07-21 12:50:56 9453 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S00R10 PUBLIC  SAVE   89-07-21 12:50:56 92C0 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0PB12 PUBLIC  SAVE   89-07-21 12:50:56 963F 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0PI12 PUBLIC  SAVE   89-07-21 12:50:56 92C0 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0PR12 PUBLIC  SAVE   89-07-21 12:50:56 B9A9 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0S0SR12 PUBLIC  SAVE   89-07-21 12:50:56 2C5E 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0T055B0 PUBLIC  SAVE   89-07-21 12:50:56 E6E1 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0T05500 PUBLIC  SAVE   89-07-21 12:50:56 C300 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0T05580 PUBLIC  SAVE   89-07-21 12:50:56 F67F 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0T075F0 PUBLIC  SAVE   89-07-21 12:50:56 8EBF 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0T075N0 PUBLIC  SAVE   89-07-21 12:50:56 F27C 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0T07500 PUBLIC  SAVE   89-07-21 12:50:56 46B6 660241123081 DSNAME VOLSER1.INFONTS
APS580I  C0T15500 PUBLIC  SAVE   89-07-21 12:50:56 B2EC 660241123081 DSNAME VOLSER1.INFONTS
APS580I  T1D0BASE PUBLIC  SAVE   89-07-21 12:50:56 3F06 660241123081 DSNAME VOLSER1.INFONTS
APS580I  T1GDP500 PUBLIC  SAVE   89-07-21 12:50:56 3F06 660241123081 DSNAME VOLSER1.INFONTS
APS580I  T1L00CRB PUBLIC  SAVE   89-07-21 12:50:56 3F06 660241123081 DSNAME VOLSER1.INFONTS
APS580I  T1L00CR1 PUBLIC  SAVE   89-07-21 12:50:56 3F06 660241123081 DSNAME VOLSER1.INFONTS
APS580I  T1L02773 PUBLIC  SAVE   89-07-21 12:50:56 3F06 660241123081 DSNAME VOLSER1.INFONTS
APS580I  T1S0AE10 PUBLIC  SAVE   89-07-21 12:50:56 3D83 660241123081 DSNAME VOLSER1.INFONTS
APS580I  T1000420 PUBLIC  SAVE   89-07-21 12:50:56 3F06 660241123081 DSNAME VOLSER1.INFONTS
APS580I  T1000424 PUBLIC  SAVE   89-07-21 12:50:56 3F06 660241123081 DSNAME VOLSER1.INFONTS
```

Figure 80. Sample output from a resource-marking report

Note: A question mark with a member of a partitioned data set indicates that no marking data is available for that particular member. For an example, see C0D0S012 in .

Using page segments

You can store a page segment in the DPF resource library only if the page segment is marked PUBLIC and is defined to PSF as a *hard page segment*.

Hard page segments are named in a Map Page Segment (MPS) structured field of a MO:DCA-P page, in a page definition, or in an overlay. Hard page segments are loaded in the printer as separate objects at the start of either page processing or overlay processing, and they are retained until printing of the data set is finished. By contrast, *soft page segments* are segments that are not named in the MPS structured field. Soft page segments are sent to the printer each time they are called, as part of the page data. You cannot store soft page segments in the DPF library.

In a line-format print data set, a page segment included by an Include Page Segment (IPS) record is *hard* if it is defined in the page definition with a page segment list. If it is not defined in the page definition, the page segment is *soft* and cannot be stored in the DPF library.

If a page segment is called in an overlay, it is a hard page segment if it is named in the overlay resource MPS structured field. Most products that create overlays, such as OGL, automatically specify hard page

segments. In addition, if an overlay that includes page segments is marked by the APSRMARK utility, PSF and DPF treats those page segments as hard page segments.[12]

If a page segment is called in a MO:DCA-P page, the page segment is hard or soft, depending on whether it is named in the MPS record of the active environment group for the page. For MO:DCA-P pages that are created by DCF, an option to name or map the page segment is provided on the segment-include command. When PSF processes an MPS structured field, any page segment that is specified in that MPS is treated as a hard page segment by PSF and DPF. Similarly, when an overlay that includes page segments is marked by the APSRMARK utility, PSF and DPF treat those page segments as hard page segments.[12]

Using DPF fonts and printer-resident fonts with APSRMARK

In a PSF font library, you can define fonts for printers equipped with DPF and for printers equipped with resident fonts. However, you cannot define outline fonts for use with DPF, because DPF does not support outline fonts. Also, a single PSF font library can be shared for printers equipped with DPF and for printers with resident fonts.

Using the APSRMARK utility, you can mark a font so that it is identified to PSF for use with one of these:

- DPF only
- DPF and printer-resident fonts
- Neither (the font is considered PRIVATE)

Marking a font for use with only DPF requires the following APSRMARK SYSIN control-statement parameters:

- INDD
- OUTDD
- PUBLIC
- MEMBER

Marking a font for use with DPF and printer-resident fonts requires the following APSRMARK SYSIN control-statement parameters:

- INDD
- OUTDD
- PUBLIC
- MEMBER
- RRID
- RRTIME
- RRDATE

Changing how fonts are marked

You might want to use the APSRMARK utility to change marks for:

- Fonts that are already marked for use by DPF only.
- Fonts that are already marked for DPF use and for printer-resident use.
- Fonts already marked PRIVATE.

Fonts already marked for use by DPF only

If the font is already marked for use only by DPF, and you want to also mark it for printer-resident use, follow the procedures described in Chapter 19, "Supporting resident fonts," on page 221. If you want to

---

[12] APSRMARK adds an appropriate MPS structured field for each soft page segment in an overlay that is marked for use with DPF.

mark the font PRIVATE, follow the procedures that are described in "Using APSRMARK to mark resources" on page 245 and use the PRIVATE keyword in the APSRMARK SYSIN control statement.

Fonts already marked for DPF and printer-resident use

If you want to mark a font for use by DPF only, follow the procedures that are described in Chapter 19, "Supporting resident fonts," on page 221, and use a value of 00000000 in the RRID keyword.

If you want to mark the font PRIVATE, follow the procedures that are described in Chapter 19, "Supporting resident fonts," on page 221, and use the PRIVATE keyword in the APSRMARK SYSIN control statement.

Fonts already marked PRIVATE

If you want to mark a font for use by DPF and for use as a printer-resident font, and the font is already marked PRIVATE (or it is not marked at all), follow the procedures described in Chapter 19, "Supporting resident fonts," on page 221. If you want to mark the font for only DPF, follow the procedures described in "Using APSRMARK to mark resources" on page 245.

Chapter 24. Using resources for color management

Data object resources, including color management resources (CMRs), are stored in the printer, are inline in the print data set, or are installed with resource access tables (RATs) in user or system object container path libraries in UNIX files (HFS or zFS files). PSF uses z/OS UNIX System Services to access data object resources in path libraries specified in the OBJCPATH parameter on the PRINTDEV statement of the PSF startup procedure or the USERPATH parameter on the OUTPUT JCL statement. An AFP resource installer is used to associate CMRs with data objects, so data objects can be reproduced accurately on different printers.

For more information about CMRs, data object resources, color printing, and the AFP resource installer, see color and grayscale printing in *PSF for z/OS: User's Guide*.

PSF limitations for color management resources

PSF has these limitations for data object resources used for color management:

- PSF does not manage data object resources with names greater than 8 characters with installation Exit 7.
- PSF supports only data object resources from path libraries on printers that support Unicode.

Setting up PSF to use resources for color management

To use data object resources for color management, you must define object container path libraries to PSF by doing these steps:

1. Use an AFP resource installer to install data object resources, including CMRs, in path libraries. The AFP resource installer creates data object RATs and CMR RATs, in the appropriate resource directories, that map the resources to a path and file name that PSF can access (be sure that the AFP resource installer is set up with the correct permissions to access the path libraries). When a data object is installed, CMRs can be associated with it.

2. Identify the path libraries where the resources are found:

   - Identify system object container path libraries in the PSF startup procedure with the OBJCPATH parameter on the PRINTDEV statement. See OBJCPATH.
   - Identify user path libraries with the USERPATH parameter on the OUTPUT JCL statement. See *PSF for z/OS: User's Guide*.

3. Use z/OS UNIX System Services to set access permissions to the directories and files in the path libraries. Read permissions must be set for each path directory and file PSF is accessing. If the permissions for each directory or file are not set correctly, PSF cannot access the path library. For more information about z/OS UNIX System Services, see *z/OS UNIX System Services Command Reference*.

Chapter 25. Using installation exits

This information describes how to use the installation exits supplied with PSF. This information documents intended Programming Interfaces that the customer can use to write programs to obtain the services of PSF for z/OS.

PSF provides installation exits for your use in coding and installing modifications to PSF functions. For example, with these exits, you can:

- Create your own separator pages, or modify separator-page formats supplied by PSF.
- Modify, add, or suppress output records.
- Modify System Management Facilities (SMF) type 6 records.
- Inspect, redirect, or suppress PSF messages.
- Manage resources (other than TrueType and OpenType fonts)

The installation exits that PSF supports are written in either assembly language or C language. Table 28 on page 257 lists the assembly-language name and the C-language name for each of the exits.

Table 28. Installation exit names			
Exit	**Assembly Name**	**C Name**	**Supported By**
Exit 1	APSUX01	APSUC01	PSF deferred-printing and AFP Download Plus
Exit 2	APSUX02	APSUC02	PSF deferred-printing and AFP Download Plus
Exit 3	APSUX03	APSUC03	PSF deferred-printing and AFP Download Plus
Exit 4	APSUX04	APSUC04	PSF deferred-printing and AFP Download Plus
Exit 5	APSUX05	APSUC05	PSF deferred-printing and AFP Download Plus
Exit 6	APSUX06	APSUC06	PSF deferred-printing and AFP Download Plus
Exit 7	APSUX07	APSUC07	PSF deferred-printing and AFP Download Plus
Exit 8	APSUX08	APSUC08	PSF deferred-printing and AFP Download Plus
Exit 14	APSUX14	APSUC14	PSF direct-printing
Exit 15	APSUX15	APSUC15	Download for z/OS and AFP Download Plus
Exit 16	APSUX16	APSUC16	PSF direct-printing

General rules for exit routines

Code processing in exit routines is an integral part of PSF and can affect performance. Keep the following points in mind when you are using exit routines:

- Exit routines must be coded by experienced programmers who are aware of overall system implications.
- Exit control blocks are provided in assembly language and C language.

- Use only the common control blocks passed to this exit. For a description of the common control blocks, see "Installation exit control blocks" on page 265.
- PSF runs above the 16 MB line. Exits can run in 24-bit or 31-bit addressing mode. However, 31-bit addressing is recommended, so that more functional subsystem applications (FSAs) can run in an address space. If any PSF exit is AMODE 24, PSF and JES allocate their control blocks below the 16 MB line when the control blocks are passed to the PSF exit.

  Note: Only assembler exits can be AMODE 24; C exits are always AMODE 31.
- The exit code must reference a field in a common control block only by the name associated with that field in the control block. Do not use absolute displacements; they can change from one release of PSF to another.
- Do not write exit code that issues an explicit or implied WAIT because it puts the entire PSF FSA into a WAIT state.
- Default exits, Exit 1 and Exit 2, use the non-expanded area of the ECAWKBUF area in APSUECA. If you need to pass data between exits, extend the ECAWKBUF area. (For more information, see "Exit table, APSUCOM" on page 279.) PSF and the default exits do not use the expanded area. Because all exits have access to the same ECAWKBUF area, it is possible for one exit to overwrite an area used by another exit. You must be careful not to allow this to happen.
- If certain exit-routine functions are wanted for a subset of PSF-controlled printers, consider running your printers in multiple PSF address spaces and using a STEPLIB statement to start selected exits only in those address spaces in which they are needed. The data sets on the STEPLIB statement must be Authorized Program Facility (APF)-authorized. (See the STEPLIB statement in the example in Figure 81 on page 258.)

  If the data set containing the exit is in the MVS LINKLIST, the exit is available to all PSF address spaces. If the exit is to be used by a subset of PSF address spaces:

  – Specify the data set containing the exit on the STEPLIB statement.
  – Make sure that the data set containing the exit is not in the MVS LINKLIST.

```
//APSWPRO1 PROC
//STEP01 EXEC PGM=APSPPIEP,REGION=2M
//STEPLIB DD DSN=SYS1.PSF.EXITLIB,DISP=SHR
//JOBHDR OUTPUT PAGEDEF=06080,FORMDEF=0101
//JOBTLR OUTPUT PAGEDEF=06080,FORMDEF=0101
//DSHDR OUTPUT PAGEDEF=06080,FORMDEF=0101
//MSGDS OUTPUT PAGEDEF=06080,FORMDEF=0101
//FONT01 DD  DSN=SYS1.FONTLIB,DISP=SHR
//PSEG01 DD  DSN=SYS1.PSEGLIB,DISP=SHR
//OLAY01 DD  DSN=SYS1.OVERLIB,DISP=SHR
//PDEF01 DD  DSN=SYS1.PDEFLIB,DISP=SHR
//FDEF01 DD  DSN=SYS1.FDEFLIB,DISP=SHR
//PRINTR1 CNTL
//PRINTR1 PRINTDEV  FONTDD=*.FONT01,
//     OVLYDD=*.OLAY01,
//     PSEGDD=*.PSEG01,
//     PDEFDD=*.PDEF01,
//     FDEFDD=*.FDEF01,
//     JOBHDR=*.JOBHDR,
//     JOBTRLR=*.JOBTLR,
//     DSHDR=*.DSHDR,
//     MESSAGE=*.MSGDS,
//     SETUP=FORMS,
//     BUFNO=5,
//     PAGEDEF=06080,
//     FORMDEF=0101,
//     CHARS=(60D8),
//     PIMSG=YES,
//     TRACE=YES,
//     DATACK=BLOCK
//PRINTR1 ENDCNTL
```

Figure 81. Use of STEPLIB statement for exit routine in PSF startup procedure

JCL for assembler exits

Figure 82 on page 259 contains sample JCL statements that can be used for compiling and link-editing an assembler exit on z/OS.

If you want to use this JCL to compile a PSF separator exit (APSUX01, APSUX01P, APSUX01S, APSUX02, APSUX02P, APSUX02S) with the additional Enhanced Distribution Information (ESS) SYSOUT parameters, you must change the PARM keyword on the EXEC statement to:

```
PARM=(SYSPARM(ESS),OBJECT)
```

```
//PROCEXIT JOB  MSGLEVEL=(1,1)
//***********************************************************************
//ASM     EXEC  PGM=ASMA90,PARM=OBJECT,REGION=4M
//SYSLIB   DD  DSN=SYS1.MACLIB,DISP=SHR
//         DD  DSN=SYS1.MODGEN,DISP=SHR
//SYSUT1   DD  UNIT=(SYSDA,SEP=SYSLIB),SPACE=(CYL,(10,5)),DSN=&SYSUT1
//SYSPUNCH DD  DUMMY
//SYSPRINT DD SYSOUT=*,DCB=(BLKSIZE=3509),UNIT=(,SEP=(SYSUT1,SYSPUNCH))
//SYSLIN   DD  DISP=(,PASS),UNIT=SYSDA,SPACE=(CYL,(1,1),RLSE),
//             DCB=(BLKSIZE=400),DSN=&&LOADSET
//SYSIN    DD  *
//***********************************************************************
//*** INSERT THE EXIT SOURCE CODE HERE AFTER THE SYSIN DD STATEMENT   *
//***********************************************************************
/*
//LKED   EXEC  PGM=IEWL,PARM='MAP,LET,LIST,NCAL,XREF,RENT',REGION=512K
//SYSUT1   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSPRINT DD  SYSOUT=*
//***********************************************************************
//* BE SURE TO CHANGE THE SYSLMOD DD STATEMENT BELOW TO POINT TO THE  *
//*  CORRECT LINKLIB. ALSO PUT THE CORRECT NAME OF THE EXIT IN THE    *
//*  ENTRY AND NAME STATEMENTS BELOW (SUCH AS APSUX07, NOT APSUX07T). *
//***********************************************************************
//SYSLMOD  DD DSN=xxxxxxx.xxxxxx.LINKLIB,DISP=SHR
//SYSLIN   DD  DSN=&&LOADSET,DISP=(OLD,DELETE)
//         DD  *
  ENTRY xxxxxxx
  NAME xxxxxxx(R)
/*
```

Figure 82. Sample JCL for compiling and link-editing assembler exits

Specific rules for C exits

Keep the following points in mind when you are using exit routines written in C language.

- The main function name in a C exit must be the same as the load module name. If it is different and PSF attempts to start the exit, PSF stops the FSA and issues APS097I with a return code of 12.

- Do not use the PRINTF Runtime Library Function in a C exit (APSUC14 or APSUC16) for direct printing mode; using the PRINTF function can cause PSF to go into a wait state.

- A system abend of 0C1 occurs if APSUC01 or APSUC02 is link-edited without either APSUMACC or APSUTIMC.

- If a C exit is being used, the FSS and each FSA run as a "Process". Verify that the SYS1.PARMLIB member BPXPRMxx values for MAXPROCSYS and MAXPROCUSER are large enough for each PSF FSA and the FSS to have a PROC (process). The operator command **D OMVS,O** displays the current values for these configuration parameters.

- The C exit runs in the Language Environment®. If problems occur because the Language Environment runtime options are inappropriate, you can change them by creating a CEEUOPT object file and linking it with each exit. See *z/OS Language Environment Programming Guide* for details about changing the Language Environment runtime options.

- Verify that a valid OMVS segment is defined in RACF for the user ID, group ID, or both that is associated with the PSF startup procedure.

- Table 29 on page 260 shows some of the errors that can occur when installation exits (in this case, APSUX01 for assembler Exit 1 and APSUC01 for C Exit 1) are incorrectly named, link-edited, or both. In

the table, `Exit Language` indicates the language in which the installation exit is written, `Exit Name` indicates the name of the exit, and `Link-Edit` refers to whether the exit is link-edited as a C exit or an assembler exit. An exit is link-edited as a C exit by adding the SYSLIB DD card:

```
//SYSLIB DD DSN=CEE.SCEELKED,DISP=SHR
```

Table 29. Errors from incorrectly naming or link-editing installation exits

Exit Language	Exit Name	Link-Edit	Error and Result
C	APSUX01	C	1, 5
C	APSUX01	Assembler	1, 2, 3
C	APSUC01	Assembler	2, 3
Assembly	APSUX01	C	2, 6
Assembly	APSUC01	Assembler	1, 4
Assembly	APSUC01	C	1, 2, 4

Errors and Results:

1. Exit language and exit name conflict.
2. Link-edit conflict with exit language.
3. Link-edit fails with unresolved symbols. No call library is specified.
4. When PSF attempts to start the exit, PSF stops the FSA and issues APS097I with a return code of 12.
5. When PSF attempts to start the exit, a system abend of 0C4, RC(10) occurs.
6. The exit works.

JCL for C exits

This information contains JCL that can be used when you are using C exits, including JCL for:

- Compiling a C exit
- Prelinking a C exit
- Link-editing a C exit
- Link-editing APSUMACC and APSUTIMC

Compiling a C exit

shows sample JCL statements that can be used as a guideline for compiling a C exit on z/OS.

```
//COMPILE   EXEC PGM=CCNDRVR,REGION=32M,
//   PARM=('NOMAR,DEF(MVS),SOURCE,LIST,INLINE(NOAUTO,
//         REPORT,500,2000),SS,LO,RENT,ILP32')
//STEPLIB   DD   DSN=CEE.SCEERUN,DISP=SHR
//          DD   DSN=CBC.SCBCCMP,DISP=SHR
//SYSLIB    DD   DSN=SYS1.SAPSHDR.H,DISP=SHR
//          DD   DSN=CEE.SCEEH.H,DISP=SHR
//          DD   DSN=CEE.SCEEH.SYS.H,DISP=SHR
//SYSLIN    DD   DSN=PSFMVS.EXIT.OBJECT(APSUC01),DISP=SHR
//SYSPRINT  DD   SYSOUT=*
//SYSOUT    DD   SYSOUT=*
//SYSCPRT   DD   SYSOUT=*
//SYSUT1    DD   UNIT=VIO,SPACE=(32000,(30,30)),
//               DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200)
//SYSUT4    DD   UNIT=VIO,SPACE=(32000,(30,30)),
//               DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200)
//SYSUT5    DD   UNIT=VIO,SPACE=(32000,(30,30)),
//               DCB=(RECFM=FB,LRECL=3200,BLKSIZE=12800)
//SYSUT6    DD   UNIT=VIO,SPACE=(32000,(30,30)),
//               DCB=(RECFM=FB,LRECL=3200,BLKSIZE=12800)
//SYSUT7    DD   UNIT=VIO,SPACE=(32000,(30,30)),
//               DCB=(RECFM=FB,LRECL=3200,BLKSIZE=12800)
//SYSUT8    DD   UNIT=VIO,SPACE=(32000,(30,30)),
//               DCB=(RECFM=FB,LRECL=3200,BLKSIZE=12800)
//SYSUT9    DD   UNIT=VIO,SPACE=(32000,(30,30)),
//               DCB=(RECFM=FB,LRECL=137,BLKSIZE=882)
//SYSUT10   DD   SYSOUT*
//SYSUT14   DD   UNIT=VIO,SPACE=(32000,(30,30)),
//               DCB=(RECFM=FB,LRECL=3200,BLKSIZE=12800)
//SYSUT15   DD   SYSOUT*
//SYSIN     DD   DSN=PSFMVS.EXIT.CSOURCE(APSUC01),DISP=SHR
```

Figure 83. Sample JCL for compiling a C exit

Notes:

1. PSF C exits are only supported in 32-bit mode. Therefore, when you are compiling a C exit, make sure that you use the default ILP32 compile option rather than LP64.

2. During PSF installation, SYS1.SAPSHDR.H is the data set where the source code for the header files that are included with PSF is placed.

3. PSFMVS.EXIT.OBJECT is the data set where the object from the compile is placed.

4. APSUC01 is the name of the C exit that is being compiled.

5. PSFMVS.EXIT.CSOURCE is the name of the data set that contains the source code for the C exit.

6. When PSF is installed, the source code for the sample C exits is placed in SYS1.SAMPLIB.

Prelinking a C exit

Figure 84 on page 261 shows sample JCL statements that can be used as a guideline for prelinking a C exit on z/OS.

```
//PRELINK   EXEC PGM=EDCPRLK,REGION=32M,
//   PARM=('NONCAL')
//STEPLIB   DD   DSN=CEE.SCEERUN,DISP=SHR
//          DD   DSN=CBC.SCBCCMP,DISP=SHR
//SYSMSGS   DD   DSN=CEE.SCEEMSGP(EDCPMSGE),DISP=SHR
//SYSLIB    DD   DSN=CEE.SCEECPP,DISP=SHR
//          DD   DSN=CBC.SCLBCPP,DISP=SHR
//SYSIN     DD   DSN=PSFMVS.EXIT.OBJECT(APSUC01),DISP=SHR
//SYSMOD    DD   DSN=PSFMVS.EXIT.PRELINK(APSUC01),DISP=SHR
//SYSPRINT  DD   SYSOUT=*
//SYSOUT    DD   SYSOUT=*
//*
```

Figure 84. Sample JCL for prelinking a C exit

Notes:

1. PSFMVS.EXIT.OBJECT is the data set that contains the C exit code to be prelinked. The code is placed in this data set by the compile JCL (see Figure 83 on page 261).

2. PSFMVS.EXIT.PRELINK is the data set where the prelinked code from the prelinker is placed.

Link-editing a C exit

Figure 85 on page 262 shows sample JCL statements that can be used as a guideline for link-editing a C exit on z/OS.

```
//STEP1  EXEC PGM=IEWL,
//   PARM='MAP,LET,LIST,XREF,RENT,AMODE=31,RMODE=ANY',
//   REGION=512K
//SYSUT1   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSPRINT DD  SYSOUT=*
//SYSLMOD  DD  DSN=SYS1.PSF.EXITLIB,DISP=SHR
//SYSLIB   DD  DSN=CEE.SCEELKED,DISP=SHR
//SYSLIN   DD  DSN=PSFMVS.EXIT.PRELINK(APSUC01),DISP=SHR
//         DD  *
 ENTRY APSUC01
 NAME APSUC01(R)
/*
```

Figure 85. Sample JCL for link-editing a C exit

Notes:

1. PSFMVS.EXIT.PRELINK is the data set that contains the C exit code to be linked. The code is placed in this data set by the prelink JCL (see Figure 84 on page 261).
2. C exit APSUC01 is link-edited into data set SYS1.PSF.EXITLIB.

Link-editing APSUMACC and APSUTIMC

Figure 86 on page 262 shows sample JCL statements that can be used as a guideline for link-editing APSUMACC and APSUTIMC modules on z/OS. These modules must be link-edited into the same load module as the C exit that uses them.

```
//STEP1  EXEC PGM=IEWL,
//   PARM='MAP,LET,LIST,XREF,RENT,AMODE=31,RMODE=ANY',
//   REGION=512K
//SYSUT1   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSPRINT DD  SYSOUT=*
//SYSLMOD  DD  DSN=SYS1.PSF.EXITLIB,DISP=SHR
//SYSLIB   DD  DSN=CEE.SCEELKED,DISP=SHR
//SYSLIN   DD  DSN=PSFMVS.EXIT.PRELINK(APSUC01),DISP=SHR
//         DD  DSN=PSFMVS.ASSM.TEXT(APSUMACC),DISP=SHR
//         DD  DSN=PSFMVS.ASSM.TEXT(APSUTIMC),DISP=SHR
//         DD  *
 ENTRY APSUC01
 NAME APSUC01(R)
/*
```

Figure 86. Sample JCL for link-editing APSUMACC and APSUTIMC modules

Note: C exit APSUC01, assembler module APSUMACC, and assembler module APSUTIMC are link-edited into load module APSUC01 in SYS1.PSF.EXITLIB.

Changing exit routines

During PSF installation, the assembly-language source code for all the exit routines is installed into SYS1.SAMPLIB. The source code in C language is also installed into SYS1.SAMPLIB for most of the exit routines. You can either view the source routines online or print them. You can then modify, assemble, and install (or link-edit with the replace option) any of these routines to provide your own version of the PSF exits. Table 30 on page 263 lists these routines and summarizes their functions.

Table 30. PSF exit routines in SYS1.SAMPLIB		
Sample Routine		
Assembler Name	**C Name**	**Description**
APSUX01	APSUC01	Source routine for default job headers. See Figure 45 on page 149 for an example.
APSUX01P	APSUC01P	Source routine for job headers that print faster on communication-attached printers, except the 4224 and 4234 printers. See Figure 46 on page 150 for an example.
APSUX01S		Source routine for job headers that use guaranteed print labeling (described in *PSF for z/OS: Security Guide*).
APSUX02	APSUC02	Source routine for default job trailers.
APSUX02P	APSUC02P	Source routine for job trailers that print faster on communication-attached printers, except for the 4224 and 4234 printers.
APSUX02S		Source routine for job trailers that use guaranteed print labeling (described in *PSF for z/OS: Security Guide*).
APSUX03	APSUC03	Source routine for default data set separators.
APSUX04X	APSUC04X	Sample (skeleton) source routine for processing line-data records.
APSUX04Y	APSUC04Y	Sample (skeleton) source routine for processing structured fields.
APSUX05X	APSUC05X	Sample (skeleton) source routine for modifying SMF type 6 record processing.
APSUX06X	APSUC06X	Sample (skeleton) source routine for modifying PSF message processing.
APSUX07Q		Sample source routine for using the JCL FORMS= value to determine whether to inhibit print job finishing for this job.
APSUX07R		Sample source routine for keeping or deleting overlays, page segments, or both at the resource delete call.
APSUX07S	APSUC07S	Sample source routine for loading all page definitions and form definitions from DASD with each use.
APSUX07T	APSUC07T	Sample source routine for mapping a FORMS= value to a PAGEDEF and FORMDEF pair.
APSUX07U	APSUC07U	Sample source routine for altering the member-name prefix of resource objects (permits the use of objects with nonstandard prefixes for production or secure work).
APSUX07V	APSUC07V	Sample source routine for indicating how PSF can do many of the functions that are requested on the initialization and begin-data-set calls.
APSUX07W	APSUC07W	Sample source routine for indicating how PSF can do many of the functions that are requested on the resource access and resource delete calls.
APSUX07X	APSUC07X	Sample (skeleton) source routine that explains exit structure.

Table 30. PSF exit routines in SYS1.SAMPLIB (continued)		
Sample Routine		
Assembler Name	**C Name**	**Description**
APSUX07Y	APSUC07Y	Sample (skeleton) source routine that demonstrates: • Exit involvement, based on printer name and type • Substitution of a page definition or a page segment • Issuance of a message for page segment substitutions • Retention of an overlay over data set boundaries.
APSUX07Z	APSUC07Z	Sample (skeleton) source routine that demonstrates: • Exit involvement based on printer name and type • Substitution of a job-header form definition.
APSUX08X	APSUC08X	Sample source routine for starting Line-Mode Migration.
APSUX14X	APSUC14X	Sample source routine for starting 3800 Line-Mode Conversion in direct-printing mode.
APSUX15M	APSUC15M	Sample source routing that provides an indication of whether the data set is the first, next, last, or only data set in the output group. Download for z/OS or AFP Download Plus sends the information through the **-opa** parameter to the server daemon.
APSUX15X	APSUC15X	Sample source routine to provide the printer device name and the output group name for Download for z/OS or AFP Download Plus to transmit to its server.
APSUX16X	APSUC16X	Sample (skeleton) source routine for the direct printing exit.

Note: PSF provides object code for the default separator-page exits (APSUX01, APSUX02, and APSUX03 modules). PSF does not provide object code for the other exits—logical-record processing, SMF type 6 record processing, PSF-message processing, resource processing, Line-Mode Migration, direct printing 3800 Line-Mode Conversion, print parameters, and direct printing.

During PSF initialization, the exit subcomponent issues a BLDL request to locate any user-provided exit routines that are prefixed with APSU*lnn* (where *l* is X for assembly language or C for C language and *nn* is replaced with the exit numbers: 01, 02, 03, 04, 05, 06, 07, 08, 14, 15, 16).

Note: You must make sure that you use the correct name when you are compiling and link-editing assembler exits or PSF is unable to locate them. For example, for samples APSUX01P and APSUC01P, use the names APSUX01 and APSUC01.

The exit routines can be in a data set pointed to in a STEPLIB DD statement in the PSF startup procedure, or they can be in a LINKLIB in the standard MVS search order. You can provide a different set of exit routines for each PSF address space in your system by using a unique STEPLIB for each PSF startup procedure, or you can provide a single set of exit routines for all PSF-controlled printers in the operating system by placing them in a LINKLIST data set.

Notes:

1. All exit routines must be reenterable and must be link-edited with the RENT,LET attributes.
2. Exit routines must be in an authorized library.
3. Code for installation exits can be written in 24-bit (assembler exits only) or 31-bit addressing mode.
4. PSF program maintenance might require the reassembly of installation exits.

5. All exit routines are run in supervisor state with a protection key of 1. All system services available in supervisor state are available with these exits.
6. The module name (in printable EBCDIC) must be placed in the object code, starting at the sixth byte (for exits written in assembly language). If a dump is created, the module name is printed in the dump. (PSF ESTAE is in effect during the processing of exits.)

Installation exit control blocks

Because the installation exits that PSF supports can be written in either assembly language or C language, the installation exit control blocks must also be provided in both assembly language and C language. During PSF installation, the source code for the assembler control blocks is installed into SYS1.MACLIB and the source code for the corresponding C header files is installed into SYS1.SAPSHDR.H. Table 31 on page 265 lists the assembly language name and the corresponding C header file name for each installation exit control block.

Note: Unless otherwise specified, the assembly name of a control block is used in this publication to represent either the assembly control block or the C control block.

When PSF calls an exit routine, register 1 points to a one-word parameter list that contains the address of the APSGEXTP area. This area contains the address of the APSUECA exit-communication area, pointers to JSPA, and pointers to the input records associated with Exit 4, Exit 6, Exit 7, and Exit 16. When the exit routine returns to PSF, PSF does not check register 15 for a return code.

Table 31. Installation exit control blocks		
Assembler Name	**C Name**	**Description**
APSGEXTP	APSGEXTC	Installation exit parameter area
APSLCRL	APSUCRLC	Concatenated retrieval list
IEFDOKEY	APSUDOKC	Dynamic output key mapping
IEFDOTUM	APSUDOTC	Dynamic output text unit mappings
APSUECA	APSUECAC	Exit-communication area
IAZIDX	APSUIDXC	Index table
IEFJESCT	APSUJESC	Job Entry Subsystem communication table
IEFJMR	APSUJMRC	Job management record
IAZJSPA	APSUJSPC	Job separator page data area
APSUXP16	APSUP16C	Installation Exit 16 parameter area
APSURLST	APSURLSC	Exit 7 resource list
	APSUSEPC	Separator exits header file
	APSUSEP	Separator exits header file for APSUC01 and APSUC02
	APSUSEPP	Separator exits header file for APSUC01P and APSUC02P
IEFSJTRC	APSUSJCC	SJF SWBTUREQ Services return and reason codes
IEFSJTRP	APSUSJTC	SJF SWBTUREQ retrieve and parameter list
IFASMFR	APSUSMFC	SMF records macro

APSGEXTP area

The format of APSGEXTP is shown in Figure 87 on page 266 - Figure 89 on page 268. Values in parentheses are in hexadecimal.

For information about the contents of XTP7, see "XTP7 parameter area format" on page 421.

```
 Offsets  Type      Length  Name          Description

 0    (0) Structure    40  APSGEXTP       PSF-exit-parameter area


 ==============================================================================
 The following fields provide input to the installation exit processor routines.
 ==============================================================================
 0    (0) Address       4  XTPECAP        Exit-communication-area address.
 4    (4) Address       4  XTPJSPAP       Pointer to JES separator page
                                          area of the last data set
                                          (with a separator, might be zero).
 8    (8) Address       4  XTPRECP        Pointer to input record.
12    (C) Signed        4  XTPRECL        Input-record length.


 ==============================================================================
 The following fields are set by the installation exit processor routines to
 provide an interface back to PSF. On input to processors, XTPRIXP points to a
 record index entry. On output, if the processor indicates "write index",
 XTPRIXP must point to the newly created index header.
 ==============================================================================

16   (10) Character     4  *              Reserved.
20   (14) Address       4  XTPRIXP        Pointer to record table index.
24   (18) Unsigned      1  XTPPIND        Processing indicator.
25   (19) Bitstring     1  XTP4FLAG       Exit 4 or Exit 14 flags.
      1...  ....           XTP4NTY        PSF will send end-of-transmission
                                          (EOT) indicator to Exit 4.
      .1..  ....           XTP4NEOT       Normal EOT has occurred.
      ..1.  ....           XTP4ABNE       Abnormal EOT has occurred.
      ...1  ....           XTP4NCAL       Call Exit 4 one time.
                                          When this flag is set ON, Exit 4 will
                                          not be called again until EOT or the
                                          next copy of the data set or the next
                                          data set. This flag is reset when
                                          Exit 4 is called for the first record
                                          of the next copy of the data set or
                                          the next data set.
      ....  1...           XTP4CONV       Do line-mode conversion. When
                                          this flag is set on, 3800 Line-
                                          Mode Conversion is activated.
                                          This flag is reset when
                                          Exit 4/14 is called for the
                                          first record of the
                                          transmission.
      ....  .1..           XTP4WINA       Wide or narrow paper flag. This
                                          flag is only used when XTP4CONV
                                          is set on (3800 Line-Mode
                                          Conversion is active) and the
                                          printer is in cut-sheet
                                          emulation mode. When this flag
                                          is set on, landscape pages are
                                          generated for printing in cut-
                                          sheet emulation mode. When this
                                          flag is set off, portrait pages
                                          are generated for printing in
                                          cut-sheet emulation mode. This
                                          flag is reset when Exit 4/14 is
                                          called for the first record of
                                          the transmission (the next copy
                                          of the data set or the next
                                          data set).
```

Figure 87. Format of the APSGEXTP area (Part 1 of 3)

```
        ....  ..1.            XTP4LMRG        Line merge flag. This flag is
                                              only used when XTP4CONV is set
                                              on (3800 Line-Mode Conversion is
                                              active). When this flag is set
                                              on, PSF performs the line merge
                                              function like the 3800 printer
                                              in "line mode."
                                              This flag is reset when Exit
                                              4/14 is called for the first
                                              record of the transmission.
        ....  ...1            *               Reserved.
26  (1A) Character     4  *                   Reserved.
30  (1E) Bitstring     1  XTP5FLAG            Exit 5 SMF 6 flags.
        1...  ....            XTP5NEWS        Processing JESNEWS data set.
        .111  1111            *               Reserved.
31  (1F) Unsigned      1  XTP4DPLX            Duplex value. This field is
                                              only valid when XTP4CONV is set on
                                              (3800 Line-Mode Conversion is active).
                                               0 = Not specified (defaults to
                                                   formdef value)
                                               1 = Simplex
                                               2 = Normal duplex
                                               3 = Tumble duplex
                                              This field is set to zero when
                                              Exit 4/14 is called for the
                                              first record of the transmission.
32  (20) Unsigned      2  XTP4XOFF            Set horizontal offset-
                                              pels at 240 pels/inch.
                                              Valid only when
                                              XTP4CONV is set ON.
34  (22) Unsigned      2  XTP4YOFF            Set vertical offset-
                                              pels at 240 pels/inch.
                                              Valid only when
                                              XTP4CONV is set ON.
36  (24) Character     4  *                   Reserved.

===============================================================================
The following fields provide input to the installation Exit 6 (message exit)
===============================================================================
 0   (0) Structure    24 XTP6                 Exit 6 input record.
 0   (0) Character     8  XTP6PROC            FSA identifier:
                                              The label of the PRINTDEV statement.
 8   (8) Character     8  XTP6MSG             Message ID (APSXXXXI).
16  (10) Character     1  XTP6TYPE            Message TYPE.
                                              W=WTO:  Most WTO messages
                                              are asynchronous.  Async
                                              messages belong to the data
                                              set on the printer, which may
                                              be different from the one
                                              being processed by PSF.
                                              XTPJSPAP points to the JSPA
                                              related to the data set
                                              currently being processed by PSF.
17  (11) Bitstring     1  XTP6FLAG            Exit 6 indicators
        1...  ....            XTP6INTP        Message is targeted for the
                                              interrupt message page.
        .111  1111            *               Reserved.
```

Figure 88. Format of the APSGEXTP area (Part 2 of 3)

```
18   (12) Character     1  XTP6FLG1        Exit 6 processing indicators.
      1... ....            XTP6TFSA        Terminate the FSA. PSF will
                                           complete the processing of the
                                           current data set (the one
                                           being processed by PSF, which
                                           may be different from the one
                                           on the printer), and terminate
                                           the FSA.
      .1.. ....            XTP6HOLD        Release the data set currently
                                           being processed by PSF to JES,
                                           and mark the data set as
                                           unprintable.
      ..1. ....            XTP6TJOB        Terminate the job associated with
                                           the interrupt message page.
      ...1 1111            *               Reserved.
19   (13) Bitstring     1  XTP6ACT         Action to be taken by PSF:
                                           Set by PSF on input;
                                           can be altered by the exit.
                                           If XTP6ACT is set to 0 by the
                                           exit, the message is
                                           suppressed.
      1... ....            XTP6WTO         Put msg to op. console.
      .1.. ....            XTP6SEC         Put msg to sec. console.
      ..1. ....            XTP6PAP         Put msg to paper (msg data set).
      ...1 1111            *               Reserved.
20   (14) Character     4  *               Reserved.

==============================================================================
The fields for installation Exit 7 (DSECT) are found in the appendix for
"Field Descriptions for Exit 7".
==============================================================================
 0   (0) Structure    233  XTP7            Exit 7 input record.

==============================================================================
The following fields are used for installation Exit 8 (Line-Mode Migration)
==============================================================================
 0   (0) Structure     10  XTP8            Exit 8 input record.
 0   (0) Character      1  XTP8FLAG        Exit 8 flags.
      1... ....            XTP8MIG         Do Line-Mode Migration for a 4245
                                           FCB flag. When this flag is set on,
                                           Line-Mode Migration is activated
                                           for a 4245 FCB (FCB2). This flag is
                                           reset when Exit 8 is called.
      .1.. ....            XTP8MIG4        Do Line-Mode Migration for a 4248
                                           FCB flag. When this flag is set on,
                                           Line-Mode Migration is activated
                                           for a 4248 FCB (FCB4). This flag is
                                           reset when Exit 8 is called.
      ..1. ....            XTP8PBSP        Honor page boundary spacing. When
                                           this flag is set on, page boundary
                                           spacing is honored in Line-Mode
                                           Migration. This flag is reset when
                                           Exit 8 is called.
      ...1 1111            *               Reserved.
 1   (1) Character      9  *               Reserved.
```

Figure 89. Format of the APSGEXTP area (Part 3 of 3)

Exit-communication areas: APSUECA and APSUECE

APSGEXTP provides addressability to the exit-communication area (APSUECA) and to its extension (APSUECE). These areas provide the interface among the exit routines.

The format of APSUECA is shown in - . Values that are enclosed in parentheses are in hexadecimal.

```
   Offsets  Type      Length  Name            Description
   0    (0) Structure    *    APSUECA         Exit-Communication Area.
   0    (0) Character    *    ECABASE         Base section of ECA.
   0    (0) Character    4    ECAID           Control-block ID set to the
                                              character string UECA.
   4    (4) Signed       4    ECALEN          Length of exit-communication area.
   8    (8) Character    6    ECAPTF          Relative PTF date.
  14    (E) Character    2    ECAREL          Release level.
  16   (10) Signed       4    ECARCODE        RRB PUT return code.
  20   (14) Address      4    ECAUCOMP        Address of exit APSUCOM table.
  24   (18) Address      4    ECAPUTP         Address of exit PUT routine.
  28   (1C) Address      4    ECABLKP         Address of exit block-letter routine.
  32   (20) Address      4    ECASABP         Address of APSPFSAB. (PSF usage only.)
  36   (24) Address      4    ECAECEP         Address of exit-communication
                                              extension.
  40   (28) Address      4    ECAMARP         3800 -3/8 migration aid
                                              pointer.
  44   (2C) Character    4    ECAFORM         Form name in output JCL
                                              (FORMS= ).
  48   (30) Bitstring    1    ECAMFCS         Mark form carrier strip flag.
  49   (31) Character    6    ECARNDM         Separator page random number.
  55   (37) Character    1    ECAFLAG         ECA flags.
       1...  ....             ECAMFTRL        Mark forms inhibit flag for
                                              job trailer.
                                              0 = job trailers may be
                                              marked.
                                              1 = do not mark job trailers.
=================================================================================
The combination of these ECALSTDS and ECAFSTDS flags and their meanings are:
    00 = Middle data set (not first or last data set)
    01 = First data set
    10 = Last data set
    11 = Only data set
=================================================================================
       .1..  ....             ECALSTDS        Last data set.
                                              0 = Not the last data set of the job.
                                              1 = Last data set of job.
       ..1.  ....             ECAFSTDS        First data set.
                                              0 = Not the first data set of the job.
                                              1 = First data set of the job.
       ...1  ....             ECACUTSH        Cut-sheet output.
                                              0 = Not cut-sheet output.
                                              1 = Cut-sheet output - Either cut-
                                                  sheet printer or continuous
                                                  forms printer with BTS.
       ....  1...             ECADUPEN        Duplex enabled.
                                              0 = Duplex not enabled.
                                              1 = Duplex enabled - Duplex available
                                                  from at least one media source.
       ....  .111      *                      Reserved.
```

Figure 90. Format of exit-communication area, APSUECA (Part 1 of 2)

If you do not want PSF to generate separator pages, delete or modify the separator-page exits. For more information, see .

```
56   (38) Address      4  ECARECAD    Address of a record to be put
                                      to PSF from an exit processor.
60   (3C) Signed       4  ECARECLN    Length of exit record.
64   (40) Bitstring    1  ECADRF      Record description flags.
     1... ....            ECADSR      Indicates data-stream record.
     .1.. ....            ECALMR      Indicates line-mode record.
     ..1. ....            ECAANSI     Record contains ANSI control
                                      characters.
     ...1 ....            ECAMACH     Record contains machine
                                      control characters.
     .... 1...            *           Reserved (PSF use only).
     .... .1..            *           Reserved (PSF use only).
     .... ..1.            *           Reserved (PSF use only).
     .... ...1            ECAOPJ      OPTCD=J used for record.
65   (41) Signed       1  ECAPTYPE    Product type of call:
                                      1 = PSF (ECAPSF)
                                      2 = Download (ECADOWN)
                                      3 = AFP Download Plus (ECAAFPDP)
66   (42) Signed       2  ECATRNC     Transmission count.
68   (44) Bitstring    1  ECAFLAGS    Exit-control flags.
     1... ....            ECASLANT    Slant letters requested.
     .1.. ....            ECALJUST    Left-align block letter
                                      lines.
     ..1. ....            ECAPBLK     Performance-block letters.
     ...1 1111            *           Reserved.
69   (45) Bitstring    1  *           Reserved.
70   (46) Signed       2  ECAGWRK     General-use work field.
72   (48) Character    8  ECABLKIN    Input to block letter routine.
80   (50) Character   72  ECAUSAVE    Exit processor's save area.
152  (98) Character   72  ECAGSAVE    General-use exit save area.
224  (E0) Character   72  ECABSAVE    24-bit interface return save area.
296  (128) Character  72  ECAMSGSV    Exit 6 (message) exit save area.
368  (170) Character  72  ECARSCSV    Exit 7 (resource) exit save area.
440  (1B8) Character  72  ECACSAVE    C exit routine save area.
512  (200) Character   *  ECAWKBUF    The work buffer (ECAWKBUF)
                                      length depends on a field in
                                      APSUCOM. Minimum length of
                                      the ECA, including the work
                                      buffer, is 2039 bytes.The
                                      The minimum length of the
                                      work buffer is 1599 bytes.
```

Figure 91. Format of exit-communication area, APSUECA (Part 2 of 2)

The format of APSUECE is shown in <u>Figure 92 on page 271</u> - <u>Figure 93 on page 272</u>. Values that are enclosed in parentheses are in hexadecimal.

```
  Offsets   Type      Length   Name          Description

   0    (0) Character     *    APSUECE       Exit extension.
   0    (0) Character     4    ECEID         Control block ID set 'UECE'.
   4    (4) Signed        4    ECELEN        Length of ECE.
   8    (8) Character     6    ECEPTF        Relative PTF date.
  14    (E) Character     2    ECEREL        Release level.
  16   (10) Address       4    ECEFECAP      Addr of FSA ECA, used by
                                             Exits 1-4, 6, 7, 8, 14, 15, 16.
  20   (14) Address       4    ECENECAP      Addr of NST ECA, used by Exit 5.
  24   (18) Character     8    ECEPRMD       PRMODE defined in JCL output.
  32   (20) Character     4    ECEFLASH      Flash defined in JCL output.
  36   (24) Character     8    ECEDPDEF      PAGEDEF defined in PRINTDEV.
  44   (2C) Character     8    ECEDFDEF      FORMDEF defined in PRINTDEV.
  52   (34) Address       4    ECEFSSWK      FSS work area printer.
  56   (38) Character     8    ECEPDEVK      PRINTDEV SWB token.
  64   (40) Character     8    ECEOUTTK      Print data set SWB token.
  72   (48) Character     8    ECEJDVTN      Print data set JDVT name.
  80   (50) Character     2    ECEPRTT       Printer type (eg. X'3800')
                                             that the device is emulating
                                             (from the IPDS STM).
  82   (52) Character     1    ECEPRTM       Printer model number (e. g.
                                             X'03') that the device is
                                             emulating (from the IPDS STM).
  83   (53) Bitstring     1    ECEFLAG       Exit flags.
       1... ....               ECEJESS       JES supports Enhanced SYSOUT
                                             Support (ESS) JCL.
       .1.. ....               ECERPM        RPM is attached.
       ..1. ....               ECEDPF        DPF is attached.
       ...1 ....               ECEPSFD       PSF direct is attached.
       .... 1...               ECEWPM        WPM is attached.
       .... .111               *             Reserved.
  84   (54) Character     8    ECEFORM       Form name in output JCL (FORMS= ).
  92   (5C) Character     4    ECEPRTDT      Printer device type, stamped
                                             on the printer in EBCDIC
                                             (from the IPDS OPC, or IPDS
                                             STM if none in the OPC).
  96   (60) Character     3    ECEPRTMD      Printer model number, stamped on
                                             the printer in EBCDIC (from the
                                             IPDS OPC, or IPDS STM if not in
                                             the OPC).
  99   (63) Character     1    *             Reserved.
 100   (64) Address       4    ECEBTBL       Address of block letter tables.
                                             Do not modify ECEBTBL; it is used
                                             by APSUBLK.
 104   (68) Signed        4    ECEJOBLP      Logical pages in job.
 108   (6C) Signed        4    ECEJOBIM      Impressions (sides) in job.
 112   (70) Signed        4    ECEJOBPP      Physical sheets in job (includes
                                             header, message, and trailer sheets).
 116   (74) Address       4    ECEWKPTR      Address of exit work area.
 120   (78) Signed        4    ECEWKLEN      Length of exit work area.
 124   (7C) Address       4    ECETUPTR      SWBTU address.
 128   (80) Signed        4    ECETULEN      SWBTU length.
 132   (84) Signed        4    ECEREC        Logical record count (doesn't include
                                             header or trailer page records).
 136   (88) Signed        4    ECELPCNT      Logical page count for the data set
                                             (includes header and trailer pages).
 140   (8C) Signed        4    ECEPPCNT      Physical sheet count for the data set
                                             (includes header and trailer sheets).
 144   (90) Signed        4    ECEDSCNT      Physical duplex sheet count (includes
                                             header and trailer sheets).
 148   (94) Signed        4    ECESSCNT      Physical simplex sheet count (includes
                                             header and trailer sheets).
 152   (98) Signed        4    ECENUP1C      Physical N_UP 1 sheet count (includes
                                             header and trailer sheets).
```

Figure 92. Format of exit-communication area extension, APSUECE (Part 1 of 2)

```
156   (9C) Signed      4  ECENUP2C     Physical N_UP 2 sheet count (includes
                                        header and trailer sheets).
160   (A0) Signed      4  ECENUP3C     Physical N_UP 3 sheet count (includes
                                        header and trailer sheets).
164   (A4) Signed      4  ECENUP4C     Physical N_UP 4 sheet count (includes
                                        header and trailer sheets).
168   (A8) Signed      2  ECESCPY      Separator copy count.
170   (AA) Signed      2  ECETXMS      Total transmission required.
================================================================================
The following addresses are valid only for assembler exits.
================================================================================
172   (AC) Address     4  ECEX01P      Address of Exit 1.
176   (B0) Address     4  ECEX02P      Address of Exit 2.
180   (B4) Address     4  ECEX03P      Address of Exit 3.
184   (B8) Address     4  ECEX04P      Address of Exit 4.
188   (BC) Address     4  ECEX05P      Address of Exit 5.
192   (C0) Address     4  ECEX06P      Address of Exit 6.
196   (C4) Address     4  ECEX07P      Address of Exit 7.
200   (C8) Address     4  ECEX08P      Address of Exit 8.
204   (CC) Address     4  ECEX09P      Reserved.
208   (D0) Address     4  ECEX10P      Reserved.
212   (D4) Address     4  ECEX11P      Reserved.
216   (D8) Address     4  ECEX12P      Reserved.
220   (DC) Address     4  ECEX13P      Reserved.
224   (E0) Address     4  ECEX14P      Address of Exit 14
                                        (in direct print mode only).
228   (E4) Address     4  ECEX15P      Address of Exit 15.
232   (E8) Address     4  ECEX16P      Address of Exit 16
                                        (in direct print mode only).
236   (EC) Signed      4  ECEIMPS      Impressions (sides)
                                        printed for the data set.
240   (F0) Character   12 ECEDSID      FSI GETDS data set ID.
252   (FC) Address     4  ECELMO       Reserved for LMO.
256  (100) Address     4  ECEPUTC      Address of APSUPUTC.
260  (104) Address     4  ECEBLKC      Address of APSUBLKC.
264  (108) Character   8  ECEFSSNM     FSS name (writer proc name).
272  (110) Character   4  ECEFCB       FCB name in OUTPUT JCL.
276  (114) Character   12 *            Reserved.
```

Figure 93. Format of exit-communication area extension, APSUECE (Part 2 of 2)

If repositioning is required, PSF can enter the exit routines for header pages (Exit 1 and Exit 3) more than once, but the header pages might not reprint unless the repositioning calls for the header pages to reprint. When PSF enters either Exit 1 or Exit 3, PSF sets ECATRNC to the current transmission count, which can be used to determine whether the exit is reentered. An installation exit must not attempt to skip pages or duplicate pages if the exits are reentered; such an attempt might interfere when PSF is repositioning for error recovery.

The following fields are described in more detail:

ECEFSSWK

Address of 32 bytes of functional subsystem (FSS) storage.

This address is passed to all exits in the same address space. When multiple functional subsystem applications (FSAs) are running in an FSS, the same address is passed to all the FSAs.

The exits can use this storage to communicate among different FSAs or to save data between FSA sessions. This storage is not freed until the FSS is stopped.

The FSS storage is for the exclusive use of the exits. PSF does not read or modify this storage.

The exit must serialize the use of the FSS storage because more than one FSA subtask (or an FSA and the NST subtask) can be active at the same time in the FSS address space.

ECEOUTTK

The scheduler work block (SWB) token for the OUTPUT JCL statement of the print data set.

PSF uses ECEOUTTK to retrieve the keywords that are specified in the OUTPUT JCL statement of the print data set.

JES passes this token to PSF in the functional subsystem interface (FSA) GETDS parameter list (IAZFSIP).

For more information about SWBs and how to use the SWB token to retrieve keywords, see *z/OS MVS Using the Functional Subsystem Interface*.

Note: The SWB contains JCL values that are merged from the DD statement, the OUTPUT statement, and the JES defaults according to the resource hierarchy. JES merges the values before it allocates the print data set to PSF. Therefore, when the JCL values are returned, no way exists to determine whether a particular value was specified on the print data set or defaulted to by JES.

ECEPRTT

ECEPRTT contains the printer device type that the printer is functionally emulating, as provided in the IPDS Sense Type and Model command. The functional emulation extends only as far as IPDS deviations and PSF software dependencies for the support of certain functions. Not all of the functions that are supported are identical.

ECEPRTM

ECEPRTM contains the printer model that the printer is functionally emulating, as provided in the IPDS Sense Type & Model command. The functional emulation extends only as far as IPDS deviations and PSF software dependencies for the support of certain functions. Not all of the functions that are supported are identical.

ECEPRTDT

ECEPRTDT contains the printer device type, as provided in the IPDS Obtain Printer Characteristics Product Identifier Self-Defining Field; that field is in the XOA-OPC command. If the printer does not contain a device type in that field, ECEPRTDT matches ECEPRTT, and ECEPRTMD matches ECEPRTM.

ECEPRTMD

ECEPRTMD contains the printer model, as provided in the IPDS Obtain Printer Characteristics Product Identifier Self-Defining Field. If the printer does not contain a model in that field, ECEPRTDT matches ECEPRTT, and ECEPRTMD matches ECEPRTM.

ECEWKPTR

The address of a local work area for the PSF and AFP Download Plus exits.

ECEWKPTR can be used as a local work area by installation-written exits. The work area length is 1024 bytes.

The separator exits supplied by PSF exits use this area. Therefore, if those exits are used, this area must not be used to retain data across exit invocations.

ECETUPTR

The address of the SWBTU for the print data set's OUTPUT JCL statement.

You can use the SWBTUREQ macro to obtain the JCL parameters that are specified for the print data set. The SWBTU address is input to the SWBTUREQ macro. ECETULEN contains the SWBTU length.

The separator exits that are supplied by PSF contain an example of how to start the SWBTUREQ macro.

For more information about the SWBTUREQ macro, see *z/OS MVS Programming: Authorized Assembler Services Reference SET-WTO*.

Note: The SWBTU contains JCL values that are merged from the DD statement, the OUTPUT statement, and the JES defaults according to the resource hierarchy. JES merges these values before it allocates the print data set to PSF. Therefore, when the JCL values are returned, no way exists to determine whether a particular value was specified on the print data set or defaulted to by JES.

ECELPCNT

The logical page count for the data set, including header and trailer pages. The header page is added to the count after APSUX01 is returned to PSF while the trailer page is added after APSUX02 is returned.

ECEPPCNT

The physical sheet count for the data set, including header and trailer sheets. The header sheet is added to the count after APSUX01 is returned to PSF while the trailer sheet is added after APSUX02 is returned.

ECEDSCNT

The physical duplex sheet count, including header and trailer sheets. The header sheet is added to the count after APSUX01 is returned to PSF while the trailer sheet is added after APSUX02 is returned.

ECESSCNT

The physical simplex sheet count, including header and trailer sheets. The header sheet is added to the count after APSUX01 is returned to PSF while the trailer sheet is added after APSUX02 is returned.

ECENUP1C

The physical N_UP 1 sheet count, including header and trailer sheets. The header sheet is added to the count after APSUX01 is returned to PSF while the trailer sheet is added after APSUX02 is returned.

ECENUP2C

The physical N_UP 2 sheet count, including header and trailer sheets. The header sheet is added to the count after APSUX01 is returned to PSF while the trailer sheet is added after APSUX02 is returned.

ECENUP3C

The physical N_UP 3 sheet count, including header and trailer sheets. The header sheet is added to the count after APSUX01 is returned to PSF while the trailer sheet is added after APSUX02 is returned.

ECENUP4C

The physical N_UP 4 sheet count, including header and trailer sheets. The header sheet is added to the count after APSUX01 is returned to PSF while the trailer sheet is added after APSUX02 is returned.

Sequence of exit invocation

This is a typical flow of exit control in the sequence of exit interpretation:

1. Exit 1: Job-header page
2. Exit 3: Data set separator page
3. Exit 4: Examine first spool record
4. Exit 4: Examine next spool record (until end of data set)
5. Exit 4: End-of-data-set call
6. Exit 2: Job-trailer page
7. Exit 5: Write SMF type 6 record

Note: After the trailer page prints, calls to Exit 5 are asynchronous. Calls to Exit 6 might be asynchronous also.

Control flow with Exit 7

The following list is a typical flow of exit control that includes the use of Exit 7 for resource management for a single data set job:

1. Exit 7: Initialization call
2. Exit 7: Access and load calls for the default page definition and form definition
3. Exit 7: Access and load calls for the job header page definition and form definition
4. Exit 7: Access and load calls for the job trailer page definition and form definition
5. Exit 7: Access and load calls for the data set header page definition and form definition
6. Exit 7: Access and load calls for the message data set page definition and form definition
7. Exit 7: Begin-data-set call
8. Exit 1: Job-header page
9. Exit 7: Access and load calls for resources that are used on the job-header page

10. Exit 3: Data set separator page

11. Exit 7: Access and load calls for resources that are used on the data set separator page

12. Exit 4: Examine first spool record

13. Exit 7: Resource-access call for page definition

14. Exit 7: Resource-access call for form definition

15. Exit 7: Access and load calls for resources that are identified in the page definition.

16. Exit 7: Access and load calls for resources that are identified in the form definition.

17. Exit 4: Examine next spool record (until end of data set)

18. Exit 7: Access and load calls for resources that are identified in the input data stream

19. Exit 4: End-of-data-set call

20. Exit 2: Job-trailer page

21. Exit 7: Access and load calls for resources that are used on the job-trailer page

22. Exit 7: Resource-delete calls

23. Exit 7: End-data-set call

24. Exit 5: Write SMF type 6 record

Note: After the trailer page prints, calls to Exit 5 are asynchronous. Calls to Exit 6 might also be asynchronous.

The following list is a typical flow of exit control that includes the use of Exit 7 for resource management for a multiple data set job (two data sets):

1. Exit 7: Initialization call

2. Exit 7: Access and load calls for the default page definition and form definition

3. Exit 7: Access and load calls for the job header page definition and form definition

4. Exit 7: Access and load calls for the job trailer page definition and form definition

5. Exit 7: Access and load calls for the data set header page definition and form definition

6. Exit 7: Access and load calls for the message data set page definition and form definition

7. Exit 7: Begin-data-set call

8. Exit 1: Job-header page

9. Exit 7: Access and load calls for resources that are used on the job-header page

10. Exit 3: Data set separator page

11. Exit 7: Access and load calls for resources that are used on the data set separator page

12. Exit 4: Examine first spool record

13. Exit 7: Resource-access call for page definition

14. Exit 7: Resource-access call for form definition

15. Exit 7: Access and load calls for resources that are identified in the page definition.

16. Exit 7: Access and load calls for resources that are identified in the form definition.

17. Exit 4: Examine next spool record (until end of data set)

18. Exit 7: Access and load calls for resources that are identified in the input data stream

19. Exit 4: End-of-data-set call

20. Exit 7: Resource-delete calls

21. Exit 7: End-data-set call

22. Exit 5: Write SMF type 6 record

23. Exit 7: Begin-data-set call

24. Exit 3: Data set separator page

25. Exit 7: Access and load calls for resources that are used on the data set separator page
26. Exit 4: Examine first spool record
27. Exit 7: Resource-access call for page definition
28. Exit 7: Resource-access call for form definition
29. Exit 7: Access and load calls for resources that are identified in the page definition.
30. Exit 7: Access and load calls for resources that are identified in the form definition.
31. Exit 4: Examine next spool record (until end of data set)
32. Exit 7: Access and load calls for resources that are identified in the input data stream
33. Exit 4: End-of-data-set call
34. Exit 2: Job-trailer page
35. Exit 7: Access and load calls for resources that are used on the job-trailer page
36. Exit 7: Resource-delete calls
37. Exit 7: End-data-set call
38. Exit 5: Write SMF type 6 record

Note: After the trailer page prints, calls to Exit 5 are asynchronous. Calls to Exit 6 might also be asynchronous.

Separator-page exits: Exit 1, Exit 2, and Exit 3

PSF provides modules that create default separator pages of three different types: job headers, job trailers, and data set headers. As is shown in Figure 45 on page 149, Figure 47 on page 151, and Figure 48 on page 152, these separator pages are similar in format to the JES separator pages. Also, if data set printing is interrupted and then resumes, PSF provides a job-continuation page to show where printing continues. The formats of the job-header page and the job-continuation page are identical, except that the continuation page prints CONT at the bottom (instead of START).

Three exits are available for creating separator pages:

APSUX01/APSUC01
Job-header pages (including job-continuation pages)

APSUX02/APSUC02
Job-trailer pages

APSUX03/APSUC03
Data-set-header pages

PSF also provides assembly-language and C-language source routines in SYS1.SAMPLIB that you can modify, assemble, and install (or link-edit with the replace option) to create your own separator-page exit modules. These routines, which are listed in Table 30 on page 263, run in the same address space as PSF.

Input for separator-page exits

When a separator-page exit is entered, it contains these:

Register 1
Pointer to the address of parameter area APSGEXTP

Register 13
Pointer to caller's save area

Register 14
Return address

Register 15
Entry-point address

Output for separator-page exits

When control returns from the exit routine to PSF, the contents of the registers are the same as they were at entry to the exit routine. As is explained in "Service routines" on page 281, if the exit-PUT service routine (APSUPUT) recognizes an error condition during record processing, the ECA return code (ECARCODE) is set. If this field is not zero, PSF issues message APS950I.

Considerations for separator-page exits with multiple pages

If data set printing is interrupted and PSF repositions to a separator page when printing resumes, PSF calls the separator-page exit again. The call to the separator-page exit causes all separator pages to be printed. For example, if your job-trailer exit produces three pages and a paper jam occurs on the third page of the trailer, PSF reprints the first and second trailer page, in addition to the third trailer page.

Processing multiple-page job trailers with forms marking

For details about processing multiple-page job trailers with forms marking, see "Forms marking of multiple-page job trailers" on page 155.

Programming with separator-page exits

These considerations apply to the separator-page exits:

- When a separator-page exit routine is entered, register 1 points to a one-word parameter list with the address of the APSGEXTP area. APSGEXTP contains pointers to the exit-communication area (APSUECA) and to the JES separator-page area (IAZJSPA).

  APSUECA contains pointers to:

  - APSUCOM, the exit-communication table
  - APSUBLK, the block-letter service routine
  - APSUPUT, the exit-put service routine
  - APSUECE, the exit-communication area extension

  IAZJSPA provides information to produce JES separator pages—job headers, job trailers, and data set headers. You can use IAZJSPA to duplicate this information for your own separator-page exit routines. Figure 94 on page 279 shows the format of IAZJSPA.

- When a separator-page exit routine is entered, register 13 points to an 18-word save area that can save the caller's registers. The APSUECA control block contains a save area, ECAUSAVE, for the exit routine.

- Use register 12 as a base register for exit routines. If register 12 points to the beginning of an exit routine, and the module name begins in the sixth byte of the routine, the module name appears in the SDUMP title page if an abend occurs.

- Service routines APSUBLK and APSUPUT require that register 5 contain the address of APSUECA. For details on the interface to these routines, see "Service routines" on page 281.

- Before you call APSUBLK, set the ECAPBLK field of APSUECA if you want to generate an optional (faster-running) separator page. For more information, see "Block-letter routine, APSUBLK" on page 282.

  If you use an optional separator page, the job-separator-page OUTPUT statement in the PSF startup procedure must specify a page definition with a page size no larger than 7⅛ × 7 inches and a font no larger than 12 pitch. [13]

  Note: A 12-pitch font produces text that is centered on the separator page.

- If you are using PSF assembler samples for Exit 1, Exit 2, or Exit 3, the line length is set to 127 characters. If you decide to use slanted block letters (ECASLANT), you might lose the last few characters of your last block letter. To prevent this from happening, you can left-align the block letters

---

[13] Pitch refers to character width and is based on the number of characters that can be placed in a linear inch. For example, a 12-pitch font prints 12 characters per inch.

(ECALJUST), use the smaller performance block letters (ECAPBLK), or increase the line length to 132 characters.

If you are using the C samples for Exit 1, Exit 2, or Exit 3, the line length is set to 132 characters; therefore, you can use slanted block letters without losing the last few characters of the last block letter.

- As is explained in "Exit table, APSUCOM" on page 279, the APSXUCOM macro provides a table for converting the Julian date to a year/month/day format (for example, 99 Jun 03).

- As is explained in "Block-letter tables, APSUBTBL" on page 280, APSUBTBL provides the translate tables for the block letters created by APSUBLK for the separator pages.

- The separator-page lines can be either line-format records or structured fields. These lines can be composed of information obtained from the following control-block fields:
  - JSPAJBNM for job name
  - JSPAJBID for job number
  - JSPJSOCL for SYSOUT class
  - JSPJGRP1 for output-group identifier 1
  - JSPJGRP2 for output-group identifier 2
  - JSPJGRPN for output-group name
  - JSPJGRPD for output-group destination name
  - JSPJPNAM for programmer name
  - JSPJRMNO for room number
  - JSPADEVN for device name
  - JMRCPUID for system name

  You can also use these additional informational fields for the data-set-header exit routine:
  - JSPJDSSN for step name
  - JSPJDSDD for data set DD name
  - JSPJSOCL for SYSOUT class
  - JSPJPRIO for data set priority
  - JSPJDSPN for procedure name

JES job-separator-page area—IAZJSPA

IAZJSPA (if written in assembly-language code) or APSUJSPC (if written in C-language code) contains job-level information for a data set that generates JES separator pages—job headers, job trailers, and data set headers. You can use IAZJSPA to duplicate this information for your own separator-page exit routines. If the separator-page exit requires information that is not provided in IAZJSPA, the information might be available in a JES exit.

A pointer to the modified IAZJSPA is passed to the PSF separator-page exit. To modify IAZJSPA in JES2, use Exit 23; in JES3, use Exit IATUX45.

Figure 94 on page 279 shows the format of IAZJSPA. Values inside parentheses are in hexadecimal.

```
Offsets    Length        Name           Description
  0 (0)        4          JSPAID         Control block identifier set
                                          to JSPA
  4 (4)        2          JSPALEN        Length of JSPA
  6 (6)        1          JSPAFLG1       Flags
  7 (7)        1          *              Reserved
  8 (8)        8          JSPAJBNM       Job name
 16 (10)       8          JSPAJBID       Job identifier
 24 (18)       8          JSPADEVN       Device name
 32 (20)       4          JSPADEVA       Device address
 36 (24)       4          JSPAJMR        JMR address
 40 (28)       8          JSPJGRPN       Output-group name
 48 (30)       2          JSPJGRP1       Output-group identifier 1
 50 (32)       2          JSPJGRP2       Output-group identifier 2
 52 (34)       8          JSPJGRPD       Output-group destination name
 60 (38)       4          JSPJRMNO       Room routing number
 64 (40)      20          JSPJPNAM       Programmer's name
 84 (54)      24          JSPJDSNM       Data set name, fully qualified
               8          JSPJDSPN       Data set procedure name
               8          JSPJDSSN       Data set step name
               8          JSPJDSDD       Data set DD name
108 (6C)       1          JSPJSOCL       SYSOUT class
109 (6D)       1          JSPJPRIO       Data set priority
112 (70)       4          JSPAUSR1       Reserved for user
116 (74)       4          JSPAUSR2       Reserved for user
```

Figure 94. Format of JES job-separator-page area, IAZJSPA

JES determines the fields, starting with JSPJGRPN at offset 40 (X'28'). The JSPA is mapped by the IAZJSPA control block. For more information about IAZJSPA, see your JES publications.

shows the format of a job-separator-page area extension, JSPEXT. Values inside parentheses are in hexadecimal. JSPEXT is mapped by the IAZJSPA control block. For more information about IAZJSPA, see your JES publications.

```
Offsets   Length  Name           Description
  0    (0)     2   JSPEXNUM       Number of extensions
  2    (2)     2   JSPEXLEN       Length of extensions
  4    (4)    16   *              Reserved
 20   (14)     2   JSPCELEN       Length of common extension
 22   (16)     2   JSPCEVSN       Version number
 24   (18)     4   JSPCECID       Common extension identifier
 28   (1C)     8   JSPCEUID       User ID
 36   (24)     8   JSPCESEC       Security label
 44   (2C)    53   JSPCEDSN       Data set resource name
 97   (61)     3   *              Reserved
100   (64)     4   JSPCESEG       Segment ID
```

Figure 95. Format of JES job-separator-page area extension, JSPEXT in IAZJSPA

Exit table, APSUCOM

The exit table, APSUCOM, contains:

- A 4-byte length field
- A table for converting the Julian date to a year/month/day format

The length field represents the length of the exit-communication area, APSUECA. It is built once during the initialization of PSF. Increasing the value that is given for this length field increases the work buffer at the end of APSUECA. The work buffer can store information that is passed from exit to exit for use by several PSF installation-exit routines.

The APSXUCOM macro can be used to create a new APSUCOM exit table. Not only can you change the length field, but you can also set input parameters to change the three-character month names in the Julian conversion table.

Note: As is explained in "Block-letter routine, APSUBLK" on page 282, APSUBLK passes block letters to the exit in the work buffer, ECAWKBUF. To prevent APSUBLK from overwriting data, expand the buffer and pass the data in the expanded portion of ECAWKBUF.

To create a new APSUCOM exit table:

1. Code the APSXUCOM macro.

   The variable names for the months of the year are alphanumeric, with a maximum length of 3 characters. With default values underscored, the syntax for the APSXUCOM macro is:

   ```
   [SYMBOL] APSXUCOM [BUFF=nnnnnnn|1599],
      [JAN= {ccc|JAN}], [FEB= {ccc|FEB}], [MAR= {ccc|MAR}],
      [APR= {ccc|APR}], [MAY= {ccc|MAY}], [JUN= {ccc|JUN}],
      [JUL= {ccc|JUL}], [AUG= {ccc|AUG}], [SEP= {ccc|SEP}],
      [OCT= {ccc|OCT}], [NOV= {ccc|NOV}], [DEC= {ccc|DEC}]
   ```

   BUFF= {*nnnnnnn*|1599}
   The size of the work buffer, ECAWKBUF, in the exit-communications area, APSUECA. *nnnnnnn* is a decimal number 1599 - 1048576. (For more information about APSUECA, see Figure 90 on page 269 - Figure 91 on page 270.)

   JAN= {ccc|JAN}
   The variable name for January

   FEB= {ccc|FEB}
   The variable name for February

   MAR= {ccc|MAR}
   The variable name for March

   APR= {ccc|APR}
   The variable name for April

   MAY= {ccc|MAY}
   The variable name for May

   JUN= {ccc|JUN}
   The variable name for June

   JUL= {ccc|JUL}
   The variable name for July

   AUG= {ccc|AUG}
   The variable name for August

   SEP= {ccc|SEP}
   The variable name for September

   OCT= {ccc|OCT}
   The variable name for October

   NOV= {ccc|NOV}
   The variable name for November

   DEC= {ccc|DEC}
   The variable name for December

2. Assemble the following module:

   ```
            START  0
   APSUCOM  CSECT
            APSXUCOM BUFF=nnnnnnn,.....
            END    APSUCOM
   ```

   PSF requires 1599 bytes of buffer space; so increase the value *nnnnnnn* by the additional space you need. For example, if you need 100 bytes, code BUFF=1699.

3. Link-edit APSUCOM either into the library that contains the PSF exit routines that use this module or into any authorized LINKLIST library.

Upon the next initialization of the functional subsystem, FSS, the new APSUCOM exit table takes effect.

Block-letter tables, APSUBTBL

The block-letter tables, APSUBTBL, consist of:

- A translate table
- An index table
- A performance block-letter table
- A default block-letter table

You can use APSUBTBL to create new block letters, add new block letters, and change existing block letters. The default tables produce block characters from the following input:

- Uppercase letters, A–Z
- Symbols, $, #, and @
- Numerals, 0– 9
- Lowercase letters, a–z, which are translated to uppercase letters for printing
- Superscript EBCDIC numerals, which are translated to numerals (0–9) for printing
- All other characters, which are translated to blanks

PSF provides the assembly-language source code in SYS1.SAMPLIB for the block-letter tables, APSUBTBL. You can modify, assemble, and install, or link-edit with the **replace** option, APSUBTBL to create your own block-letter tables.

If installation separator page Exit 1, Exit 2, or Exit 3 is installed, APSUBTBL must also be installed. If APSUBTBL is not installed, PSF issues message APS959I, and processing continues without the separator page exits.

Service routines

PSF provides these service routines:

- Block-letter routine, APSUBLK
- Exit-put routine, APSUPUT
- SWBTUREQ macro call routine, APSUMACC
- TIME SVC routine, APSUTIMC

PSF also provides these service routine interfaces:

- Block-letter routine interface, APSUBLKC
- Exit-put routine interface, APSUPUTC

Block-letter routine interface, APSUBLKC

The block-letter service routine interface is an assembler routine that is used to call APSUBLK from a C exit routine. It puts the address of APSUECA into register 5 (which is required by APSUBLK), calls APSUBLK, and passes the return code from APSUBLK back to the calling C exit. APSUBLKC uses ECACSAVE as its own save area.

Input for APSUBLKC

When APSUBLKC is entered, it contains these:

Register 1
Pointer to the address of two parameters:

- Address of parameter area APSGEXTP
- Address of 4-byte return code that holds the return code of APSUBLK

Register 13
Pointer to caller's save area

Register 14
Return address

Register 15
Entry-point address

Output for APSUBLKC

APSUBLKC passes the return code from APSUBLK back in the return code parameter.

Block-letter routine, APSUBLK

The block-letter service routine, APSUBLK, builds block letters for separator pages. Block letters can be straight or slanted, and centered or left-aligned. They are 12 characters high (using 12 printed lines) and can be either 12 or 8 characters wide.

The default separator page uses a block-letter width of 12 characters. However, using a smaller block-letter width causes separator pages to print faster. To specify a block-letter width of eight characters, set ECAPBLK on. (To see printed examples of these header pages, see Figure 45 on page 149 and Figure 46 on page 150.)

The print lines that are created for separator pages are put in ECAWKBUF, beginning at the second print position. The first print position is reserved for the printer-control characters for the first block-letter print line.

Input for APSUBLK

When the block-letter service routine, APSUBLK, is entered, it contains these:

General Register 5
The address of the exit-communication area, APSUECA.

ECABLKIN
The name (ranging from 1 - 8 characters) to be used by the block letters.

ECALJUST
If set on, used to left-align the block letters.

ECASLANT
If set on, used to slant the block letters.

ECAPBLK
If set on, used to build performance characters.

Output for APSUBLK

When APSUBLK leaves the service routine, it passes the block letters to the exit in the work buffer, ECAWKBUF. The format of the output in this buffer is shown in Figure 96 on page 283.
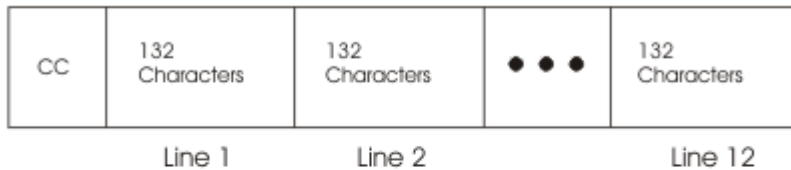
Figure 96. Format of the output in the work buffer, ECAWKBUF

Programming considerations for APSUBLK

The service routine produces corresponding block characters from the tables in APSUBTBL.

The separator routines that are supplied by PSF do not generate slanted block letters. However, you can set the ECASLANT field in APSUECA to produce block letters that are slanted to the right. For the location of this field, see Figure 90 on page 269 - Figure 91 on page 270.

The separator routines that are supplied by PSF generate block letters that are centered on the 12 output lines. You can set the ECALJUST field in APSUECA to produce left-aligned block letters, starting in the first print position.

Note: During centering calculations, ECALJUST ignores blanks to the right of the last character in the input field.

If you are using the PSF assembler samples for Exit 1, Exit 2, or Exit 3, the line length is set to 127 characters. If you decide to use slanted block letters (ECASLANT), you might lose the last few characters of your last block letter. To prevent this from happening, you can left-align the block letters (ECALJUST), use the smaller performance block letters (ECAPBLK), or increase the line length to 132 characters. If you are using the C samples for Exit 1, Exit 2, or Exit 3, the line length is set to 132 characters; therefore, you can use slanted block letters without losing the last few characters of the last block letter.

APSUBLK saves the caller's registers in the area addressed by register 13. When APSUBLK is entered, register 5 must provide APSUBLK with the address of APSUECA. APSUBLK uses ECAGSAVE as its own save area. When control is returned from APSUBLK, register 15 is reset to 0.

Exit-put routine interface, APSUPUTC

The exit-put service routine interface is an assembler routine that is used to call APSUPUT from a C exit routine. It puts the address of APSUECA into register 5 (which is required by APSUPUT), calls APSUPUT, and passes the return code from APSUPUT back to the calling C exit. APSUPUTC uses ECACSAVE as its own save area.

Input for APSUPUTC

When APSUPUTC is entered, it contains these:

Register 1
    Pointer to the address of two parameters:

- Address of parameter area APSGEXTP
- Address of 4-byte return code that holds the return code of APSUPUT

Register 13
    Pointer to caller's save area

Register 14
    Return address

Register 15
    Entry-point address

Output for APSUPUTC

APSUPUTC passes the return code from APSUPUT back in the return code parameter.

Exit-put routine, APSUPUT

The exit-put service routine, APSUPUT, enables the separator-page exit routines to put records into the PSF print-record stream.

When APSUPUT is entered, it expects the address of APSUECA in register 5. APSUPUT saves the caller's registers in the area pointed to by register 13. APSUPUT uses ECAGSAVE as its own save area.

APSUPUT uses a return-code field, ECARCODE, to identify previous failures of APSUPUT during the creation of a single separator page. Separator-page exit routines must not modify the ECARCODE field.

When APSUPUT is entered, register 15 is reset. If an error occurs while a record is put into the PSF print record stream, register 15 and ECARCODE are set and APSUPUT does not run again for this particular separator page.

Input for APSUPUT

When APSUPUT is entered, it contains these:

General Register 5
    The address of the exit-communications area (APSUECA).

APSUECA
    Before you run APSUPUT, the following fields must be set:

- The address of the record in ECARECAD.
- The record length in ECARECLN.
- The record-description flags in ECADRF, which identify the kind of record that APSUPUT is to process. These flags are also described in the IDXFLAG1 field, which is shown in .

Output for APSUPUT

Upon leaving the service routine, APSUPUT passes the ECARCODE and the contents of register 15.

SWBTUREQ macro-call routine, APSUMACC

Because MVS macros cannot be issued from a C exit routine, the assembler routine APSUMACC is called from the C exit and APSUMACC issues the SWBTUREQ macro. APSUMACC completes the SWBTUREQ parameter area. APSUMACC uses ECACSAVE as its own save area.

During PSF installation, the source code (in assembly language) for APSUMACC is installed into SYS1.SAMPLIB. You can either view the source routine online or print it. You can then modify, assemble, and install (link-edit with the Replace option) this routine. APSUMACC must be link-edited with any C exit that uses it and placed in the same load module as the C exit. See for more details.

Input for APSUMACC

When APSUMACC is entered, it contains these:

Register 1
Pointer to the address of three parameters:

- Address of parameter area APSGEXTP
- Address of SWBTUREQ parameter area
- Address of 4-byte return code from SWBTUREQ

Register 13
Pointer to caller's save area

Register 14
Return address

Register 15
Entry-point address

Output for APSUMACC

APSUMACC passes the return code from SWBTUREQ back in the return code parameter and passes back a completed SWBTUREQ parameter area.

TIME SVC routine, APSUTIMC

Because MVS macros cannot be issued from a C exit routine, the assembler routine APSUTIMC is called from the C exit and APSUTIMC issues the TIME SVC. APSUTIMC unpacks the date and time received from the TIME SVC, formats it, and completes the C exit routine date and time structure. APSUTIMC uses ECACSAVE as its own save area.

During PSF installation, the source code (in assembly language) for APSUTIMC is installed into SYS1.SAMPLIB. You can either view the source routine online or print it. You can then modify, assemble, and install (link-edit with the Replace option) this routine. APSUTIMC must be link-edited with any C exit that uses it and placed in the same load module as the C exit. See "Link-editing APSUMACC and APSUTIMC" on page 262 for more details.

Input for APSUTIMC

When APSUTIMC is entered, it contains these:

Register 1
Pointer to the address of two parameters:

- Address of parameter area APSGEXTP
- Address of date and time fields

Register 13
Pointer to caller's save area

Register 14
Return address

Register 15
Entry-point address

Output for APSUTIMC

APSUTIMC completes the date and time structure passed from the calling C exit routine.

Logical-record processing exit: Exit 4

PSF provides an exit that you can use to create an APSUX04 or APSUC04 routine for modifying records in the printed output, adding records, and suppressing records. This routine runs in the same address space as PSF does.

You can modify, assemble, and install (or link-edit) the following sample (skeleton) source routines in SYS1.SAMPLIB to create an exit for logical-record processing:

- APSUX04X/APSUC04X (for processing traditional line data)
- APSUX04Y/APSUC04Y (for processing data that is already composed into pages)

For more information about line-data records, see *PSF for z/OS: User's Guide* and *Advanced Function Presentation: Programming Guide and Line Data Reference.*

For more information about XML data, see *Extensible Markup Language (XML) 1.0 Specification* at W3C (www.w3.org).

For more information about structured fields that compose page data, see *Mixed Object Document Content Architecture Reference.*

Input for Exit 4

When the logical-record processing exit is entered, the contents of the registers are:

Register 1
Pointer to the address of parameter area APSGEXTP

Register 13
Pointer to caller's save area

Register 14
Return address

Register 15
Entry-point address

For a logical record call:

XTPRIXP
Pointer to index table entry for the print record

XTPRECP
Pointer to print record

XTPRECL
Length of record

For a line-mode conversion call at the beginning of each transmission of a data set:

XTP4NCAL
Call Exit 4 only for the first record of the data set transmission

XTP4CONV
Do conversion flag

XTP4XOFF
Horizontal offset value

XTP4YOFF
Vertical offset value

XTP4WINA
Wide or narrow paper flag

XTP4LMRG
Line merge flag

XTP4DPLX
Duplex field

Note: XTP4CONV, XTP4XOFF, XTP4YOFF, XTP4WINA, XTP4LMRG, and XTP4DPLX are valid only if this is the record call for the first record of a data set transmission.

For an end-of-transmission call:

XTP4ABNE
> Abnormal end

XTP4NEOT
> Normal end

Output for Exit 4

XTPPIND
> The exit might set the XTPPIND indicator to:
>
> - Write the current print-data record without change.
> - Skip the current print-data record.
> - Write the print records as described in new print-index entries constructed by the exit.
>
>   **Note:** These print records might or might not include the original print record.
>
> The APSGEXTP control block, which is shown in - , provides the values that are used to set XTPPIND.

XTP4NCAL
> Call Exit 4 for the first record of the data set transmission only.

These fields are used only at the first record of the data set transmission:

XTP4CONV
> Do conversion flag

XTP4XOFF
> Horizontal offset value

XTP4YOFF
> Vertical offset value

XTP4WINA
> Wide or narrow paper flag

XTP4LMRG
> Line merge flag

XTP4DPLX
> Duplex field

Other fields:

XTP4NTY
> Request exit notification at end of transmission

XTPRIXP
> Pointer to index

Programming considerations for Exit 4

These considerations apply to Exit 4:

- When Exit 4 receives control, no processing occurred. When Exit 4 sees the first record of a new data set, no resource-access processing took place. Therefore, for the first record, Exit 4 does not recognize other resources that the current data set might require.
- When the exit routine is entered, the contents of register 13 point to a standard 18-word save area that can be used to save the caller's registers. The APSUECA control block contains a save area, ECAUSAVE, for the exit processors.
- When the exit routine is entered, register 1 points to a one-word parameter list with the address of the parameter area, APSGEXTP. The XTPRECP field in APSGEXTP contains a pointer to the current print-data record. The XTPECAP field in APSGEXTP contains a pointer to the exit-communication area, APSUECA.

The APSUECA work buffer, ECAWKBUF, can be used to build records and indexes. If this work buffer is too small, see "Exit table, APSUCOM" on page 279 for information about expanding ECAWKBUF. If you expand ECAWKBUF, PSF does not use the additional storage, which exceeds the APSUECA default size, between successive exits. Therefore, the PSF exits can use this storage to maintain information between successive exits.

- Information describing the current print record is in the JES index table, IAZIDX, not in the APSUECA control block.
- Use register 12 as a base register for exit routines. If register 12 points to the beginning of the exit routine and the module name starts at the sixth byte in the routine, the module name appears in the PSF SDUMP, `Symptoms for This Dump`, if the routine has an abend. The PSF component name and the subcomponent names can also appear on the SDUMP title page.
- This routine can set the record-processing indicator XTPPIND in the parameter area APSGEXTP to show these:

  X'00'
  Write the current record. This is the default when the processing indicator is not explicitly set.

  X'01'
  Skip the current record.

  X'02'
  Write more records as defined in a new print-index table constructed by the exit. If the current record is to be printed, it must be referenced by the new index table. The format of this index table is described in Figure 98 on page 290.

  Note: When split records are processed, and the record is not changed, the default X'00' is coded.

- To request 3800 Line-Mode Conversion of 3800 line-mode resources to AFP print resources:

  – Determine whether line-mode conversion is required for the current data set. Set the XTP4CONV flag to the correct value.

  – Set the horizontal and vertical offsets, XTP4XOFF and XTP4YOFF. These fields cause adjustments that can correspond to the adjustments allowed on the 3800-1 printer. The adjustments are based on 240 pels per inch; an adjustment of 2 inches is coded as 480. The defaults are set for a vertical adjustment of 1/2 inch and a horizontal adjustment of 0 inches.

  – Determine whether line merge is required for the current data set. Set the XTP4LMRG flag to the correct value. If the data set does not use line merge, this flag is ignored.

  – If you want duplexing, set XTP4DPLX to the correct duplexing value.

  – If the printer is in Cut-Sheet Emulation mode, set XTP4WINA. Set the flag ON for landscape printing, or OFF for portrait printing (which is the default).

  – Set the line-mode flag XTP4NCAL to ON to request that the logical-record processing exit is called only once for each time a collated copy of the data set is processed. The default setting is OFF, meaning that the exit is called for every logical record.

    Note: Line-mode conversion requires a call to the exit only at the beginning of each data set transmitted.

- To add records or to change the length of the current record, consider that:

  – Information describing the current record is in the index table entry for this record. On entry to Exit 4, XTPRIXP (the pointer to the record-table index) in the APSGEXTP block contains the address of the index-table entry for this record.

    Note: When split records are processed, the index points to the last section of the record when Exit 4 is entered. XTPRECP points to the beginning of the entire record.

  – An index table must be built by using the JES IAZIDX format that describes the records to be written instead of, or in addition to, the current record. If the current record is to be included, it must also be referenced in the new index table.

    The format of IAZIDX is shown in Figure 98 on page 290. Values inside parentheses are in hexadecimal.

When control is returned to PSF, the XTPRIXP field in the APSGEXTP area must contain the address of the newly created index table. If the record that is added is a control record whose first byte is the X'5A', the record-description flags—IDXANSI and IDXDSR—in the index entry must be set to show that the record is a data-stream record that contains ANSI control characters.

– The IAZIDX index table, and the print-data records it describes, can be placed in the ECAWKBUF area of the APSUECA. If this area is too small, see "Exit table, APSUCOM" on page 279 for information about expanding ECAWKBUF. If you expand ECAWKBUF, PSF does not use the additional storage (that exceeds the APSUECA default size) between successive calls. Therefore, the PSF exits can use this storage to maintain information between successive calls.

– When a record is being added, the record that was being processed during this exit is not printed unless the newly created index table is updated to point to this record.

– If the exit saves (buffers) records or data across page boundaries, the exit must be able to handle PSF repositioning. PSF repositions to a page boundary. After PSF repositions and passes the next record to the exit, the exit must return the same data to PSF as it returned when the record was originally passed to the exit. For example, record 3 was the last record on page 10, and the exit returned ABC to PSF. Record 4 was the first record on page 11, and the exit returned DEF to PSF. If PSF repositioned to page 11 and passed record 4 to the exit again, the exit must return DEF to PSF. If this was not done, data might be lost.

– For Exit 4, PSF provides the end-of-transmission (EOT) indicators (first 3 bits in Figure 97 on page 289) and the 3800 Line-Mode Conversion indicators in field XTP4FLAG in APSGEXTP, as is shown in Figure 97 on page 289.

```
   Bit        Field      Meaning
1... ....    XTP4NTY     Exit 4  sets this bit to request an EOT notification
                         from PSF. For notification to be received, this bit
                         must be on every time Exit 4 is entered.
.1.. ....    XTP4NEOT    PSF sets this bit on to show that a normal EOT has
                         occurred (no more logical records will be processed in
                         this transmission). Exit 4 should set the processing
                         indicator (XTPPIND) to 2 and pass any remaining records
                         in the buffer back to PSF.  When this bit is set, no
                         records will be passed. If no more records remain,
                         XTPPIND does not need to be set.
..1. ....    XTP4ABNE    PSF sets this bit on to show that an abnormal EOT has
                         occurred, either because of a printer I/O error or
                         because of a JES operator command (reposition, cancel,
                         or interrupt). Exit 4 should discard any records
                         remaining in the print buffer, and return to PSF. (PSF
                         will not process any returned records if an abnormal
                         EOT has occurred.)
...1 ....    XTP4NCAL    PSF sets this bit OFF when Exit 4 is called for the
                         first record of the transmission; this maintains the
                         current processing, allowing the logical record exit
                         to be called for every record.  Exit 4 may set this bit
                         ON to request that the logical record exit be called
                         only for the first record of each data set transmission
                         (next copy of the data set or the next data set). This
                         bit does not affect the EOT call.
.... 1...    XTP4CONV    PSF sets this bit OFF when Exit 4/14 is called for the
                         first record of the transmission. Exit 4/14 may set
                         this bit ON for the first record of the transmission to
                         request that conversion for 3800 line-mode printing be
                         performed. If Exit 4 sets this bit ON for any other
                         records of the transmission, the bit is ignored.
.... .1..    XTP4WINA    Valid only if CSE and 3800 Line-Mode Conversion. PSF
                         sets this bit OFF when Exit 4/14 is called for the first
                         record of the transmission; this causes portrait pages
                         to be generated. Exit 4/14 can set this bit ON for the
                         first record of the transmission to request that
                         landscape pages be generated. If Exit 4 sets this bit
                         ON for any records of the transmission, the bit is
                         ignored.
.... ..1.    XTP4LMRG    Only valid if 3800 Line-Mode Conversion. PSF sets this
                         bit OFF when Exit 4/14 is called for the first
                         transmission. Exit 4/14 can set this bit ON for the
                         first record of the transmission to request that line
                         merge be activated. If Exit 4 sets this bit ON for any
                         other records of the transmission, the bit is ignored.
```

Figure 97. Exit 4/Exit 14 indicators in XTP4FLAG field

In Figure 98 on page 290, the first 12 bytes represent a fixed-length header. The information that starts at field IDXENTRL at offset 12 (X'0C') is repeated for each index entry. This control block is mapped by IAZIDX. For more information about IAZIDX, see your JES publications.

```
Offsets     Length          Name        Description

              Fixed-Length Index Header

   0 (0)        4            IDXID       Index-table identifier set to IDX
   4 (4)        2            IDXNUM      Number of table entries
   6 (6)        2            IDXTOK      JES-supplied token
   8 (8)        4            *           Reserved

              Repeating Index Entry

  12 (0C)       2            IDXENTRL    Length of this index entry
  14 (0E)       2            IDXRECL     Length of print data record
                                         associated with this index entry
  16 (10)       1            IDXFLAG1    Flags describing print data record
         1... ....           IDXDSR      Indicates AFPDS record
         .1.. ....           IDXLMR      Indicates line-mode record
         ..1. ....           IDXANSI     Indicates that the record contains
                                         ANSI control characters
         ...1 ....           IDXMACH     Indicates that the record contains
                                         machine control characters
         .... 1...           IDXSRS      Split record, start of record
         .... .1..           IDXSRM      Split record, middle of record
         .... ..1.           IDXSRE      Split record, end of record
         .... ...1           IDXOPJ      OPTCD=J used for record
  17 (11)       3            *           Reserved
  20 (14)       4            IDXRADR     Address of the print data record
                                         associated with this index entry
  24 (18)       8            IDXRECID    Record identifier token
```

Figure 98. Format of JES index table, IAZIDX

SMF type 6 record exit: Exit 5

You can use this exit to modify System Management Facilities (SMF) type 6 records. SMF collects and records system and job-related information that your installation can use to evaluate usage on your system. If you did not provide an APSUX05 or APSUC05 routine, PSF creates a standard type 6 record and writes it to the SMF data set.

Note: You can use this exit to suppress SMF type 6 records for JESNEWS data sets.

PSF provides a sample (skeleton) SMF type 6 record-processing routine that you can use to create, assemble, and install (or link-edit) an exit that modifies SMF type 6 records. (For more information, see) This routine runs in the same address space as PSF.

For a description of the SMF type 6 records that PSF creates, see . For detailed information about SMF type 6 records, see *z/OS MVS System Management Facilities (SMF)*.

Input for Exit 5

When the SMF type 6 record-processing exit is entered, the contents of the registers are:

Register 1
Pointer to the address of parameter area APSGEXTP

Register 13
Pointer to caller's save area

Register 14
Return address

Register 15
Entry-point address

XTP5NEWS
    JESNEWS data set indicator

XTPRECP
    Pointer to SMF type 6 record

XTPRECL
    Length of record

Output for Exit 5

XTPPIND
    The exit might set the XTPPIND indicator to:

    X'00'
        Write the SMF type 6 record.
    X'01'
        Skip the SMF type 6 record.

Programming considerations for Exit 5

The following considerations apply to Exit 5:

- When Exit 5 is entered, register 13 points to a standard 18-word save area that can be used to save the caller's registers. The APSUECA control block contains a save area, ECAUSAVE, in case Exit 5 needs to make external calls.

- When the exit routine is entered, register 1 points to a one-word parameter list with the address of the parameter area, APSGEXTP. APSGEXTP contains a pointer to the exit communications area, APSUECA.

    Exit 5 can use the APSUECA work buffer, ECAWKBUF. If this work buffer is too small, see "Exit table, APSUCOM" on page 279 for information about expanding ECAWKBUF. If you expand ECAWKBUF, PSF does not use the additional storage (that exceeds the APSUECA default size) between successive calls. Therefore, the PSF exits can use this storage to maintain information between successive calls.

- Use register 12 as a base register for exit routines. If register 12 points to the beginning of the exit routine, and if the module name starts at the sixth byte in the routine, the module name appears in the PSF SDUMP, `Symptoms for This Dump`, if the routine has an abend. The PSF component name and subcomponent names can also appear on the SDUMP title page.

- Control block IFASMFR maps the fields of the SMF type 6 record. For more information, see "Structure of SMF type 6 records" on page 161.

    Note: Fields in IFASMFR provided by SVC 83 are not available to Exit 5.

- PSF can be processing a new data set during the same time that Exit 5 is processing the SMF type 6 record for a preceding data set. This might result in Exit 5 being called for the preceding data set at the same time one of the other exits is being called for the current data set.

- Exit 5 is called from the NOTIFY subtask; the following exits are called from the FSA subtask:

  - Exit 1
  - Exit 2
  - Exit 3
  - Exit 4
  - Exit 6
  - Exit 7

    Exit 5 can be running during the same time that another exit in the FSA subtask is running for a new data set.

- Two exit communication areas (ECAs) are created for each FSA. ECEFECAP points to the ECA that is passed to the following exits:

  - Exit 1

– Exit 2

 – Exit 3

 – Exit 4

 – Exit 6

 – Exit 7

ECENECAP points to the ECA that is passed to Exit 5.

- You can change the contents of the SMF type 6 record, but you cannot change its length.

- You cannot change the address of the SMF type 6 record in XTPRECP. If you do change the address, it is ignored.

Message exit: Exit 6

APSUX06 or APSUC06 defines a message exit that is given control before most of the messages that are generated by PSF FSA APS*xxx*I and APS*xxx*A, are issued.

You can use this exit to suppress or redirect messages. You can redirect messages to the operator console, the security console, or both. To suppress or redirect messages, use the XTP6ACT flag. The exit cannot request that a message destined for the operator or the security console is printed with the data set.

You can use this exit to release the current data set to JES and request that the system hold it. To do this, set the XTP6HOLD flag. After the data set runs to completion, PSF marks the data set as one that cannot be selected.

You can use this exit to stop a printer FSA. To do this, set the XTP6TFSA flag. After the data set runs to completion, PSF stops the FSA.

If you set both the XTP6HOLD and XTP6TFSA flags, PSF stops the FSA and the system does not hold the data set.

You can also stop a job for a message that PSF indicated is going to the interrupt message page. To do this, set the XTP6TJOB flag.

PSF provides a sample routine for inspecting PSF messages. You can use this routine to create, assemble, and install (or link-edit) an exit to inspect messages. (For more information, see Table 30 on page 263.) This routine runs in the same address space as PSF.

Input for Exit 6

When the message-inspection exit is entered, the contents of the registers are:

Register 1
    Pointer to the address of parameter area APSGEXTP

Register 13
    Pointer to caller's save area

Register 14
    Return address

Register 15
    Entry-point address

XTP6PROC
    FSA identifier:
    The label of the PRINTDEV statement.

XTP6MSG
    Message ID (APS*xxx*I, APS*xxxx*I, APS*xxx*A, or APS*xxxx*A)

XTP6FLAG

> **XTP6INTP**
> > Indicate interrupt message page.

XTP6ACT

> Message action to be taken by PSF:

> **XTP6WTO**
> > Put message to the operator console.

> **XTP6SEC**
> > Put message to the security console.

> **XTP6PAP**
> > Print message with the data set.

Output for Exit 6

XTP6TFSA
> Stop the FSA.

XTP6HOLD
> Request that the system hold the data set. PSF marks the data set as one that cannot be selected.

XTP6TJOB
> Request that PSF ends the job.

XTP6ACT
> Message action requested by the exit. You can redirect the message to any or all of the destinations by setting the correct combination of flags.

> **XTP6WTO**
> > Put the message to the operator console.

> **XTP6SEC**
> > Put the message to the security console.

> **XTP6PAP**
> > Print the message with the data set.

> **XTP6SUP**
> > Suppress the message.

Programming considerations for Exit 6

The following considerations apply to Exit 6:

- When the exit routine is entered, register 13 points to a standard 18-word save area that can be used to save the caller's registers. Exit 6 might receive control while one of the other exits is active. If Exit 6 needs to make an external call, the code must use ECAMSGSV, a unique save area provided for this exit in the ECA.

- When the exit routine is entered, register 1 points to a one-word parameter list with the address of the parameter area, APSGEXTP. APSGEXTP contains a pointer to the exit-communications area, APSUECA.

  Exit 6 can use the APSUECA work buffer, ECAWKBUF. If this work buffer is too small, see "Exit table, APSUCOM" on page 279 for information about expanding ECAWKBUF. If you expand ECAWKBUF, PSF does not use the additional storage (that which exceeds the APSUECA default size) between successive exits. Therefore, the PSF exits can use this storage to maintain information between successive exits.

  Notes:

  1. The transmission-count field, ECATRNC, is not updated before this exit is called.

  2. ECEOUTTK and ECEJDVT are not updated before this exit is called.

- Use register 12 as a base register for exit routines. If register 12 points to the beginning of the exit routine, and the module name starts at the sixth byte in the routine, the module name appears in the

PSF SDUMP, `Symptoms for This Dump`, if the routine has an abend. The PSF component name and subcomponent names can also appear on the SDUMP title page.

- XTP6, a dsect in APSGEXTP, contains information that Exit 6 needs to determine its action. The pointer XTPRECP within APSGEXTP provides the address of XTP6. XTP6 is mapped within macro APSGEXTP. For more information about APSGEXTP, see Figure 87 on page 266 - Figure 89 on page 268.

- The message action, XTP6ACT, can be changed by the exit in one of these ways:
  - If the exit does not alter XTP6ACT, PSF takes the action that the message originally called for.
  - If the exit sets XTP6ACT to 0, PSF suppresses the message.
  - If the exit sets XTP6ACT to one or more actions, PSF takes those actions.

- The following restrictions apply to XTP6ACT:
  - Action messages, noted by the format APS*xxx*A or APS*xxxx*A, cannot be suppressed. If the exit requests that an action message is suppressed, PSF ignores the request and sends the message to the operator.
  - The exit cannot request that a message is printed with the data set, XTP6PAP, unless that was the original destination set by PSF.
  - If the message is suppressed, PSF still traces the message.
  - PSF issues some messages twice, once to the operator and once for printing with the data set. When a message is to be redirected, you might want to code a check for duplicate messages.

- Consider the following restrictions:
  - This exit is entered before the FSA part of PSF issues a message from PSF. The printer runs asynchronously to this exit, and is not affected by its actions. While this exit is running, the FSA part of PSF is not running.

    If any of these messages are issued, processing is not transferred to the exit:

    ```
    APS000I  APS005I  APS038I  APS046I  APS051I  APS060I  APS605I  APS957I
    APS001I  APS006I  APS042I  APS047I  APS052I  APS226I  APS608I
    APS002A  APS022I  APS043I  APS048I  APS053I  APS516I  APS610I
    APS003A  APS023I  APS044I  APS049I  APS054I  APS517I  APS922I
    APS004I  APS025I  APS045I  APS050I  APS055I  APS551I  APS955I
    ```

    Processing is not transferred to this exit in any of the following conditions:
    - For any message generated by the APSRMARK utility—messages APS580I to APS599I.
    - For any message generated by the trace formatter utility, APSTRFMT—messages APS960I to APS979I.
    - For any message generated by the APSRFLU utility—messages APS1700I to APS1711I.
    - For any message generated from the Notify Subtask (NST)—messages APS062I to APS068I, APS070I to APS072I, APS075I to APS078I, APS8560I, APS8562I to APS8564I, and APS8566I.
    - For any message generated by the operator interface—messages APS620I - APS699I, except for message APS691I.

  - The error that the message reports is not necessarily related to the data set that PSF is currently processing. The JSPA also provides information about the data set that PSF is processing (not necessarily about the data set to which the message belongs).
  - When this exit is called, the JSPA pointer, XTPJSPAP, might be set to 0.

Resource-management exit: Exit 7

PSF can call APSUX07 or APSUC07 at various times while it processes a print data set during an FSA session. Exit 7 controls these calls by setting the appropriate flags in the XTP7NOTY field of APSGEXTP. (For a listing of the XTP7 section in APSGEXTP, see "XTP7 parameter area format" on page 421.) This set of calls controls many functions, including resource-management functions. See "Programming

considerations for Exit 7" on page 296 for information about all the functions that Exit 7 controls. You use some of the resource-management functions to:

- Specify whether fonts are mapped to FOCA outline fonts, if the target printer supports outline fonts.
- Specify that PSF verifies that the FOCA outline font exists in the host library when raster fonts are mapped to outline fonts.
- Specify whether a resource is loaded from the host library, even if it already resides in virtual storage or in the printer.
- Specify which resources of a specific type are saved in the printer across data set boundaries.
- Substitute one resource for another.
- Enforce the security standards of your installation by ending jobs that attempt to access resources for which they are not authorized.

You can select resource-management functions for all resource types or for specific resource types. For example, you can use the exit to manage fonts but not overlays.

Keep in mind: Exit 7 cannot manage TrueType and OpenType fonts.

Invoking Exit 7

The call to Exit 7 can be initiated from different places in PSF. A call to Exit 7 is defined by the following fields in the XTP7 section of APSGEXTP:

XTP7ETYP
Exit type

XTP7RTYP
Resource type

Note: For BCOCA, GOCA, and PTOCA resources, this field indicates a page segment resource type. For IOCA resources, this field indicates either a page segment resource type or an object container resource type. XTP7RTYP indicates a page segment resource if the IOCA resource object is found in the page segment library. It indicates an object container resource if the IOCA resource is found in the object container library.

XTP7DSTY
Data set type

XTP7DSAT
Auxiliary data set type

Each call provides access to a specific area of PSF resource management. For a listing of APSGEXTP, see Figure 87 on page 266 - Figure 89 on page 268.

Sequence of events

1. If Exit 7 is present at PSF initialization,[14] PSF sets the XTP7INIT flag in the XTP7ETYP field to 1, to indicate that this is the first call to the exit, and then calls the exit. Next, Exit 7 might set flags in the XTP7NOTY fields to specify the resources and events for which the exit wants control.
2. Additional calls to the exit depend on the status of the XTP7NOTY flags that the exit sets during the INIT.
3. If the exit is not installed, no calls or references to the exit occur.

Exit 7 subroutine calls

The input and output for the Exit 7 subroutine calls (summarized in Table 32 on page 296) are described in terms of their associated processing indicators within each control block. Table 42 on page 448 references the input and output for each call, and Appendix F, "Field descriptions for Exit 7," on page 421, provides a listing of the control blocks.

---

[14] For a description of how PSF determines that an exit is present, see "Changing exit routines" on page 262.

Table 32. Summary of Exit 7 subroutine calls

Subroutine call and page number	Description
"Initialization (INIT) call" on page 301	Defines the control parameters of the resource exits.
"Begin-data-set call (BDSC)" on page 305	Controls the processing of the data set.
"Resource-access call (RAC)" on page 306	Controls processing for individual resources; for example: • Specifying whether a resource is loaded from the host library, even if it is already in virtual storage or in the printer. • Substituting one resource for another.
"Resource-load call (RLC)" on page 310	Requests or releases control of serially reusable resources by using ENQ or DEQ. (ENQ and DEQ are z/OS system macros. For more information about system macros, see the macro publication for z/OS.) You can also use RLC to collect statistics.
"Resource-deletion call (RDC)" on page 311.	Controls resource deletion at data set end (DSE).
"End-data-set call (EDSC)" on page 313	Alters the processing that is done with the begin-data-set call (BDSC).
"Termination call (TERM)" on page 313.	If needed, restores processing states at the time an FSA is stopped.

Programming considerations for Exit 7

The following general considerations apply to Exit 7:

• Upon entry to this routine, register 1 points to a one-word parameter list that contains the address of APSGEXTP. APSGEXTP contains a pointer to the exit-communications area, APSUECA. The work buffer, ECAWKBUF, in the ECA control block can be used to build records and indexes. This provides a method for maintaining information between successive exits. (For more information about ECAWKBUF, see Figure 90 on page 269–Figure 91 on page 270.)

• Exit 7 can receive control at the same time that one of the other exits is active. Exit 7 always uses ECARSCSV, a unique save area provided for this exit in the ECA.

  Additional pointers are provided for data areas that are not specific to PSF. For example, the JES job-separator page data area, IAZJSPA, is pointed to by the XTPJSPAP field in the APSGEXTP control block.

• Upon entry to this routine, register 13 points to a standard 18-word save area that can be used to save the caller's registers.

• Use register 12 as a base register for all exit routines. If register 12 points to the beginning of an exit routine, and the module name begins in the sixth byte of the routine, the module name appears on the SDUMP title page if an abend occurs.

  Note: PSF components and subcomponents also appear on the SDUMP title page.

Page segment considerations

Before loading a page segment in the printer, PSF can convert it to ensure that the page segment is compatible with the units of logical measurement (L-units) of the page. Unlike an overlay, a page segment does not specify its own L-units. A page segment included on a page inherits the L-units of that page.

Because PSF supports multiple L-units, multiple copies of the same page segment can be loaded in the printer. Although they appear to be the same page segment, they are different versions of the page segment, loaded into the printer with different L-units.

Special performance considerations

- XTP7NOTY flags are found in the APSGEXTP parameter area. See "XTP7 parameter area (DSECT)" on page 421 for more information. XTP7NOTY flags must not be turned on unless a specific function is intended.
- Retaining an often-used overlay or page segment over data set boundaries can improve printer throughput by eliminating the need to load the resource for each data set. However, do not retain resources arbitrarily. Page segments can take up substantial amounts of storage, and retaining overlays can force PSF to retain unneeded resources.

IOCA considerations

IOCA resources can be placed in the page segment library and the object container library. On entry to Exit 7, the XTP7RTYP field indicates a page segment resource type if the IOCA object is found in the page segment library; otherwise, it indicates an object container resource type if the IOCA object is found in the object container library. Because IOCA resources reside in libraries with other resource types, IBM recommends that you establish a naming convention to differentiate these resources. IBM does not enforce a prefix for the eight-character resource name, but you need to define a convention that identifies each type of resource. For IOCA resources, IBM recommends a two-character prefix of I1.

BCOCA, GOCA, and PTOCA considerations

BCOCA, GOCA, and PTOCA resources can be placed in the page segment library. On entry to Exit 7, the XTP7RTYP field indicates a page segment resource type for BCOCA, GOCA, and PTOCA resources. IBM recommends that you establish a naming convention to differentiate these resources. IBM does not enforce a prefix for the eight-character resource name, but you need to define a convention that identifies each type of resource in the page segment library.

IBM recommends a two-character prefix:

B1
    BCOCA resources

G1
    GOCA resources

S1
    Page segment resources

PTOCA resources do not have a recommended prefix.

Overlay considerations

Before an overlay that contains fonts, page segments, or data object resources can be printed, the overlay and its associated resources must be loaded in the printer. Fonts, page segments, and data object resources must be defined in one of two ways:

- In the active environment group (AEG) of the current page definition. PSF makes sure that the resources are loaded before the page that is using the resources prints.
- In the AEG of the overlay itself. These resources are identified as *nested* resources, and must remain in the printer if the overlay remains loaded. A nested resource is a resource mapped in the overlay.

For more information about the AEG, see *Mixed Object Document Content Architecture Reference*.

If Exit 7 is not set up to be called at Resource-Deletion for overlays, PSF deletes all overlays at data set end. One exception to this rule exists: if an overlay is specified in the PRINTDEV, it is kept along with any resources used by that overlay. If Exit 7 is called at Resource-Deletion for overlays, the exit is in full control and the only overlays that are kept are the ones with the RLSTKEEP flag set ON. If you are

specifying overlays in the PRINTDEV and using the RDC for overlays, you must keep the PRINTDEV overlays. (For more information about deleting resources, see "Full control of resource deletion" on page 312.)

Considerations on restartable abends

PSF automatically restarts when certain abend reason codes occur. See "Restarting from an abend" on page 330 for more information about restartable abends.

Because restarting after an abend can cause duplicate pages to print without notification to the operator or the job submitter, you might want to prevent PSF from automatically restarting after abends. With Exit 7, you can prevent PSF from automatically restarting by using the following flag:

XTP7RSTR
    Controls restarting of the abend.

PSF sets XTP7RSTR to 1 when an abend occurs that is restartable; otherwise, XTP7RSTR is set to 0. You can prevent PSF from restarting after a restartable abend, by setting the flag to 0 in Exit 7 during the TERM call. You cannot cause PSF to restart after an abend by setting the flag to 1. This flag does not remain set between calls.

XTP7ABRC contains the abend reason code. For example, if the abend reason code is E19, then XTP7ABRC contains X'0E19'. If no abend occurred, then XTP7ABRC contains X'0000'.

Considerations related to page definitions and form definitions

The following page definitions and form definitions are useful when you are working with Exit 7:

User-specified
    A page definition specified in the PAGEDEF or FCB parameter in the JCL or a form definition specified in the FORMDEF parameter in the JCL. These definitions are stored in a user library.

Auxiliary
    A page definition or a form definition specified in the PSF startup procedure for use in an auxiliary data set. Auxiliary data sets are:

- Job headers
- Job trailers
- Data set headers
- Message data sets

JES default
    A page definition specified in the JES2 or JES3 initialization statements.

Inline
    A page definition or a form definition within the print data set.

Modified-default
    A page definition or form definition that was the default specified in the PRINTDEV statement of the PSF startup procedure but that was modified by a OUTPUT JCL statement, the PRINTDEV statement, or the JES default font. A page definition is modified by a font list specified in the CHARS or USC parameter. A form definition is modified by *groupvalue* in the COPIES parameter, or by the FLASH parameter.

Processing considerations

The following considerations apply to the processing of page definitions and form definitions:

- An inline page definition or form definition is read into PSF directly from the user's data set. PSF uses the inline page definition or inline form definition only for the data set that contains it, and deletes it after the processing of that data set is complete.
- The name of a user-specified page definition or form definition is passed to PSF by JES. If unable to find or use an existing version of that resource, PSF finds the resource in the host library and loads it into

virtual storage. If the resource is in a user library, it is loaded into virtual storage from the user library, even if an existing system version of the resource might be used. PSF uses the user library resource only for the data set that specifies it. For more information, see "User libraries in deferred-printing mode" on page 51.

- A JES default page definition appears to PSF as if it were a user-specified page definition. PSF does not identify the page definition as a default page definition. No JES default form definition can be specified in JES.

- Device-default and auxiliary page definitions and form definitions are loaded at the beginning of the FSA session, and remain loaded until the session ends.

- When you are using a modified-default page definition or form definition, PSF copies the default resource that is specified in the PRINTDEV statement and modifies it with the respective parameters from the OUTPUT JCL statement or from the PRINTDEV statement. PSF modifies the default resource at the same time it is setting up the environment for the data set. No new resource object is created; therefore, the resource exit is never called with an APSURLST entry for a modified-default page definition or form definition.

Selection hierarchy

For information about the selection hierarchy of page definitions and form definitions, see "How PSF selects resources" on page 174.

Considerations for ending the processing of data sets

By setting the XTP7TDS flag to 1, you can use Exit 7 to end processing of the current data set. You can set this flag during the BDSC, the RAC, or the RLC.

When processing of a data set ends, the data set is typically purged. To request that the data set is returned to JES, set the XTP7HQUE flag to 1. PSF then marks the data set as one that cannot be selected.

When processing of a data set ends, PSF adds APS954I to the message data set. To add text to this message, supply:

- A pointer to your text in XTP7MSGP

- The length of the text in XTP7MSGL; up to 64 bytes

Note: If either XTP7MSGP or XTP7MSGL is 0, PSF does not add text to APS954I. XTP7TDS, XTP7HQUE, XTP7MSGP, and XTP7MSGL are in the XTP7 section of APSGEXTP. For more information about XTP7, see "XTP7 parameter area format" on page 421.

Considerations related to user libraries

The following processing considerations apply to resources stored in a user library:

- Exit 7 cannot substitute user library resources. PSF ignores a request to do so.

- Exit 7 cannot keep a user library resource across data set boundaries. User library resources are deleted before the Resource-Deletion Call (RDC). For details, see "Resource-deletion call (RDC)" on page 311.

- Exit 7 is not passed a concatenation retrieval list (APSLCRL) for user library resources.

Considerations related to inline resources

For all data formats except XML, PSF supports these inline resources:

- BCOCA
- Character sets
- Coded fonts
- Code pages
- Form definitions
- GOCA

- IOCA
- Object containers
- Overlays
- Page definitions
- Page segments
- PTOCA
- TrueType and OpenType fonts

BCOCA, GOCA, IOCA, and PTOCA resources are contained in the page segment library. The following processing considerations apply to inline resources:

- Exit 7 cannot substitute inline resources. PSF ignores a substitution request for an inline resource.
- Exit 7 cannot keep an inline resource across data set boundaries. PSF deletes inline resources before the Resource-Deletion Call (RDC).
- Exit 7 is not passed a concatenation retrieval list (APSLCRL) for inline resources.
- Exit 7 cannot force a reload of an inline resource. When PSF is unable to reload a resource, RLSTRRI is set ON. If Exit 7 requests that the resource is reloaded, the request is ignored and no error occurs.

Considerations for resource substitution

PSF supports substitution of all resource types. These considerations apply to resource substitution:

- Exit 7 cannot substitute inline resources. PSF ignores a substitution request for an inline resource.
- Exit 7 cannot substitute user library resources. PSF ignores a substitution request for a user library resource.
- Exit 7 only substitutes a system library resource with another system library resource.
- Object container resources must be substituted for object container resources of the same type. The registration ID (object-type OID) must be the same for both resources or an error results.
- A resource that is referenced by an OID cannot be substituted.
- An object container resource with secondary resources can be substituted only for an object container with the identical secondary resources. If the secondary resources for the substituted object container are different, PSF does not use them.

Considerations related to FOCA outline fonts

PSF supports FOCA outline fonts and raster fonts. When the printer does not support outline fonts, PSF maps an outline font to an equivalent raster font. This mapping function is always active; it occurs before the resource access call. For more information, see "Font mapping" on page 235.

For performance considerations, see "Using outline fonts" on page 189. For information about changing the mapping, see "Using the font mapping tables" on page 237.

The following processing considerations apply to outline fonts:

- If Exit 7 is called and a font substitution request is made, but the printer does not support the type of font that is substituted, PSF stops processing the data set and issues error messages. PSF does not map the font to an equivalent raster font or outline font, because the substitution by Exit 7 overrides all other actions.

  When you request that an outline font is reloaded, you might experience some performance degradation if several coded fonts are using the same code page or character set. All the fonts that are sharing the code page or character set are deactivated and then reactivated, all from the same reload request. (Deactivating and reactivating an outline coded font does not make it necessary to send all the font data to the printer again; only the commands necessary to associate the code page and the character set are sent.)

- When Exit 7 deletes fonts, it deletes them at the coded font level; that is, Exit 7 deletes both the code page and the character set. Because different outline coded fonts might use the same code pages and character sets, other instances of these font members are deleted (and made unusable) at the same time.

Initialization (INIT) call

The INIT call is made once after the FSA is initialized for each device. By modifying XTP7NOTY flags (listed in "XTP7 parameter area format" on page 421), you can use the INIT call to set up for future calls to Exit 7. As defaults, XTP7NOTY flags are set inactive; however, you can modify these flags to:

- Control resource access by type[15]; set any of the XTP7NACC flags.
- Load resources by type[15]; set any of the XTP7NLD flags.
- Delete resources by type; set any of the XTP7NDEL flags.
- Receive notification when a data set begins; set XTP7EBDS=1.
- Receive notification when a data set ends; set XTP7EEDS=1.
- Receive notification when an FSA stops; set XTP7ETRM=1.

The INIT modifies the reasonable resource loaded values (RRLV). PSF uses the RRLV to determine the number of resources that are saved in a printer or in virtual storage at the end of each data set. For more information, see "Partial control of resource deletion" on page 311.

INIT also provides pointers in XTP7CRLP to concatenation retrieval lists (CRLs). The CRLs contain information about system resource libraries as they were set up by use of the PSF startup procedure.

Many of the Exit 7 options that are specified with the INIT call can be specified in the Printer Inventory. Table 33 on page 301 lists these options. For more information about specifying these Exit 7 options in the Printer Inventory, see the equivalent Printer Inventory parameter in Table 7 on page 66. For information about the XTP7 fields in Exit 7, see "XTP7 fields" on page 429.

Note: When the Printer Inventory is in use for an FSS (PARM=('INV=*piname*') is specified on the EXEC statement), the settings for the options in Table 33 on page 301 are ignored on the INIT call.

Table 33. Exit 7 options specified in the Printer Inventory		
Exit 7	**Printer Inventory**	**Description**
XTP7APSF	Acknowledgement level	Controls when PSF requests an acknowledgement.
XTP7CINR	Capture inline resources	Controls the capturing of inline resources when connected to DPF or WPM.
XTP7ACSE	Check CSE fit	Controls how PSF checks the pages for cut-sheet emulation mode.
XTP7CLOS	Close libraries when idle	Controls whether PSF closes the resource libraries when a printer is inactive.
XTP7C2SI	Consolidate IM1 images	Controls the consolidation of a multiple-celled IM1 image.
XTP7WIDE	CSE orientation	Controls whether to print portrait or landscape pages when cut-sheet emulation and 3800 media origin are used. For more information, see "Media origin" on page 335.

---

[15] The *resource-access* event occurs whenever the job refers to the resource by name. The *resource-load* event occurs when the resource is loaded either into the printer or, for page definitions and form definitions, into PSF virtual storage. A resource can be loaded into the printer more than once during the processing of a data set.

Exit 7	Printer Inventory	Description
XTP7NSF	CSE sheet eject	Controls sheet eject in cut-sheet emulation mode. For more information, see "Things to consider when using cut-sheet emulation" on page 208.
XTP7PRMD	Default process mode	Sets the default processing mode that is used to print data sets containing both single-byte and double-byte fonts.
XTP7ETFF **XTP7EFFH**	Eject to front facing	Controls eject to front facing before the job-header page, before the start of a new document, or both.
XTP7ENDC	End SNA conversation	Controls the ending of the PSF SNA LU1 conversation.
XTP7RCF	Fonts	Controls the number of fonts PSF retains in printer storage between print jobs.
XTP7RFD	Form definitions	Controls the number of form definitions PSF retains in printer storage between print jobs
XPT7HLCF	Highlight communications failure message	Controls whether PSF highlights communications failure message APS6501I.
XTP7IHIB	Inhibit recovery	Controls whether PSF recovers from an error. For more information, see "Inhibiting PSF recovery" on page 326.
XTP7BINA	Input Tray Substitution: Source tray	Sets the first source tray mapping for Side and Edge Sensitive Paper Handling. For more information, see Chapter 16, "Using Side and Edge Sensitive Paper Handling," on page 209.
XTP7SIMA	Input Tray Substitution: Substitute trays: Simplex	Sets the first substitution simplex tray for Side and Edge Sensitive Paper Handling. For more information, see Chapter 16, "Using Side and Edge Sensitive Paper Handling," on page 209.
XTP7DUPA	Input Tray Substitution: Substitute trays: Duplex	Sets the first substitution duplex tray for Side and Edge Sensitive Paper Handling. For more information, see Chapter 16, "Using Side and Edge Sensitive Paper Handling," on page 209.
XTP7BINB	Input Tray Substitution: Source tray	Sets the second source tray mapping for Side and Edge Sensitive Paper Handling. For more information, see Chapter 16, "Using Side and Edge Sensitive Paper Handling," on page 209.
XTP7SIMB	Input Tray Substitution: Substitute trays: Simplex	Sets the second substitution simplex tray for Side and Edge Sensitive Paper Handling. For more information, see Chapter 16, "Using Side and Edge Sensitive Paper Handling," on page 209.

Table 33. Exit 7 options specified in the Printer Inventory (continued)

Table 33. Exit 7 options specified in the Printer Inventory (continued)		
Exit 7	**Printer Inventory**	**Description**
XTP7DUPB	Input Tray Substitution: Substitute trays: Duplex	Sets the second substitution duplex tray for Side and Edge Sensitive Paper Handling. For more information, see Chapter 16, "Using Side and Edge Sensitive Paper Handling," on page 209.
XTP7BINC	Input Tray Substitution: Source tray	Sets the third source tray mapping for Side and Edge Sensitive Paper Handling. For more information, see Chapter 16, "Using Side and Edge Sensitive Paper Handling," on page 209.
XTP7SIMC	Input Tray Substitution: Substitute trays: Simplex	Sets the third substitution simplex tray for Side and Edge Sensitive Paper Handling. For more information, see Chapter 16, "Using Side and Edge Sensitive Paper Handling," on page 209.
XTP7DUPC	Input Tray Substitution: Substitute trays: Duplex	Sets the third substitution duplex tray for Side and Edge Sensitive Paper Handling. For more information, see Chapter 16, "Using Side and Edge Sensitive Paper Handling," on page 209.
XTP7BIND	Input Tray Substitution: Source tray	Sets the fourth source tray mapping for Side and Edge Sensitive Paper Handling. For more information, see Chapter 16, "Using Side and Edge Sensitive Paper Handling," on page 209.
XTP7SIMD	Input Tray Substitution: Substitute trays: Simplex	Sets the fourth substitution simplex tray for Side and Edge Sensitive Paper Handling. For more information, see Chapter 16, "Using Side and Edge Sensitive Paper Handling," on page 209.
XTP7DUPD	Input Tray Substitution: Substitute trays: Duplex	Sets the fourth substitution duplex tray for Side and Edge Sensitive Paper Handling. For more information, see Chapter 16, "Using Side and Edge Sensitive Paper Handling," on page 209.
XTP7SENS	Input Tray Substitution	Controls whether to use Side and Edge Sensitive Paper Handling. For more information, see Chapter 16, "Using Side and Edge Sensitive Paper Handling," on page 209.
XTP7IPSU	Interrupt message page	Controls the suppression of the interrupt message page. See "Interrupt message page" on page 329.
XTP7IPCC	Interrupt message page: Copies	Controls the number of copies of the interrupt message page that are produced. See "Interrupt message page" on page 329.
XTP7IPMF	Interrupt message page: Mark page	Controls the marking of the interrupt message page. See "Interrupt message page" on page 329.
XTP7IPOS	Interrupt message page: Offset page	Controls the offset stacking of the interrupt message page. See "Interrupt message page" on page 329.

Table 33. Exit 7 options specified in the Printer Inventory (continued)		
Exit 7	**Printer Inventory**	**Description**
XTP7IMSG	Issue intervention messages	Controls whether PSF issues a message to the operator console when the printer is SNA- or TCP/IP-attached and requires intervention.
XTP7LMCM	Line-Mode Conversion paper-length errors	Controls whether AFP Download Plus reports message APS973I for Line-Mode Conversion paper-length errors.
XTP7LCNT	Line-Mode Migration LINECT	Controls whether AFP Download Plus uses line count (LINECT) in Line-Mode Migration for calculating the number of lines on a page.
XTP7MTOF	Map to outline fonts	Controls whether PSF maps fonts to outline fonts. For more information, see "Mapping to outline fonts" on page 236.
XTP7ROC	Object containers	Controls the number of object containers PSF retains in printer storage between print jobs.
XTP7JOG	Offset stacking	Controls offset stacking. For more information, see "Separating copy marking and offset stacking" on page 158.
XTP7HDF	3800 compatibility: Override default font	Controls whether PSF allows the 3800 to replace the hardware default font with the first font in the current font list. For more information, see "Font selection" on page 335.
XTP7RPD	Page definitions	Controls the number of page definitions PSF retains in printer storage between print jobs.
XTP7RPS	Page segments	Controls the number of page segments PSF retains in printer storage between print jobs.
XTP7PRND	Prune double-byte fonts	Controls whether PSF prunes double-byte raster fonts to reduce the amount of font data sent to the printer. See "Font pruning of raster fonts" on page 188.
XTP7PRNS	Prune single-byte fonts	Controls whether PSF prunes double-byte raster fonts to reduce the amount of font data sent to the printer. See "Font pruning of raster fonts" on page 188.
XTP7RFNF	Recover from font not found	Controls whether PSF makes sure the outline font that is derived from the mapped font exists before proceeding. For more information, see "Recovering from font not found" on page 236.
XTP7RDSR	Release data set when repositioning	Controls whether PSF is to release spool data sets it obtained from JES when it is repositioning to handle an exception reported by the printer. For more information, see "Disposition of spool data sets during repositioning" on page 325.
XTP7FMT	Resolution	Sets a format resolution for all jobs. For more information, see "Specifying format resolution" on page 217.

Table 33. Exit 7 options specified in the Printer Inventory (continued)		
Exit 7	**Printer Inventory**	**Description**
XTP7SDCH	Send default character	Controls the sending of the default character information for outline single-byte code pages to the printer. For more information, see "Sending default characters in single-byte outline fonts" on page 234.
XTP7MDSD	Send msgs to SYSOUT	Controls the message data set redirection.
XTP738MO	Set media origin to 3800 origin for: Data set	Controls whether to set the media origin to the 3800 media origin. For more information, see "Media origin" on page 335.
XTP7DHMO	Set media origin to 3800 origin for: Data set header	Controls whether to set the media origin to the 3800 media origin. For more information, see "Media origin" on page 335.
XTP7JHMO	Set media origin to 3800 origin for: Job header	Controls whether to set the media origin to the 3800 media origin. For more information, see "Media origin" on page 335.
XTP7JTMO	Set media origin to 3800 origin for: Job trailer	Controls whether to set the media origin to the 3800 media origin. For more information, see "Media origin" on page 335.
XTP7MDMO	Set media origin to 3800 origin for: Message data set	Controls whether to set the media origin to the 3800 media origin. For more information, see "Media origin" on page 335.
XTP7SUCM	Suppress copy marks	Controls whether PSF is to tell the printer to suppress copy marks. For more information, see "Suppressing copy marks" on page 157.

Input and output indicators for INIT

For a complete cross-reference of the XTP7 and RLST fields that are used as input and output, see Table 42 on page 448.

Programming considerations for INIT

With Exit 7, you can run the system ENQ or DEQ macros on the resource library that contains the resource about to be loaded at the resource-load call. PSF supplies the following information:

- CRLs of all partitioned data set names
- Pointers to the CRL for each type of resource
- A flag that enables INIT to retain a specific CRL

  On returning from the exit, PSF frees the storage for each CRL unless XTP7CRLF is set. CRL flags are supplied for all exit calls; however, only at the INIT call can the CRL storage be freed. See "XTP7 parameter area format" on page 421.

  Note: To use PSF storage efficiently, retain CRLs only if you need them. If you are using the ENQ or DEQ system macros with selected partitioned data sets, you can store information, such as VOLSERs, in the exit work buffer, ECAWKBUF, so that PSF can free CRL storage.

Begin-data-set call (BDSC)

The begin-data-set call (BDSC), which is optional, is controlled by the XTP7EBDS flag. This call is made before the processing of the data set begins; that is, before any separator-page exits begin. The begin-

data-set call is made for all new data sets and for data sets that are restarted after an interrupt. Fields in the Exit Communications Area (APSUECA) and its extension (APSUECE), which contain information for the data set from the user's JCL, are available for use.

Note: Any parameters that are specified in the BDSC override the parameters that are set in the initialization call or the Printer Inventory.

You can use the BDSC to:

- Control eject to front facing (ETFF). See "XTP7ETFF" on page 431 and "XTP7EFFH" on page 431.
- End processing of the data set (XTP7TDS). See "Considerations for ending the processing of data sets" on page 299.
- Modify XTP7RRLV and XTP7NOTY. For XTP7RRLV, see "Partial control of resource deletion" on page 311. For XTP7NOTY, see "Initialization (INIT) call" on page 301.
- Control the marking and offset stacking of the interrupt message page. See "Interrupt message page" on page 329.
- Control the setting of the media origin to the 3800 media origin. See "Media origin" on page 335.
- Control converting a multiple-celled IM1 image. See "XTP7C2SI" on page 432.
- Control the page checking for Cut-Sheet Emulation mode. See "XTP7ACSE" on page 430.
- Set a format resolution on a data set boundary. See "Specifying format resolution" on page 217.
- Control the message data set redirection. See "XTP7MDSD" on page 432.
- Limit the recovery done by PSF. See "Inhibiting PSF recovery" on page 326.
- Set the default processing mode. See "XTP7PRMD" on page 437.
- Control the sending of the default character information for outline single-byte code pages to the printer. See "Sending default characters in single-byte outline fonts" on page 234.
- Control the capturing of inline resources when connected to DPF or WPM. See "XTP7CINR" on page 436.
- Suppress the printing of copy marks. See "XTP7SUCM" on page 436.
- Control offset stacking. See "Separating copy marking and offset stacking" on page 158.
- Obtain summary data about resources that are used to print documents. See "XTP7ASAP" on page 438.
- Suppress print job finishing. See "XTP7PJO" on page 438.

Input and output indicators for BDSC

For a complete cross-reference of the XTP7 and RLST fields that are used as input and output, see Table 42 on page 448.

Programming considerations for BDSC

None.

Resource-access call (RAC)

The resource-access call (RAC), which is optional, is controlled by the XTP7NACC flags. This exit call, which is selected based on resource type, occurs when PSF determines that it needs a resource to continue printing. The resource access call is made once for each resource in each map, any time the map is processed. You can use this call to do these:

- End processing of the data set (XTP7TDS). For information about messages that are issued when data sets are ended, see "Considerations for ending the processing of data sets" on page 299.

  Note: Ending a data set overrides other processing.

- Substitute a resource by changing the resource name in the RLSTNAME field of the APSURLST control block. Setting the RLSTSMSG flag in the RLSTAFLG field causes message APS563I to be issued when a

resource is substituted. Text can be inserted and is pointed to by RLSTMSGP. The insert length is contained in RLSTMSGL. The maximum length is 64 bytes.

If you want to substitute for a resource, you must substitute the resource name each time the map is processed. For example, the exit must substitute a page segment name at the Map Page Segment (MPS) resource access call and the Include Page Segment (IPS) resource access call.

Notes:

1. If the exit requests a substitution message (APS563I) each time the resource access call is invoked for the resource, multiple substitution messages can be issued for a resource.

2. No substitution can be made for inline or user library resources.

3. If Exit 7 requests substitution, and PSF determines that the request is accepted, PSF attempts to retrieve the substituted resource only from the library from which the original resource is retrieved.

- Force a load or reload[16] of the resource by setting the RLSTLOAD flag in the RLSTAFLG field of the APSURLST control block.

    – If the resource is not loaded, the action is ignored.

    – If the resource is loaded, the resource is deleted and loaded from the library.

    – When PSF processes certain map structured fields, it cannot reload the resource. For example, when the exit is called for the Map Page Segment (MPS), the page segment can be reloaded. When the exit is called for the Include Page Segment (IPS), PSF is not in a state in which it can reload the page segment.

    When unable to reload the resource, PSF sets RLSTRRI (resource-reload-ignored) ON before it calls the exit. PSF ignores any request to reload the resource and no error occurs. PSF continues processing the resource as though the reload is not requested.

    Note: If a substitution is made and a load is requested, the substituted resource is loaded.

    ⚠️ **Attention:** RLSTLOAD reacquires specific selected resources from the resource library. PSF has logic to minimize the load activity; however, using the RLSTLOAD capability overrides PSF's ability to minimize this activity. Abusing the RLSTLOAD capability, such as requesting all resources of a type to be reloaded, can cause PSF and the printer to perform below expectations.

- Mark a resource as a priority-level resource (PLR). For information about PLRs, see "Partial control of resource deletion" on page 311.

- Modify the XTP7NOTY and XTP7RRLV fields. For information about XTP7NOTY, see "Initialization (INIT) call" on page 301. For information about XTP7RRLV, see "Partial control of resource deletion" on page 311.

Input and output indicators for RAC

For a complete cross-reference of the XTP7 and RLST fields that are used as input and output, see Table 42 on page 448.

Programming considerations for RAC

- Exit 7 is called to process a mapped resource each time PSF encounters a structured field for the resource:

    – Include Object (IOB) structured field

    – Include page overlay (IPO) structured field

    – Include page segment (IPS) structured field

    – Map coded font (MCF) structured field

    – Map Data Resource (MDR) structured field

---

[16] To have the exit request that a resource is reloaded, set RLSTLOAD to 1. PSF does not use an existing version of the resource, but loads the resource from the host library.

- Map medium overlay (MMO) structured field
- Map page overlay (MPO) structured field
- Map page segment (MPS) structured field

Even if the resource that contains the structured field remains loaded, subsequent references to that structured field result in calls to Exit 7.

Substitutions made to resources in that structured field do not remain in effect. Each time the resource is referenced, PSF calls the exit to give it an opportunity to request substitution.

Note: Because the exit is called each time that the resource is mapped, Exit 7 can request the substitution message (APS563I) for each occurrence of a substitution. As a result, if a resource is mapped many times in a data set, two or more substitution messages might possibly be issued.

- The resource exit cannot substitute inline resources. PSF ignores any request to substitute an inline resource.
- The exit cannot request that an inline resource is reloaded.
- The resource exit cannot substitute a user library resource name for another resource. PSF ignores an exit request to substitute a user library resource.
- Only system resources can be substituted. PSF forces the substituted resource to be loaded from the originating library. For example, when a system resource is substituted, PSF searches only in the system library for the new substituted resource, even if the substituted resource resides in the user library.
- The exit can request that a user library resource is reloaded.
- Other programming considerations apply for resources of the following types:
  - Overlays
  - Page segments
  - Fonts
  - Page definitions
  - Form definitions

RAC and overlays

Overlays are mapped resources contained in a map medium overlay (MMO) structured field or a map page overlay (MPO) structured field. For PSF to print an overlay that is mapped in an MPO, the page must contain an include page overlay (IPO) structured field. Each time PSF encounters an MMO, an MPO, or an IPO structured field, it checks to see if Exit 7 is active for overlays at access time. If it is active, PSF builds a chain of resource-list control blocks (RLSTs) containing one entry for each overlay in the MMO field, the MPO field, or the IPO field. (For more information about structured fields, see *Mixed Object Document Content Architecture Reference*.) The exit is called once, and PSF processes the RLST changes when it regains control from the exit.

RAC and page segments

Hard page segments (segments that are loaded in the printer) are found in a map page segment (MPS) structured field and are processed like overlays. Soft page segments (page segments sent inline with data) are not found in an MPS structured field, and are processed differently.

Note: Page segments included with the IDB structured field are always considered soft page segments.

You must be aware that:

- Exit 7 allows retention of hard page segments over data set boundaries.

  If a hard page segment is loaded from a previous data set, it need not be loaded again.

  If a job calls for a soft page segment, and that page segment is loaded from an earlier data set in that job, PSF treats the soft page segment as if it were a hard page segment.

- When soft page segments are declared, the reload option is ignored. For more information about reloading resources, see "Resource-access call (RAC)" on page 306.
- PSF calls the exit each time PSF encounters an MPS or Include Page Segment (IPS) structured field.

For more information about page segment structure, see *Mixed Object Document Content Architecture Reference*.

RAC and fonts

Fonts are resources that are specified in a Map Coded Font (MCF) structured field. A font can be called out by a coded font name, a code-page–character-set pair, or a GRID. An MCF structured field can contain single-entry fonts or multiple-entry fonts. Each time PSF encounters an MCF structured field, it checks to see whether Exit 7 is active for fonts at access time. If Exit 7 is active, PSF builds a chain of resource-list control blocks (RLSTs) containing one entry for each single-entry font in the MCF field.

For dynamically defined fonts (fonts with code-page and character-set pairing), PSF builds an RLST font-extension area. PSF stores the names of the code-page and character-sets in this area.

A font cannot always be represented by a single entry in the MCF structured field. Raster double-byte fonts that are dynamically defined can contain up to 190 entries. To represent these double-byte fonts, PSF builds an RLST entry for each font section. This causes the number of RLST entries in XTP7LSTC to be greater than the number of fonts mapped in XTP7RESC.

A coded font from the system library can be substituted, even though the code page or character set might reside in the user library.

During a coded font access call, PSF does not yet know the names of the code page and the character set. Therefore, PSF does not know from which library the code page and character set are loaded. RLSTUSER and RLSTINL are set to the status of the coded font only.

Restrictions

- An inline font cannot be substituted or reloaded into the printer.
- A GRID-only font cannot be reloaded. This is indicated by PSF setting the RLSTRRI bit before it calls the exit.
- When a font is being substituted, the new font type must be the same as the old font type. For example, a coded font can be substituted only for another coded font. Similarly, a code-page–character-set pair can be substituted only for another code-page–character-set pair, and a GRID must be substituted for a GRID.
- The font is considered an inline font if, during an access call for a code-page–character-set pair, any part of the pair is inline. If no part of the pair is inline, but any part of the pair is from the user library, the font is considered a user library font. In either case, even if one part is from the system library, the code page and the character set name cannot be substituted.
- Single-byte and double-byte fonts are not interchangeable.

  Note: PSF does not differentiate between single-byte and double-byte raster fonts if both are specified as coded fonts.

- For double-byte font character sets or GRID names, you cannot change font technologies. A raster font must be substituted for a raster font, or an outline font for an outline font. For coded fonts, you can change technologies.

For more information about coded fonts, code-page and character-sets, single-byte fonts, double-byte fonts, and the MCF structured field, see *Advanced Function Presentation: Host Font Data Stream Reference*.

RAC, page definitions, and form definitions

Page definitions and form definitions are not loaded in the printer and are not mapped in a structured field. However, they are treated like overlays, page segments, and fonts, with one exception: the RLST chain contains only one entry for each type of resource.

Using RAC in processing page definitions and form definitions

For an explanation of page definition and form definition types, see "Considerations related to page definitions and form definitions" on page 298.

Default and Auxiliary Processing

Default and auxiliary page definitions and form definitions are processed in two ways:

- During the initialization of PSF (Exit 7 is called for the system defaults, job header, job trailer, data set header, and message data set; this is the only time Exit 7 is called for these during the FSA):

  - XTP7AUX is set to indicate that an auxiliary resource is being processed.

  - XTP7DSAT flags are set to indicate whether the page definition or form definition is being used as a system default or as an auxiliary resource.

  - XTP7TDS (end processing of data set) is ignored.

  - Reload functions are ignored.

  - Substitutions of page definitions or form definitions can occur.

  - A substitution for the default page definition or form definition is valid for the duration of the FSA.

- While PSF is in use (called for user's data set resources):

  - If no page definition or form definition is specified in the user data set, a default is used.

  - A substitution for the default page definition or form definition is valid only for the current data set.

  - If a reload function is specified, the new default is used for the remainder of the FSA or until the next reload function. For more information about reloading resources, see "Resource-access call (RAC)" on page 306.

User-specified processing

A user can specify page definitions and form definitions in the OUTPUT JCL statement. All the functions that are listed under "Resource-access call (RAC)" on page 306 are then supported.

Inline processing

Inline page definitions and form definitions are contained in the print data set. For more information about using inline resources as printing resources, see *PSF for z/OS: User's Guide*. When inline resources are used, all the functions that are listed under "Resource-access call (RAC)" on page 306 are supported; however, no substitution for inline resources can be made.

Resource-load call (RLC)

The resource-load call (RLC) occurs before and after a resource is loaded from either a system library or a user library. This call consists of two subcalls: the load-begin subcall and the load-end subcall. These calls are optional and are controlled by the XTP7NLD field. This exit call is selected based on resource type.

The timing of the load-begin and load-end subcalls enables the exit to request or release control of serially reusable resources in the resource library by use of ENQ or DEQ. [17]

For all resource-load calls:

- PSF supplies a partitioned data set concatenation number in the RLSTPDS, in the RLSTCPDS (code page), and in the RLSTFNDS (font character set).

  Note: For restrictions, see "Programming considerations for RLC" on page 311.

- Exit 7 can modify the RLSTPLR flag for the resource that is loaded, and it can modify XTP7NOTY and XTP7RRLV. For more information about XTP7NOTY, see "Initialization (INIT) call" on page 301; for more information about XTP7RRLV, see "Partial control of resource deletion" on page 311.

---

[17] ENQ and DEQ are z/OS system macros. For more information about system macros, see the macro publication for z/OS.

- End processing of the data set (XTP7TDS). For information about issuing messages when data sets are ended, see "Considerations for ending the processing of data sets" on page 299.

Input and output indicators for RLC

For a complete cross-reference of the XTP7 and RLST fields that are used as input and output, see Table 42 on page 448.

Programming considerations for RLC

- PSF cannot ensure that the ENQ and DEQ macros are used by multiple users of a resource library. Each installation is responsible for the use of these macros.

   Note: You must not run the system ENQ macro for a member and then end processing of the data set.

- Because RLC processes only one resource at a time, only one RLST entry exists at load time.

- PSF supplies the partitioned data set concatenation number. This number is $n-1$, where n is the search-order position of the partitioned data set in the system library or the user library.

- When PSF is loading single-byte fonts, it can access the library several times. When PSF is loading double-byte fonts, it can access the library $2n+1$ times (where n is the number of sections in the double-byte font).

- At load time, special flags indicate whether the font name specified is a code page, a character set, or a coded font name.

- For double-byte fonts, the section ID is also provided.

Resource-deletion call (RDC)

The resource-deletion call (RDC) controls the deletion of resources at data set end (DSE). This call, which is optional, is controlled by the XTP7NDEL field. This exit call is selected based on resource type.

PSF deletes resources by applying a least-recently-used algorithm. PSF uses this method to maintain the number of resources that are loaded, but you can use Exit 7 to take full or partial control of resource deletions. "Partial control of resource deletion" on page 311 and "Full control of resource deletion" on page 312 describe the PSF deletion algorithm.

Input and output indicators for RDC

For a complete cross-reference of the XTP7 and RLST fields that are used as input and output, see Table 42 on page 448.

Partial control of resource deletion

You can modify the least-recently-used PSF algorithm in two ways:

- Modify the reasonable resource loaded values (RRLVs). See Table 43 on page 455 for a list of the RRLV default values that can be modified.

   PSF uses the RRLV to keep the number of resources loaded (in a printer or in virtual storage) at a manageable level. One RRLV exists for each resource type. PSF supplies the current RRLVs at every call to Exit 7. RRLVs can be set in the Printer Inventory (except the overlay RRLV) or in any call to Exit 7.

   RRLVs (except overlays and fonts) have a lower limit of 0 and an upper limit of 32767. Fonts have an upper limit of 127. However, setting the RRLVs too high can cause a resource-constrained condition, in which the printer does not have enough storage for the resources that are required to print the current page. Setting the RRLVs too low can lead to additional downloading of resources. For overlays, the RRLV is always 0. (For retention of overlays at DSE, see "Full control of resource deletion" on page 312.)

   Note: For retention of resources other than auxiliary resources at DSE, the RRLV must be larger than the number of auxiliary resources. For more information about auxiliary resources, see "Considerations related to page definitions and form definitions" on page 298.

- Mark resources as PLRs. When PSF is using RRLV in deleting resources, it deletes non-PLRs before any PLRs. A frequently used resource can be marked as a PLR; this improves its chances of being retained at DSE.

  Note: Some PLRs might not be retained. The RRLV must be large enough to include all auxiliary resources before any others can be kept.

Full control of resource deletion

You can take full control over resources kept over data set boundaries by setting the XTP7NDEL flag for the resource.

In building an RLST, PSF uses one entry for each resource currently loaded. The resource keep or delete flag (RLSTKEEP) is set according to the PSF deletion algorithm described in "Partial control of resource deletion" on page 311.

You can take full control of resources kept or deleted by modifying RLSTKEEP. On return from Exit 7, PSF checks the RLST chain and deletes resources whose RLSTKEEP flags are turned off. The number of resources that are kept is not limited to the RRLV.

If Exit 7 does not modify the RLSTKEEP flags, PSF deletes resources based on the least-recently-used algorithm.

Note: PSF tries to save all fonts, page segments, and object containers that are identified as nested resources. PSF identifies overlays with nested resources by setting the RLSTDEP flag at delete time. PSF also sets the RLSTNEST flag whenever a nested resource is referenced.

However, some restrictions apply. For more information, see "Using RDC to control resources" on page 312.

Using RDC to control resources

Keep the following considerations in mind when you are using RDC to control resources:

- PSF never deletes auxiliary page definitions, auxiliary form definitions, or auxiliary fonts. However, when the RRLV for page segments is 0 (the default), PSF deletes auxiliary page segments.
- Unless the printer does not support selective deletion, PSF does not delete a resource that is nested in another loaded resource.

  Note: Your printer might not support the selective deletion of overlays and page segments. For printers that do not support selective deletion, if PSF is requested to delete one resource of a particular type, it deletes all the resources of that type, and issues the PSF message APS551I. If a loaded overlay that contains fonts or page segments maps a deleted resource, the overlay is deleted. To determine what support your printer offers for selective deletion, see the publications for that printer.

- Exit 7 can modify the XTP7NOTY fields.
- Exit 7 can modify the RLSTPLR flag and XTP7RRLV fields; however, these changes do not take effect until the next DSE.
- Unless full control is active, PSF deletes all overlays from the printer during resource deletion, except an overlay that is specified in the PRINTDEV. If overlays are retained and are later deleted, PSF might not delete the resources nested in the deleted overlays.
- RDCs are based on the resources that are loaded at DSE, not on the resources used by the data set. For example, if a data set did not use any fonts and the respective XTP7NDEL flag was active, the exit would still be called for fonts and would contain an RLST chain with an entry for each font loaded.
- PSF deletes all inline resources at the end of the data set before the exit is called.
- PSF always deletes user library resources at the end of the data set before the exit is called.
- PSF ignores the AFPPARMS **resource-deletion-ratio** parameter or the Printer Inventory **Resource deletion ratio** parameter if XTP7NDEL calls Exit 7 at data set end (DSE) to delete coded fonts, medium overlays, page segments, or object containers. If the parameter is not 0 and the XTP7NDEL flags are set at system initialization, an error message is issued and the FSA ends

processing. If the parameter is not 0 and the XTP7NDEL flags are set at DSE, the job completes with a message to the console that indicates that the parameter is not used.

End-data-set call (EDSC)

The end-data-set call (EDSC), which is optional, is controlled by the XTP7EEDS flag. This call is made after the data set processing is complete; that is, after the resource-deletion call at data set end, which for the last data set of a job is after the trailer page. The end-data-set call is made for all new data sets and for data sets that are restarted after an interrupt. Fields in the Exit Communications Area (APSUECA) and its extension (APSUECE), which contain information for the data set from the user's JCL, are available for use.

Note: Any parameters that are specified in the EDSC override the parameters that are set in the initialization call, the begin-data-set call, or the Printer Inventory. Keep in mind that any function that is turned on at EDSC really takes effect on the next data set.

You can use the EDSC to:

- Control eject to front facing (ETFF). See "XTP7ETFF" on page 431 and "XTP7EFFH" on page 431.
- Modify XTP7RRLV and XTP7NOTY. For XTP7RRLV, see "Partial control of resource deletion" on page 311. For XTP7NOTY, see "Initialization (INIT) call" on page 301.
- Control the marking and offset stacking of the interrupt message page. See "Interrupt message page" on page 329.
- Control the setting of the media origin to the 3800 media origin. See "Media origin" on page 335.
- Control the conversion of a multiple-celled IM1 image. See "XTP7C2SI" on page 432.
- Control the page checking for Cut-Sheet Emulation mode. See "XTP7ACSE" on page 430.
- Set a format resolution. See "Specifying format resolution" on page 217.
- Control the message data set redirection. See "XTP7MDSD" on page 432.
- Limit the recovery that is done by PSF. See "Inhibiting PSF recovery" on page 326.
- Control the sending of the default character information for outline single-byte code pages to the printer. See "Sending default characters in single-byte outline fonts" on page 234.
- Control the capture of inline resources when connected to DPF or WPM. See "XTP7CINR" on page 436.
- Suppress the printing of copy marks. See "XTP7SUCM" on page 436.
- Suppress print job finishing. See "XTP7PJO" on page 438.

Input and output indicators for EDSC

For a complete cross-reference of the XTP7 and RLST fields that are used as input and output, see Table 42 on page 448.

Programming considerations for EDSC

The intention of the EDSC call is to turn off functions that are turned on during the BDSC call. Any functions that are turned on during the EDSC call do not take effect until the next data set.

Termination call (TERM)

The termination call (TERM) can be used to undo any change in processing state made during exit processing. If XTP7ETRM is active when an FSA is stopped, TERM occurs. For example, if a resource exit started the GETMAIN system macro during the INIT call, the TERM call might be used to start FREEMAIN.

In this call, you can also control whether PSF automatically restarts after a restartable abend occurs. See "Considerations on restartable abends" on page 298 for more information.

Input and output indicators for TERM

For a complete cross-reference of the XTP7 and RLST fields that are used as input and output, see .

Line-mode migration exit: Exit 8

PSF gives control to the APSUX08 or APSUC08 exit before the processing of the data set or any separator-page exit begins so the user can start Line-Mode Migration for a job on a line-mode printer, such as a 4245 or 4248. This call is made for all new data sets and for data sets that are restarted after an interrupt. Fields in the exit-communication area (APSUECA) and its extension (APSUECE), which contain information for the data set from the user's JCL, are available for use.

Input for Exit 8

When the Line-Mode Migration exit is entered, the contents of the registers are:

Register 1
Pointer to the address of parameter area APSGEXTP

Register 13
Pointer to the caller's save area

Register 14
Return address

Register 15
Entry-point address

XTP8MIG
Do Line-Mode Migration for an FCB2 flag (4245 printer).

XTP8MIG4
Do Line-Mode Migration for an FCB4 flag (4248 printer).

XTP8PBSP
Carry over spacing at page boundaries.

Output for Exit 8

XTP8MIG
Do Line-Mode Migration for an FCB2 flag (4245 printer).

XTP8MIG4
Do Line-Mode Migration for an FCB4 flag (4248 printer).

XTP8PBSP
Carry over spacing at page boundaries.

Programming considerations for Exit 8

The following general considerations apply to Exit 8:

- When this routine is entered, Register 1 points to a one-word parameter list that contains the address of APSGEXTP. APSGEXTP contains a pointer to the exit-communications area, APSUECA.
- The work buffer (ECAWKBUF) in the exit-communications area (APSUECA) can be used as a work area by the exit. PSF does not clear this area between calls; therefore, the exit can store data here between calls for one data set.
- When the exit routine is entered, the contents of Register 13 point to a standard 18-word save area that can be used to save the caller's registers.

- Use Register 12 as a base register for all exit routines. If Register 12 points to the beginning of the exit routine, and the module name begins in the sixth byte of the routine, the module name appears on the SDUMP title page if an abend occurs.
- To request Line-Mode Migration, determine whether Line-Mode Migration for an FCB2 or an FCB4 is required for the current data set. To migrate an FCB2, set the XTP8MIG flag. To migrate an FCB4, set the XTP8MIG4 flag. Keep in mind that Exit 8 can set only one of the flags at a time.
- To carry over spacing at page boundaries, set the XTP8PBSP flag.

Direct-printing, line-mode exit: Exit 14

In direct-printing mode, PSF gives the APSUX14 or APSUC14 exit control for the first record of the transmission so the user can start 3800 Line-Mode Conversion.

Input for Exit 14

When the direct-printing, line-mode exit is entered, the contents of the registers are:

Register 1
Pointer to the address of parameter area APSGEXTP

Register 13
Pointer to the caller's save area

Register 14
Return address

Register 15
Entry-point address

XTP4CONV
Do conversion flag

XTP4XOFF
Horizontal offset value

XTP4YOFF
Vertical offset value

XTP4WINA
Wide or narrow paper flag

XTP4LMRG
Line merge flag

XTP4DPLX
Duplex field

Output for Exit 14

XTP4CONV
Do conversion flag

XTP4XOFF
Horizontal offset value

XTP4YOFF
Vertical offset value

XTP4WINA
Wide or narrow paper flag

XTP4LMRG
Line merge flag

XTP4DPLX
Duplex field

Programming considerations for Exit 14

The following general considerations apply to Exit 14:

- When this routine is entered, Register 1 points to a one-word parameter list that contains the address of APSGEXTP. APSGEXTP contains a pointer to the exit-communications area, APSUECA.

- The work buffer (ECAWKBUF) in the exit-communications area (APSUECA) can be used as a work area by the exit. PSF does not clear this area between calls; so the exit can store data here between exit calls for one data set.

- When the exit routine is entered, the contents of Register 13 point to a standard 18-word save area that can be used to save the caller's register.

- Use Register 12 as a base register for all exit routines. If Register 12 points to the beginning of an exit routine, and the module name begins in the sixth byte of the routine, the module name appears on the SDUMP title page if an abend occurs.

- To request 3800 Line-Mode Conversion of 3800 line-mode resources to AFP print resources:

  - Determine whether line-mode conversion is required for the current data set. Set the XTP4CONV flag to the correct value.

  - Set the horizontal and vertical offsets, XTP4XOFF and XTP4OFF. These fields cause adjustments that can correspond to the adjustments allowed on the 3800-1 printer. The adjustments are based on 240-pels per inch; an adjustment of 2 inches is coded as 480. The defaults are set for a vertical adjustment of 1/2 inch and a horizontal adjustment of 0 inches.

  - Determine whether line merge is required for the current data set. Set the XTP4LMRG flag to the correct value. If the data set does not use line merge, this flag is ignored.

  - If you want duplexing, set XTP4DPLX to the correct duplexing value.

  - If the printer is in cut-sheet emulation mode, set XTP4WINA. Set the flag ON for landscape printing or OFF for portrait printing, which is the default.

Print parameter exit: Exit 15

You can use the APSUX15 or APSUC15 exit with Download for z/OS and AFP Download Plus to transmit more job information in the **-opa** parameter to PSF or other print servers.

Download for z/OS and AFP Download Plus call Exit 15 once for each data set processed and, upon return from the exit, append the string that contains more job information to the **-opa** parameter.

If no additional job information is provided in an exit, Download for z/OS and AFP Download Plus transmit only the following fields in the **-opa** parameter: class, destination, forms, jobid, and segmentid (if the data set is segmented).

If you use Exit 15 to send more job information with the **-opa** parameter, you might need to change or configure the receiver to use that information. How you do this depends on the receiver you are using. See your receiver's documentation for more information.

Input for Exit 15

At entry to Exit 15, the registers contain the following information:

Register 1
Pointer to the address of the APSGEXTP area.

Register 13
Pointer to an 18-word save area in field ECAUSAVE for saving the caller's registers.

Register 14
Return address.

Register 15
Entry-point address.

The following fields in APSGEXTP are valid when Exit 15 is called:

XTPECAP
Address of the exit-communications area (APSUECA).

XTPJSPAP
Address of the job separator page data area for the data set being processed (IAZJSPA).

The following fields in APSUECA are valid when Exit 15 is called:

ECAUCOMP
Address of the exit table, APSUCOM.

ECAECEP
Address of the exit-communications area extension (APSUECE).

ECAUSAVE
Save area for the exit.

ECAFSTDS
Flag that is set when this data set is the first data set in the job group.

ECALSTDS
Flag that is set when this data set is the last data set in the job group.

ECAPTYPE
A value that indicates which program called the exit. X'02' indicates that the exit is called from Download for z/OS; X'03' indicates that the exit is called from AFP Download Plus. Use this field if the exit needs to distinguish between Download for z/OS and AFP Download Plus calls.

ECAFORM
Forms value for a job. The value can come from JCL, the printer definition, or JES defaults.

The following fields in APSUECE are valid when Exit 15 is called:

ECEPDEVK
Address of the scheduler work block (SWB) token for the PRINTDEV statement.

ECEOUTTK
Address of the scheduler work block (SWB) token for the print data set's OUTPUT JCL statement.

ECEJDVTN
Print data set JDVT name.

ECETUPTR
Address of the SWBTU for the data set's OUTPUT JCL statement.

ECETULEN
Length of the SWBTU.

ECEX15P
Address of Exit 15 (when Exit 15 is written in assembly language).

ECEDSID
Data set ID assigned by JES. JES provides this ID in the functional subsystem interface (FSI) GETDS parameter list, in field GDSDSID. see *z/OS MVS Using the Functional Subsystem Interface* for more information about the GETDS parameter list.

ECEFORM
Forms value for a job. The value can come from JCL, the printer definition, or JES defaults.

Output for Exit 15

At return from Exit 15, restore the contents of all registers. Do not set a return code in register 15.

Set the following fields in the exit-communications area (ECAUECA):

ECARECAD
> Pointer to an EBCDIC string of additional pass-through fields in ECAWKBUF.

ECARECLN
> Length of the string of additional fields, or zero if no additional fields are to be transmitted.

ECAWKBUF
> An EBCDIC string of additional pass-through fields

The string of pass-through fields might look like this:

```
devname=prt6000,outgrp_name=2
```

In this example, the final format of the **-opa** parameter might look like:

```
-opa=forms=STD,class=Q,destination=PRT6,jobid=JOB00653,devname=PRT6000,outgrp_name=2
```

Note: If you include blanks or nulls (X'00') in the EBCDIC string, Download for z/OS removes them because blanks and nulls cause some remote servers to stop the pass-through parameter string. AFP Download Plus does not remove blanks and nulls but instead replaces them with X'1C'. You can also replace blanks and nulls with X'1C'.

Programming considerations for Exit 15

Consider the following points when you are coding Exit 15:

- You must be an experienced programmer who is aware of overall system implications.
- Do not issue an explicit or implied WAIT in the exit routine because it puts the FSA into a WAIT state.
- Download for z/OS and AFP Download Plus run above the 16-megabyte line. The exit can run in either 24-bit or 31-bit addressing mode. However, 31-bit addressing is recommended to allow more FSAs to run in an address space (C exits always run in 31-bit addressing mode). If the exit is AMODE 24, control blocks, which are passed to the exit, are allocated below the 16-megabyte line.
- AFP Download Plus uses Exit 15 the same way Download for z/OS does, except AFP Download Plus ignores the OUTGRP parameter because it is not supported by the receiver.
- Use only the control blocks passed to this exit.
- Code references to fields in control blocks by using the names associated with each field in the DSECT control block. Do not use absolute displacements because these displacements can change from one release of the product to another.
- Code the exit routine to be reentrant.
- The exit routine runs in supervisor state with a protection key of 1. Thus, you can use all system services available in supervisor state.
- For Exit 15 written in assembly language, use register 12 as a base register and place the module name (in printable EBCDIC) in the object code, starting at the sixth byte. If you do this, the module name appears in the SDUMP title page if an abend occurs. (ESTAE is in effect during processing of the exit.)

  Note: Sample exit APSUX15M does not follow this convention and instead uses register 10 as its base register.

- If the APSUECA work buffer (ECAWKBUF) is too small to contain the EBCDIC string of additional pass-through fields, you can expand the buffer as described in "Exit table, APSUCOM" on page 279.
- If Exit 15 is written in C, it runs in the Language Environment. See "Specific rules for C exits" on page 259 for information about writing exits in C.

Note: PSF program maintenance might require the reassembly of the exit.

Direct-print exit: Exit 16

APSUX16 or APSUC16 is the exit given control by PSF in direct-printing mode. The exit is started for the following calls:

- Begin-data-set call
- Message call
- End-data-set call

Begin-data-set call (BDSC)

PSF calls Exit 16 at the beginning of the data set. When Exit 16 receives control, no processing occurred.

To request forms marking for data set separation, use XTP16MRK. This is the only call that can be used to request forms marking.

If the bit labeled XTP16MRK is set in the Exit 16 control block, PSF sends the Mark Form order of the Execute Order Anystate command to the printer on data set boundaries.

Use XTP16RPC to request that PSF verify that the received page counter is incremented by 1 for each page that is sent to the printer. This is the only call that can be used to verify the received page counter. If the bit labeled XTP16RPC is set in the Exit 16 control block, a PSF abend occurs unless the received page counter is incremented by 1 for each page that is sent to the printer. If PSF is attached to a 3800 printer, this bit is ignored.

⚠️ **Attention:** For reasons of performance, some channel extenders do not increment the received page counter. Check with the vendor of your OEM channel extender before you set XTP16RPC.

Use XTP16APS to specify that PSF request an acknowledgement every sheet rather than an acknowledgement every page. This reduces the number of acknowledgements for jobs that are printing multiple pages per sheet (such as N-UP, duplex, and cut-sheet emulation), thereby improving the performance for these jobs. When in this mode, PSF might still need to request acknowledgements more often than once per sheet because of buffer shortages.

The amount of PSF buffer storage is controlled by the BUFNO parameter in the PRINTDEV. See BUFNO in Table 8 on page 102 for more information about this parameter.

⚠️ **Attention:** If you are using channel extenders, check with the vendor of your channel extender before you set XTP16APS. Many channel extenders do not work correctly when XTP16APS is set.

XTP16APS defaults to 0. The default specifies that PSF requests an acknowledgement every page. If this flag is set to 1, PSF requests an acknowledgement every sheet.

Input for Exit 16 begin-data-set call

When this exit is entered, the contents of the registers are:

Register 1
Pointer to the address of parameter area APSGEXTP

Register 13
Pointer to the caller's save area

Register 14
Return address

Register 15
Entry-point address

XTP16SPI
Stacked page ID

XTP16DCB
DCB address of the application program

XTP16DEV
Unit address of printer

XTP16ETY
Call type

Output for Exit 16 begin-data-set call

XTP16MRK
Forms mark flag

XTP16RPC
Received page counter flag

XTP16APS
Acknowledgement per sheet flag

Message call for Exit 16

PSF calls Exit 16 before it issues most FSA-generated messages (APS*nnn*I, APS*nnnn*I, APS*nnn*A, and APS*nnnn*A).

Use this exit call, with the XTP16ACT flag, to suppress or redirect messages. Messages can be redirected to the operator console, the security console, or both. However, if a message is intended for the operator or the security console, the exit cannot redirect it to be printed with the data set.

Input for Exit 16 message call

When the PSF direct-print exit is entered, the contents of the registers are:

Register 1
Pointer to the address of parameter area APSGEXTP

Register 13
Pointer to the caller's save area

Register 14
Return address

Register 15
Entry-point address

XTP16MSG
Message number (APS*nnn*I, APS*nnnn*I, APS*nnn*A, or APS*nnnn*A)

XTP16ACT
Message action to be taken by PSF:

> **XTP16WTO**
> Put the message to the operator console.
>
> **XTP16SEC**
> Put the message to the security console.
>
> **XTP16PAP**
> Print the message with the data set.

XTP16SPI
Stacked page ID

XTP16DCB
DCB address of the application program

XTP16DEV
Unit address of the printer

XTP16ETY
Call type

Output for Exit 16 message call

XTP16ACT
Message action requested by the exit. The message can be redirected to any or all of the destinations by setting the correct combination of flags:

> **XTP16WTO**
> Put the message to the operator console.
>
> **XTP16SEC**
> Put the message to the security console.
>
> **XTP16PAP**
> Print the message with the data set.
>
> **Note:** The exit cannot request that a message is printed with the data set unless that was the original destination set by PSF.
>
> If the exit sets XTP16ACT to 0, PSF suppresses the message.

End-data-set call (EDSC)

Exit 16 is called at the end of each data set; that is, after the data set is printed.

Input for Exit 16 end-data-set call

Register 1
Pointer to the address of parameter area APSGEXTP

Register 13
Pointer to the caller's save area

Register 14
Return address

Register 15
Entry-point address

XTP16SPI
Stacked page ID

XTP16DCB
DCB address of the application program

XTP16DEV
Unit address of printer

XTP16ETY
Call type

Output for Exit 16 end-data-set call

None.

Programming considerations for Exit 16

The following general considerations apply to Exit 16:

- When this routine is entered, register 1 points to a one-word parameter list that contains the address of APSGEXTP. APSGEXTP contains a pointer to the area, APSUECA, and a pointer to the Exit 16 parameter list.
- The following fields in APSUECA and APSUECE are valid when Exit 16 is called:

ECAECEP
APSUECE address

ECAUCOMP
APSUCOM address

ECAUSAVE
Save area

ECAWKBUF
Exit work buffer

ECEDFDEF
FORMDEF from PRINTDEV

ECEDPDEF
PAGEDEF from PRINTDEV

ECEIMPS
Impression count

ECEJOBIM
Job impression count

ECEJOBLP
Job logical page count

ECEJOBPP
Job physical page count

ECELPCNT
Logical page count

ECEPPCNT
Physical page count

ECEPRTDT
Device type from OPC

ECEPRTM
Device model

ECEPRTMD
Device model from OPC

ECEPRTT
Device type

ECEPSFD
PSF direct is attached

ECEREC
Record count

ECERPM
RPM is attached

- The work buffer (ECAWKBUF) in the exit-communications area (APSUECA) can be used as a work area by the exit. PSF does not clear this area between calls, so the exit can store data here between exit calls for one data set.

- The begin-data-set and message calls are started from the FSA subtask. The end-data-set call is started from the NST subtask.

  Because the Exit 16 calls are started from different subtasks, two ECAs are created for each printer:

  – ECEFECAP is the ECA address that is passed to the begin-data-set and message calls.

  – ECENECAP is the ECA address that is passed to the end-data-set call.

- When register 13 is entered, it points to a standard 18-word save area that is used to save the caller's register.

- Use register 12 as a base register for all exit routines. If register 12 points to the beginning of an exit routine, and the module name begins in the sixth byte of the routine, the module name appears on the SDUMP title page if an abend occurs.
- The page and record counts—ECEIMPS, ECEJOBIM, ECEJOBLP, ECEJOBPP, ECELPCNT, ECEPPCNT, and ECEREC—are counts not of the number of pages that are printed, but of the pages and records that PSF processed. PSF might have processed many pages and sent them to the printer, although the printer has not yet printed them.

  The stacked page ID, XTP16SPI, is the ID of the page at the stacker. As each page reaches the stacker, the stacked page ID is incremented. At the beginning of a data set, the stacked page ID contains either the last stacked page ID, or zero. To determine the number of pages that are stacked, calculate the difference between the end-data-set call stacked page ID and the begin-data-set call stacked page ID.
- The page counts—ECEIMPS, ECEJOBIM, ECEJOBLP, ECEJOBPP, ECELPCNT, and ECEPPCNT—include the pages on which messages are printed at the end of the data set.

  The record count (ECEREC) is the number of records received by PSF from the application.
- Processing is not transferred to the exit in any of the following circumstances:
  - If one of the following messages is issued:

    ```
    APS000I   APS006I   APS043I   APS049I   APS055I   APS066I   APS551I   APS957I
    APS001I   APS022I   APS044I   APS050I   APS060I   APS067I   APS605I
    APS002A   APS023I   APS045I   APS051I   APS062I   APS068I   APS608I
    APS003A   APS025I   APS046I   APS052I   APS063I   APS226I   APS610I
    APS004I   APS038I   APS047I   APS053I   APS064I   APS516I   APS922I
    APS005I   APS042I   APS048I   APS054I   APS065I   APS517I   APS955I
    ```

  - In response to a message generated by any of these:
    - The APSRMARK utility—messages APS580I to APS599I
    - The APSRFLU utility—messages APS1700I to APS1711I
    - The operator interface—messages APS620I - APS699I, except for message APS691I
    - The trace formatter utility, APSTRFMT—messages APS960I - APS979I
- The exit can change the message action (XTP16ACT) in one of these ways:
  - If the exit does not alter XTP16ACT, PSF takes the action that is originally intended for the message.
  - If the exit sets XTP16ACT to 0, PSF suppresses the message.
  - If the exit sets XTP16ACT to one or more actions, PSF takes those actions.
- If XTP16ACT suppresses a message, PSF still traces the message.
- The following restrictions apply to XPT16ACT:
  - Action messages (noted by the format APS*nnn*A or APS*nnnn*A) cannot be suppressed. If the exit requests that an action message is suppressed, PSF ignores the request and sends the action message to the operator.
  - The exit cannot request that a message is printed with the data set (XTP16PAP), unless that was the original destination set by PSF.
  - PSF issues some messages twice, once to the operator and once to be printed with the data set. When a message is to be redirected, you might want to code a check for the same message to avoid duplicate messages.

Chapter 26. Recovering from print errors

This information describes how:

- PSF normally recovers from errors.
- You can inhibit recovery for any pages that are already printed.
- PSF handles printer-hardware problems.
- You can respond to PSF error conditions.

Normal PSF recovery

Under JES, PSF monitors pages as they pass through the printer, and records checkpoint data in local storage. If a recoverable I/O error interrupts the processing of a data set, PSF uses this checkpoint data to resume processing at the point of interruption, with minimal operator intervention. This page-level recovery prevents duplicate or missing pages (if the operator follows documented operational procedures) because PSF knows the last successfully printed page and can restart the print job from that page if an error is encountered.

PSF also uses the JES checkpoint function to write the checkpoint data to the JES checkpoint data set at the intervals specified in the JES initialization statements. The system operator can enter JES commands to restart printing from the JES checkpoint data set.

Checkpoint intervals can be specified in places that PSF uses in the following order:

1. CKPTPAGE or CKPTSEC parameter in the user JCL.

   If both of these parameters are specified in the user JCL, JES2 or JES3 determines which is used. See the *z/OS MVS JCL Reference* for more information.

2. Applicable JES initialization parameters:

   - JES2 initialization CKPTPAGE, CKPTSEC, and CKPTMODE parameters. See "Parameters for the JES2 initialization statement" on page 125.
   - JES3 initialization CKPNTPG and CKPNTSEC parameters. See "Parameters for the JES3 initialization statement" on page 130.

Note: PSF can use only checkpoint information that is created by the same level of PSF that is currently printing the data set. For example, if a data set was checkpointed by PSF 4.6.0, but is now being printed by PSF 4.7.0, the checkpoint data is ignored and the data set is restarted from the beginning.

For some errors, sense-byte information might be provided to the operator. Some of the printers PSF supports report 24 bytes of sense information to identify the cause of every unit-check error; other printers report fewer sense bytes. Sense-byte 2 contains error-recovery action codes for conditions sensed by the printer. For more detailed sense-byte information, see the publications for your printer.

If PSF is running in direct-printing mode, error conditions that cause data loss can also cause an abend.

Disposition of spool data sets during repositioning

While repositioning to handle an exception reported by the printer, such as a paper jam or a page adjust, PSF retains the spool data sets it obtained from JES. This prevents the reordering of print data sets as a result of repositioning. Retaining the spool data sets during repositioning is a change from earlier releases of PSF. To override this change so that PSF releases data sets when repositioning, use the **Release data set when repositioning** parameter in the Printer Inventory (see Release data set when repositioning) or set XTP7RDSR to ON in the Exit 7 INIT call (see "XTP7RDSR" on page 436).

Inhibiting PSF recovery

The normal PSF error recovery actions include resetting to the point in the data path at which an error is reported and resending data from that point. Any pages that are reprinted are in the portion of the paper path of the printer where they might be damaged or imperfect and they would typically be discarded. You can use the **Inhibit recovery** parameter in the Printer Inventory (see Inhibit recovery) or XTP7IHIB in Exit 7 (see XTP7IHIB) to prevent PSF from doing recovery for any pages that are already printed. This option defaults to NO, which specifies that PSF is to do a total recovery. If the option is set to YES, PSF does not do recovery for any pages in the job that are already printed. If the setting of the inhibit option is changed within a job, PSF ignores the new setting and issues a warning message. Inhibiting recovery prevents reprinting of sensitive data, such as checks. However, only use this option at your own risk.

When an error is reported to PSF, the point at which recovery is done depends on the type of error that is reported. Each type of error is associated with some point in the data flow from PSF to the printer or within the path of paper through the printer. When the inhibit option is set, PSF does not do any recovery beyond the point at which the printer puts data onto paper (the commit station). Recovery is still allowed at any point before the commit station of the printer. If an error does occur that requires recovery for pages that are on paper, such as a paper jam, the job is ended.

When recovery is inhibited for a job, PSF causes the printer to move all printed pages to the stacker (NPRO) before the job begins and after the job completes. This is required to isolate the job from other jobs for recovery purposes and prevents reprinting of a portion of a job for which recovery is inhibited or loss of a portion of a job for which recovery is allowed. This causes numerous blank pages between jobs on a continuous forms printer and thus this function is only used on cut-sheet printers.

When recovery is inhibited, the only operator interruption or repositioning command that is allowed is the cancel command.

Printer-hardware problems

PSF provides System Display and Search Facility (SDSF) notification only to an operator authorized to receive messages for an SNA- or TCP/IP-attached printer. However, when the JES SDSF operator receives a PSF error message that indicates that a printer-hardware condition requires intervention, the type of intervention is not specified. For more specific information about SDSF, see *z/OS SDSF Operation and Customization*.

For most other printer-hardware problems, PSF runs device-dependent error-recovery procedures, issues a message to the operator, and then takes appropriate action. These actions are described in *PSF for z/OS: Messages and Codes*. If operator intervention is required (to clear a paper jam, for example), the action that is taken depends on the type of printer that is used.

If a printer is host-connected and channel-attached, PSF issues a message to the operator console, calling for operator intervention. PSF does not resume processing until the operator takes corrective action.

If the printer has a local intervention timer, such as the InfoPrint 60 printer, and the timer expires, PSF issues a message. Action is then taken as defined by the **Failure action** parameter in the Printer Inventory or the FAILURE parameter of the PRINTDEV statement (see Failure action). You can also set the timer so that it never expires.

For a printer that has no local intervention timer, such as the InfoPrint 3000 or InfoPrint 4000 printer, PSF waits for the printer operator to correct problems as they occur.

Responding to error conditions

This information describes the ways that you can respond to error conditions:

- Use the error messages that PSF produces.
- Stop a printer FSA when you cannot recover from errors.
- Restart an FSA after an abend.

- Request traces and memory dumps to diagnose PSF problems.

All messages contain information about the cause of the error. The messages are explained in *PSF for z/OS: Messages and Codes*.

Using PSF messages

PSF messages are formatted and printed in accordance with a page definition and a form definition specified in the PSF startup procedure. If a page definition and a form definition are not specified for PSF messages on the OUTPUT statement in the startup procedure, PSF looks for default page and form definitions specified in the Printer Inventory or on the PRINTDEV statement in the startup procedure. If default page and form definitions are not specified, PSF does not print the message data set.

Notes:

1. PSF messages cannot be printed in double-byte fonts. The page definition you specify to format and print messages must use only single-byte fonts.

2. To ensure that PSF messages are formatted to fit on a page, use a page definition that specifies the ACROSS printing direction. For a list of PSF-supplied page definitions that you can use to format messages in the ACROSS direction, see *PSF for z/OS: User's Guide*.

PSF issues messages in these ways:

- Message data set
- Interrupt message page
- Operator messages
- Infoprint Server common message log (see "Common message log" on page 55 for more information)

PSF can report an error in one or more messages. For example, if a paper jam occurs in your printer or if the system operator enters a FORWARD or BACKWARD space command, error messages might be reported in the message data set and the interrupt message page. Typically, these messages indicate errors:

- Repeated when the page was retransmitted
- Detected in the data stream when the page was initially sent to the printer

Message data set

PSF accumulates error messages that are generated during the processing of a data set and prints those messages at the end of the data set, preceding any separator pages.

To specify whether PSF is to print error messages at the end of the data set or redirect the messages, you can use parameters in the Printer Inventory or the PIMSG parameter, either in the PRINTDEV statement or in the JCL. For more information, see "Printing PSF error messages" on page 327 and "Redirecting the message data set" on page 328.

Printing PSF error messages

The **Print error messages** parameters in the Printer Inventory specify whether the message groups generated in the processing of a data set are printed at the end of the data set. See Print error messages and Print error messages: Maximum messages for more information. The PIMSG parameter in the user JCL or in the PRINTDEV statement can also specify whether error messages are printed at the end of the data set. The PIMSG parameter that is specified in the user JCL takes precedence over the parameter specified in the Printer Inventory or the PRINTDEV statement. For more information about the PIMSG parameter in the user JCL, see *PSF for z/OS: User's Guide*.

Note: Do not print the PSF messages data set with a page definition that uses record formatting, data suppression, or fixed text. Also, do not use a form definition that calls an electronic overlay. The correct page definition for printing the message data set is specified in the PSF startup procedure. For more information, see "PSF startup procedures for AFP printers" on page 135.

Redirecting the message data set

You can redirect the message data set as a SYSOUT data set to another CLASS or DEST for viewing or printing. This function does not include redirecting interrupt message pages that are caused by problems such as paper jams and operator commands. If you want, you can use the Printer Inventory or Exit 7 to suppress messages that are printed within a user's data set. For details, see Interrupt message page or "XTP7IPSU" on page 432.

Message data set redirection can be activated for each FSA. After message data set redirection is activated, all messages for an FSA are redirected to the specified SYSOUT data set; tell your users where they can find the error messages for their print data sets.

You can activate redirection of message data sets in one of three ways:

- Select the **Send msgs to SYSOUT** parameter in the Printer Inventory (see Send msgs to SYSOUT for more information). All the keywords that are specified on OUTPUT statement for the message data set referenced by the MESSAGE= keyword in the PRINTDEV statement is attributed to the generated SYSOUT data set.

- Set XTP7MDSD with Exit 7 (see "XTP7IPSU" on page 432). All the keywords that are specified on OUTPUT statement for the message data set referenced by the MESSAGE= keyword in the PRINTDEV statement is attributed to the generated SYSOUT data set.

- Specify PIMSG=NO on the OUTPUT statement for the message data set referenced by the MESSAGE= keyword in the PRINTDEV statement.

  When PIMSG=NO is specified on the OUTPUT statement for the message data set, all other keywords that are specified on this OUTPUT statement are also attributed to the generated SYSOUT data set. For example:

  ```
  //MSGDS  OUTPUT PAGEDEF=A06462,FORMDEF=A10110,PIMSG=NO,FORMS=STD,
  //       CLASS=M,DEST=PRT129,CHARS=60D8,DATACK=BLOCK
  ```

  In this example, each message data set is placed in a SYSOUT data set that is sent to a destination of PRT129 with a CLASS of M. All other specified attributes are used at the time the PRT129 device prints the message data set. Attributes that are specified in the Printer Inventory or PRINTDEV are not included. The PRT129 device processes the SYSOUT data set as if it were a user data set generated by the PSF FSS.

  For more information about using PIMSG in JCL, see *PSF for z/OS: User's Guide*.

The SYSOUT message data set generated is formatted as 81-byte fixed-block ANSI line data records, and can be viewed with TSO/E or printed.

For the following example, assume that CLASS=H is a held class, and each message data set is held for viewing:

```
//MSGDS  OUTPUT PAGEDEF=A06462,FORMDEF=A10110,PIMSG=NO,
//       CLASS=H
```

Notice that a page definition and a form definition are specified because some parameters on this OUTPUT statement are also used for printing messages that are generated in the middle of a user's print data set (unless such messages are suppressed by instructions in the Printer Inventory or Exit 7). The specified page definition and form definition are not used in viewing the held message-data sets.

PSF generates a separate SYSOUT data set for each print data set that contains errors. The SYSOUT data set is generated and released after the last page (or trailer page) of the associated print data set is stacked.

Note: If you specify message data set redirection when multiple FSAs are active under a single FSS, overall printer throughput might be degraded because fewer pages are printed in the same number of processor cycles. To minimize such degradation, reduce the number of FSAs for each FSS.

Interrupt message page

For some JES operator-initiated interrupts, such as the JES2 $I and $B commands, PSF prints a message within the data set, informing the user of the interrupt.

If your printer supports forms marking or offset stacking, you can identify the location of an interrupt message page that PSF inserts into your printed output. When an error such as a paper jam or a JES repositioning command occurs, PSF inserts an interrupt message page into your printed output. When PSF is marking the interrupt message page, you can request multiple copies of the page. Requesting multiple copies causes the markings to be more visible from the side of the stack of paper.

The offset stacking request is separate from the mark forms request; that is, you can request offset stacking but not request forms marking.

You can use the Printer Inventory or Exit 7 to identify the location of an interrupt message page or to suppress the interrupt message page:

Table 34. Identifying the location of an interrupt message page. # = ISPF panel; * = PIDU		
Printer Inventory	**Exit 7**	**Description**
# Interrupt message page * interrupt-message-page	XTP7IPSU	Controls the suppression of the interrupt message page.
# Interrupt message page: Mark page * mark-interrupt-message-page	XTP7IPMF	Controls forms marking for the interrupt message page.
# Interrupt message page: Offset page * offset-interrupt-message-page	XTP7IPOS	Controls offset stacking for the interrupt message page.
# Interrupt message page: Copies * interrupt-message-page-copies	XTP7IPCC	Controls the number of copies printed of the interrupt message page.

Suppressing interrupt message pages is suggested when PSF is attached to a printer with Finisher capability; otherwise, interrupt message pages separate finished documents into two finished groups with the inserted page in the middle.

Operator messages

Messages that are not associated with a print data set, such as messages that are related to separator-page processing, are sent to the operator's console.

Stopping a printer FSA

If you are unable to stop or cancel a printer with the JES commands in "Stopping PSF FSAs and FSSs" on page 147, you can use the PSF operator interface to stop the printer FSA. The syntax of the MODIFY command for stopping a printer FSA is:

```
{MODIFY|F} fss_name,FORCE,fsa_name
```

To stop a printer FSA, an operator enters the MODIFY (or F) command with the FORCE parameter. The following list describes the parameters used with the MODIFY command. All the parameters are required.

fss_name
    Specifies the name of the FSS that manages the printer FSA that is to be stopped. The *fss_name* must match the FSS name on the JES2 FSS statement or the JES3 FSSDEF statement for the FSS.

FORCE
    Specifies that the printer FSA specified by *fsa_name* is to be stopped.

fsa_name
> Specifies the printer that is to be stopped. For JES2, either of two formats, PRINTR*nn* or PRT*nnnn*, can be used for *fsa_name*.

Note: When PSF processes the FORCE command, it issues APS627I.

If the FORCE command is unsuccessful on JES3 systems, use the JES3 command to stop the printer FSA. That command is:

```
*FAIL,PRTXX
```

Restarting from an abend

PSF refreshes the FSA and restarts it for some abend conditions. (However, PSF does not request a restart more than three times within 2 hours for an abend code.) This processing enhancement increases the availability of PSF and reduces the need for operator intervention, while it retains the ability to provide diagnostic information.

For an explanation of restartable abends, see *PSF for z/OS: Diagnosis*.

Because restarting after an abend might cause duplicate pages to print without any notification to the operator or job submitter, you might want to prevent PSF from restarting after abend conditions. See "Considerations on restartable abends" on page 298 for information about how to use Exit 7 to prevent PSF from restarting after restartable abends.

Requesting traces and memory dumps

PSF provides two types of diagnostic aids: trace reports and formatted memory dumps. These aids provide a record of events within PSF that can be used in problem determination and diagnosis. For information about using PSF traces and dumps, see *PSF for z/OS: Diagnosis*.

Chapter 27. Line-Mode Migration

The Line-Mode Migration function can be used to print line-mode jobs on PSF-driven 64xx and 65xx printers in the same manner as jobs previously printed on a JES-driven line-mode printer, which is referred to as a 4245 or 4248 printer. The job JCL, FCB, and application line data do not need to be modified. However, without the use of the Line-Mode Migration function, the JCL and line data might need to be modified to print on a 64xx or 65xx printer.

Understanding Line-Mode Migration

PSF uses the Line-Mode Migration function to dynamically build and use an inline page definition for a line-mode job that uses an FCB2 or FCB4. The inline page definition is built by using the FCB specified by the FCB parameter in the job JCL. The form definition must be provided by the customer.

When spacing runs off the bottom of a page, PSF discards any spacing that remains at the end of a page and starts the next page with the first Line Descriptor (LND) structured field in the active data map. Line-mode printers differ from PSF in how they handle spacing at page boundaries because they carry over spacing to the top of the next page. You can use the Line-Mode Migration function to specify whether spacing that runs off the bottom of a page is carried over to the next page.

To build the inline page definition, PSF does these:

1. Concatenates the prefix FCB2 or FCB4 to the FCB name that is specified on the FCB parameter in the job JCL. The FCB2 prefix is used by these line-mode printers: 3203 Model 5, 3211, 3262 Model 1, 5, and 11, 4245 Model 12 and 20, 4248, 6262 Model 14. The FCB4 prefix is used by these line-mode printers: 3262 Model 5, 4248, 6262 Model 14.

2. Uses the new FCB name to retrieve the FCB in SYS1.IMAGELIB.

3. Stops processing and issues message APS982I if PSF is attempting to load an FCB4, but the FCB in the SYS1.IMAGELIB is not an FCB4.

The FCB is used to calculate the page length that goes into the inline page definition. The lines per inch and channel codes from the FCB2 and FCB4 also go into the inline page definition. All other attributes in an FCB2 and FCB4 are ignored.

The page length sets the physical printable area length that is used when the job is printed on the 64xx or 65xx printer. Some 64xx or 65xx printers (for example, the 6412 Model CTA) allow PSF to set only physical printable area if the maximum page length is set to the initial printer default (see the setup guide for your 64xx or 65xx model). The maximum physical printable area length that the 64xx or 65xx printer supports is 22.75 inches; however, the width of the paper that is loaded in the 64xx or 65xx printer must match the width of the paper used in the 4245 or 4248 printer. Therefore, some 4245 and 4248 line-mode jobs do not print on a 64xx or 65xx printer if the page length calculated from an FCB is too large. If the page length is too large, the printer generates an error. For more information about an FCB2 or FCB4, see *z/OS DFSMSdfp Utilities*.

The number of lines that are printed per page is calculated from the FCB2 or FCB4. You can reduce this number by doing these:

- Use Exit 7 to indicate that you want the LINECT parameter to override the number of lines specified in the FCB.
- Specify a smaller number in the LINECT parameter in the user's OUTPUT JCL or in the JES2 PRINTDEF statement.

   Note: JES3 does not provide a way to specify the maximum number of lines per page.

   For more information about the PRINTDEF statement, see *z/OS JES2 Initialization and Tuning Reference*.

If the FCB parameter is not specified on the job JCL, PSF stops migration processing, issues message APS970I, and puts the job on the JES hold queue. To avoid this error, you can use the FCB parameter on the JES2 PRT*nnnn* statement to specify a default FCB to be used with the job. Otherwise, if the FCB parameter is specified on the job JCL, it is used instead of the one specified on the JES2 statement. If an FCB parameter and a FORMLEN parameter are specified, PSF stops migration processing, issues message APS981I, uses the FORMLEN value to set the page length in the 64xx or 65xx printer, and ignores the FCB parameter. If a PAGEDEF parameter is specified, PSF stops migration processing and uses the page definition specified. For information about the FCB parameter on the PRT*nnnn* statement, see *z/OS JES2 Initialization and Tuning Reference*. For information about the FORMLEN parameter or the PAGEDEF parameter, see *PSF for z/OS: User's Guide*.

The 64xx or 65xx printer supports three different print quality levels: Draft, DP, and NLQ. Not all of these print qualities can be used with every font supported by a 64xx or 65xx printer; for example, Gothic can print with a quality of Draft and DP, but not NLQ. If a print quality cannot be used with a particular font, the printer generates an error. The 64xx or 65xx printer can be configured with a default font and print quality; however, the default font can be overridden by using the **Character sets** parameter in the Printer Inventory (see Character sets) or the CHARS parameter on the PRINTDEV statement or on the job JCL. Using the **Character sets** parameter or the CHARS parameter on the PRINTDEV eliminates the need to add the CHARS parameter to the job JCL. A print quality that is supported for that font must be used to prevent errors. For information about the fonts your printer supports, see the documentation provided with the printer. For information about how PSF determines which font to use when it prints a data set, see Chapter 11, "Managing resources," on page 173.

A form definition must be used to override the default print quality set in the printer. The form definition can be specified by using the **Form definition** parameter in the Printer Inventory (see Form definition) or the FORMDEF parameter on the PRINTDEV statement or on the job JCL. Using the **Form definition** parameter or the FORMDEF parameter on the PRINTDEV eliminates the need to add the FORMDEF parameter to the job JCL.

Three form definitions are provided by PSF, each one specifying a different print quality:

Form Definition
    Print Quality

F1QA0010
    NLQ

F1Q50010
    DP

F1Q10010
    Draft print quality

Each of these form definitions specifies a logical page position of 0.0 to match the page origin on a 4245 or 4248 printer. For information about the FORMDEF parameter and form definitions that are supplied with PSF, see *PSF for z/OS: User's Guide*.

Enabling Line-Mode Migration

The Line-Mode Migration Exit 8 (APSUX08 or APSUC08) is used to enable the Line-Mode Migration function and carry over spacing at page boundaries. Line-Mode Migration is only supported in deferred-printing mode. You can set these flags:

- To enable Line-Mode Migration for a line-mode job that uses FCB2, set the XTP8MIG flag. For a line-mode job that uses FCB4, set the XTP8MIG4 flag. The XTP8MIG or XTP8MIG4 flag defaults to 0, which means the job is not migrated.

- To carry over spacing at page boundaries, set the XTP8PBSP flag for a line-mode job. The XTP8PBSP flag defaults to 0, which means that spacing at page boundaries is discarded.

By using Exit 8, it is possible to print AFP jobs and 4245 or 4248 line-mode jobs on the same printer. Exit 8 has access to the job JCL parameters, such as job name and job class, which can be used to identify a 4245 or 4248 line-mode job that needs to be migrated to print on the 64xx or 65xx printer. Exit 8 must set the XTP8MIG flag to migrate a line-mode job that uses FCB2 and the XTP8MIG4 flag to migrate a line-

mode job that uses FCB4. If all 4245 or 4248 line-mode jobs are to be migrated, Exit 8 can set the XTP8MIG or XTP8MIG4 flag every time it is called; however, Exit 8 can set only one of the flags at a time.

If a 4245 or 4248 line-mode job is not migrated, PSF continues to process the job as a normal AFP job. PSF treats the FCB specified by the FCB parameter on the job JCL as a page definition and attempts to load it from the page definition libraries. If the load fails, PSF uses the default page definition to print the 4245 or 4248 line-mode job, which most likely results in data stream errors.

You can use Exit 7 to request that the LINECT parameter (on either the user's OUTPUT JCL or the JES2 PRINTDEF statement) reduce the number of lines printed on a page from what was calculated with the FCB. If the XTP7LCNT flag is set on during the INIT call, PSF uses the LINECT value if it is smaller than the number of lines specified in the FCB. For more information, see "XTP7LCNT" on page 438.

Considerations and limitations of the Line-Mode Migration function

The Line-Mode Migration function is strictly an aid for migration. Results are unpredictable if more advanced line data functions, such as SOSI and mixed data, are used when exit flag XTP8PBSP is set.

The inline page definition that is built by PSF is used for the user data set, but the page length that is determined from the FCB applies to all pages of the job, including the separator pages. If the separator page is too long for the page length set in the 64xx or 65xx printer, the separator page is truncated by the printer and the printer generates an error.

4245 and 4248 line-mode jobs can be printed on cut-sheet printers that are supported by PSF, but the job must fit on the paper that is loaded in the printer; otherwise, data stream errors occur. When the line-mode jobs are printed on cut-sheet printers, PSF uses the paper size that is returned by the printer and does not calculate the paper size from the FCB.

PSF can use the LINECT parameter to reduce the number of lines printed per page. However, if skip-to-channel commands exist in the data that specify channels that are past the LINECT number of lines, an error is generated and message APS346I is issued.

For general information about PSF exit processing, see Chapter 25, "Using installation exits," on page 257. For more information about Exit 8, see "Line-mode migration exit: Exit 8" on page 314.

Considerations and limitations when using AFP Download Plus

Paper length and width are specified with two parameters that determine the printable area to be used. The parameters are:

- **paper-length** and **paper-width** AFPPARMS parameters specified by using the AFPPARMS control statement in the startup procedure or the AFPPARMS control statement on the OUTPUT statement.
- **Paper length** and **Paper width** specified by using the Printer Inventory.

The default for paper length is 14 inches and the default for paper width is 13.2 inches. If the FCB printable area is within the parameter defaults, the page length and page width parameters do not need to be specified.

One of these can cause off-the-logical-page errors to occur when printing:

- The paper length or paper width parameter is set too small.
- The paper that is loaded at the printer is not the correct size.

Chapter 28. 3800 considerations

This information describes the considerations that are specific to 3800 printers, including:

- Font selection
- Forms marking
- Migration

Font selection

You can use the Printer Inventory or Exit 7 to indicate whether PSF tells the 3800 to replace the hardware default font with the first font in the current font list (see Table 35 on page 335):

Table 35. Selecting the hardware default font. # = ISPF panel; * = PIDU		
Printer Inventory	**Exit 7**	**Description**
# 3800 compatibility: Override default font * override-3800-default-font	XTP7HDF	Controls whether the hardware default font is replaced with the first font in the current font list.

Initially, the default indicates that the 3800 uses the hardware default font.

To increase the amount of raster pattern storage available, you can specify the first font in the current font list as the hardware default. Consider this option when data set processing is ended with PSF message APS921I.

Note: The hardware default font might differ from one printer to another and from one operating system to another. When the same application is printed on a different printer or under a different operating system; therefore, using this option can produce output incompatibilities.

Forms marking

The 3800 prints 3–5 copies of the marked form; the number is a function of the printer. The marking of forms is the same for data pages (for example, the interrupt message page) as it is for trailer pages. For more information about forms marking, see "Forms marking" on page 153.

For the 3800 printer, PSF ignores the copies that are specified in the Printer Inventory (see Interrupt message page: Copies) or with Exit 7 (see "XTP7IPCC" on page 430). Instead, the 3800 device controls the number of copies printed of the interrupt message page.

Migration

This information describes migration considerations for the 3800, including:

- Setting the media origin when you are migrating from a 3800 printer to a 3835 or 3900 printer.
- Using the PSF Line-Mode Conversion function to migrate from a 3800 line-mode printing environment to an AFP printer.

Media origin

The Printer Inventory or Exit 7 can aid in migrating from a 3800 printer to a 3835 or 3900 printer by setting the media origin of the destination printer to match that of the 3800 printer. The location of the media origin is one of the main differences between the 3800 and other AFP continuous-forms printers. For information about the media origin a printer supports, see the documentation provided with the printer.

 335

When you are migrating 3800 print jobs, you can provide a compatible media origin by including the proper presentation options in the form definition for each job. Alternatively, you can tailor the Printer Inventory or Exit 7 to enforce a 3800-compatible form origin for any designated job.

Note: The media origin override is only offered as a migration tool to aid the user in migrating from a 3800 to a more recent printer. As applications are developed or modified to take advantage of new functions, the work must be done to remove the need for the 3800 origin override. Jobs that are designated by this override must not specify capabilities that are not supported by the 3800 printer, such as N_UP, Page Modification Controls, Post-processing controls, Presentation System Setup IDs, or Presentation Fidelity Controls. If, for example, N_UP is specified in the form definition of a job that is designated as 3800-compatible, the 3800-compatible override is ignored. PSF processes the job by using the N_UP capabilities of the 3900 or 3835 printer to which the job is sent, but issues a warning message that notes the conflict between the override and the form definition.

If the media origin is set for a job that is routed to a cut-sheet printer or routed to a continuous-forms printer that does not support changeable media origin, the media origin is ignored, but no message is issued.

You can use the Printer Inventory or Exit 7 to control whether to set the media origin to the 3800 media origin (see Table 36 on page 336):

Table 36. Setting media origin options		
Printer Inventory	**Exit 7**	**Description**
Set media origin to 3800 origin for: Data set	XTP738MO	Controls whether to set the media origin to the 3800 media origin.
Set media origin to 3800 origin for: Data set header	XTP7DHMO	Controls whether to set the media origin to the 3800 media origin.
Set media origin to 3800 origin for: Job header	XTP7JHMO	Controls whether to set the media origin to the 3800 media origin.
Set media origin to 3800 origin for: Job trailer	XTP7JTMO	Controls whether to set the media origin to the 3800 media origin.
Set media origin to 3800 origin for: Message data set	XTP7MDMO	Controls whether to set the media origin to the 3800 media origin.

These options default to NO, which specifies that PSF is not to set the media origin to the 3800 media origin. If any option is set to YES, PSF sets the media origin to the 3800 media origin for that particular page or pages.

If you set one or more of the 3800 media origin options, use wide paper on the 3800, and use cut-sheet emulation, the media origin is not in the correct position. PSF assumes narrow paper (portrait pages are generated for printing in cut-sheet emulation mode).

You can also use the Printer Inventory or Exit 7 to control whether PSF sets the media origin for wide or narrow paper (see Table 37 on page 336):

Table 37. Setting media origin for wide or narrow paper		
Printer Inventory	**Exit 7**	**Description**
CSE orientation	XTP7WIDE	Controls whether to print portrait or landscape pages when you are using cut-sheet emulation and 3800 media origin.

This option defaults to NO, which specifies that PSF is to print portrait pages. If the option is set to YES, PSF assumes wide paper and prints landscape pages. This option is only valid when you are using cut-sheet emulation with one or more of the 3800 media origin flags set in the Printer Inventory or Exit 7.

Line-Mode Conversion

The PSF 3800 Line-Mode Conversion function can simplify your migration from a 3800 line-mode printing environment to an AFP printer.

All 3800 printers can operate in line mode as non-AFP printers. However, in line mode they do not use PSF or AFP resources such as page definitions and form definitions; rather, they use 3800 line-mode resources such as FCBs and GRAPHMODs. Without the PSF 3800 Line-Mode Conversion function, migrating to the AFP print environment might make it necessary for many of the resources for a 3800 line-mode job to be converted manually to AFP resources. The PSF 3800 Line-Mode Conversion function can reduce the number of required changes to resources, JCL, and application line data, and allow you to use a limited set of standard AFP functions that are not available on the 3800 line-mode printer.

This information explains what 3800 Line-Mode Conversion is, how it works, how to enable it, some limitations or restrictions that you might encounter in your particular environment, and the implementation steps you must follow.

What is IBM 3800 Line-Mode Conversion

The PSF 3800 Line-Mode Conversion function emulates the operation of a JES-driven IBM 3800 line-mode printer. It processes JCL, resources, and line data in much the same way as a 3800 line-mode printer, and it detects many common error conditions. In other words, you can print your old 3800 line-mode jobs on an AFP printer.

In one special case, 3800 Line-Mode Conversion even supports nonstandard OEM FCBs that were designed to print from perforation to perforation on non-IBM printers that are similar to the 3800 printer. PSF 3800 Line-Mode Conversion is not supported for a 3800 printer in AFP mode.

Line-mode conversion also makes available some standard AFP functions that are not offered in the 3800 line-mode printing environment:

- You can use one electronic overlay, instead of forms flash.
- You can mix AFP and 3800 line-mode jobs on the same printer.
- You can set the top and left margins to use the entire printable area, even for old 3800 line-mode jobs.
- You can use duplex printing capability.

How does 3800 Line-Mode Conversion work

The 3800 Line-Mode Conversion function enables PSF to dynamically build and use inline page definition and form definition resources for 3800 line-mode jobs. The resources are built from information in the job's FCB and JCL for the LINECT, COPIES, and FLASH parameters.

To build the inline page definition, PSF concatenates the member that is named FCB3 to the FCB value in the JCL, and accesses that member in SYS1.IMAGELIB. If no FCB is specified for the print job, PSF is unable to continue, and stops processing. The expected paper size is calculated from the FCB; it must match the size of the paper loaded in the printer. Page height, as specified in the page definition, includes the full printable page area to allow room for character ascenders and descenders that a 3800 printer processing in line mode might truncate. The default margin at the top and bottom of the page is set to 1/2 inch (as on the 3800 printer), but can be changed to allow use of the entire printable area. If LINECT is specified in the JCL, the inline page definition contains either the number of lines that are specified by LINECT or the number that is specified in the FCB, whichever is smaller.

Information for the inline form definition is derived from several sources. To set the printer origin to match the origin on the 3800 printer, PSF uses information about the dimensions of the paper that is currently loaded in the printer. If a forms flash is used for the job, the software also extracts the FLASH name from the JCL, and indicates an electronic overlay of the same name. If PSF does not find a user-created electronic overlay with the same name (prefixed with O1) in the PSF resource library, PSF issues an error message. Uncollated copies that are specified by the COPIES parameter in the JCL are used with FLASH to determine how many copy subgroups are specified in the inline form definition.

PSF also uses other information in the JCL, though that information is not integrated into the inline resources. Thus if multiple transmissions, or collated copies, are specified in the COPIES parameter, PSF produces multiple collated copies. Further, if 3800 GRAPHMODs and character sets are specified by the CHARS parameter, they are accessed as AFP fonts. If you use this method to specify fonts, be aware that unless like-named AFP fonts exist in the PSF font library before PSF is started, an error results.

How can PSF 3800 Line-Mode Conversion be enabled

The conversion function is activated through PSF Exit 4 in deferred-printing mode or through PSF Exit 14 in direct-printing mode. Exit 4 is the PSF record processing exit, which typically gains control each time PSF accesses a print record from the JES spool. Exit 14 is the PSF direct-printing 3800 Line-Mode Conversion exit, which gains control for the first record of the transmission. To enable the conversion function, set flag XTP4CONV on for the first record of the transmission for either exit. For any other records of the transmission in Exit 4, PSF ignores XTP4CONV, although no error message is issued. As a default, PSF sets this flag to OFF before it calls Exit 4/14 for the first record of the transmission.

If you use Exit 4 for any other modification of the records, they are processed normally, as though the 3800 Line-Mode Conversion function did not exist.

Note: For more information about PSF exit processing, see Chapter 25, "Using installation exits," on page 257.

You can set XTP4NCAL flag ON in Exit 4/14 to indicate that this exit is not called again for the rest of the current transmission. Because the JCL is analyzed and inline resources are built only when the first record is processed, you experience better overall performance if you set XTP4NCAL on. However, if you require Exit 4/14 functions on any record later in the current transmission, you must wait until you no longer want PSF to call Exit 4/14 for the current transmission before you set XTP4NCAL on.

When you program the logic for Exit 4/14, you identify the 3800 jobs you want converted. Because the exit has access to all the JCL parameters for the job, you might choose to identify jobs by any of these parameters—for example, Job Name or Job Class. To intermix AFP jobs and 3800 line-mode jobs, Exit 4/14 must identify each job as one kind or the other, and set the XTP4CONV flag off for AFP jobs and on for 3800 line-mode jobs.

You can set up a table in Exit 4/14 that allows the user, by specifying any of the parameters available to the exit, to select special print offsets, XTP4XOFF and XTP4YOFF, for different 3800 line-mode jobs. This enables the conversion function to emulate the print offsets that a 3800 operator might enter to match print to special forms. Exit 4/14 can set the horizontal offset value XTP7XOFF and the vertical offset value XTP4YOFF. Using these fields, you can make adjustments that correspond to the adjustments allowed on the 3800 line-mode printer. The adjustments are based on 240 pels per inch; an adjustment of 2 inches is coded as 480. The defaults are set for a vertical adjustment of 1/2 inch and a horizontal adjustment of 0 inches. These bits are valid only when the XTP4CONV flag is on.

You can set the XTP4LMRG flag in Exit 4/14 to indicate to PSF that line merge is activated. This line merge duplicates the line merge function of a 3800 while in line mode. To enable line merge, you set flag XTP4LMRG on for the first record of the transmission. For any other records of the transmission, XTP4LMRG is ignored, although no error message is issued. As a default, PSF sets this flag to off before Exit 4/14 is called for the first record of the transmission. The flag XTP4LMRG is only valid when XTP4CONV is set on.

You can set the XTP4DPLX flag in Exit 4/14 to indicate to PSF whether duplexing is enabled and if so, what type is enabled. To enable duplexing, you set field XTP4DPLX for the first record of the transmission. For any other records of the transmission, XTP4DPLX is ignored, although no error message is issued. As a default, PSF sets this field to zero before Exit 4/14 is called for the first record of the transmission. The field XTP4DPLX is only valid when XTP4CONV is set on.

The conversion function provides line-mode processing only for jobs selected by Exit 4/14. However, not all jobs that are selected by Exit 4/14 are converted. Any job that has a PAGEDEF or FORMDEF parameter coded in its JCL is ignored for line-mode processing. PSF assumes that the job already is designed for AFP printing, and processes it as a normal AFP job.

If you are using Cut-Sheet Emulation mode, you can set the XTP4WINA flag on in Exit 4/14 to indicate to PSF whether your 3800 line-mode job was designed for wide or narrow paper. If this flag is set OFF, PSF generates portrait pages to emulate narrow paper. If this flag is set ON, PSF generates landscape pages to emulate wide paper. To enable landscape pages, you set flag XTP4WINA ON for the first record of the transmission. For any other records of the transmission, XTP4WINA is ignored, although no error message is issued. As a default, PSF sets this flag to OFF before Exit 4 is called for the first record of the transmission. The flag XTP4WINA is valid only when XTP4CONV is set ON and the printer is in Cut-Sheet Emulation mode.

Considerations and limitations of the 3800 Line-Mode Conversion function

Jobs with the following requirements cannot be printed in PSF until they are manually converted for AFP printing:

- COPYMODs

  PSF cannot process jobs that use COPYMODs. If MODIFY is specified in the JCL for a job, and that job is identified in Exit 4/14 for 3800 Line-Mode Conversion processing, PSF issues an error message and , in deferred-printing mode, requests that the system hold the job. If the same job is not identified in Exit 4/14, it would print as a normal AFP job; the MODIFY parameter is ignored. Jobs for which the MODIFY parameter is specified must be converted to AFP jobs manually.

- Customized fonts

  PSF does not convert 3800 character sets and GRAPHMODs. Like-named fonts must be available as AFP bounded-box fonts in the PSF resource library. PSF comes with AFP versions of all the character sets that are supplied with the 3800 printer. For any nonstandard character sets used in line-mode print jobs, equivalent AFP fonts must be created or substituted.

- Post-processing rotation

  You need to also examine jobs that use a special post-processing routine and character set to create the 3800 version of rotated print. Although you might be able to convert the rotated character set for the AFP printer, removing the rotate post-processing step and using standard AFP fonts and AFP capabilities to rotate the job is easier.

The following restrictions apply to 3800 Line-Mode Conversion:

- 240 pel

  You can use the 3800 Line-Mode Conversion only to print 3800 line mode jobs on 240-pel printers.

- One input bin

  The 3800 Line-Mode Conversion assumes that the printer has only one input bin and uses paper only from the first input bin identified by the printer.

The following functions can be used with 3800 Line-Mode Conversion, subject to some limitations:

- Printing on cut-sheet AFP printers

  You can print 3800 line-mode jobs on cut-sheet AFP printers (such as the InfoPrint 60 printer) only if the print jobs fit on the paper that these cut-sheet printers handle. Although 3800 jobs that print on narrow paper easily fit on cut-sheet paper, 3800 jobs that are designed for wide paper probably do not fit. PSF does not rotate 3800 jobs on cut-sheet printers to try to make them fit the paper.

- Perf-to-perf printing when you are migrating from non-IBM 3800-like printers

  Support is provided to allow migration from non-IBM printers that are similar in operation to the 3800 printer, but are capable of printing print from perforation to perforation (no required half-inch margins on the top and bottom of the page). If the FCB contains a special X'0B' byte in the first byte of the line space/channel code descriptors, it is a nonstandard FCB (not created through the IEBIMAGE utility). In this case, PSF counts the top and bottom half-inch margins as a printable area when it compares expected and actual loaded paper size. PSF also always prints perf-to-perf when this FCB is used, regardless of the setting of XTP4YOFF. However, if character ascenders or descenders fall outside of the printable area, they are not printed.

An alternative for processing these jobs when you are migrating them to a 3900 printer is RPQ 8B3997—3800 Compatibility Mode for the 3900 printer. This RPQ provides a separate microcode load for the 3900 printer to enable it to run as a 3800-3 or 3800-6 printer, operating in 3800 line mode. This RPQ is intended to facilitate the migration of 3800 workloads to 3900 printers with no need to convert COPYMODs, GRAPHMODs, and JCL, or to install later levels of software. For more information about the RPQ, consult your IBM marketing representative.

Implementation checklist for PSF 3800 Line-Mode Conversion

To use the PSF conversion function, take the following steps:

- Make sure that AFP fonts are available in the PSF resource library for all the 3800 character sets and GRAPHMODs that are used by the jobs you are processing.
- Make sure that electronic overlays are created and available in the PSF resource library for all 3800 forms flashes used.
- Screen the 3800 jobs to identify the jobs that cannot be processed by PSF; that is, jobs that use COPYMODs or merging lines with different-sized characters.
- Modify PSF Exit 4/14 to indicate jobs that require the conversion function.

You are now ready to route 3800 line-mode jobs to an AFP printer.

Considerations when using AFP Download Plus

Paper length and width are specified with two parameters that determine the printable area to be used. The parameters are:

- **paper-length** and **paper-width** AFPPARMS parameters specified by using the AFPPARMS control statement in the startup procedure or the AFPPARMS control statement on the OUTPUT statement.
- **Paper length** and **Paper width** parameters specified by using the Printer Inventory.

The default for paper length is 14 inches and the default for paper width is 13.2 inches. If the FCB printable area is within the parameter defaults, the page length and page width parameters do not need to be specified.

One of these can cause off-the-logical-page errors to occur when printing:

- The paper length or paper width parameter is set too small.
- The paper that is loaded at the printer is not the correct size.

Message APS973I indicates that the page defined in the FCB does not match the physical paper length. However, this message is not issued unless XTP7LMCM is set ON in installation Exit 7 or **Report Line-Mode Conversion paper-length errors** is set to YES in the Printer Inventory.

AFP Download Plus assumes that cut-sheet emulation is not being used. Therefore, AFP Download Plus does not accept XTP4WINA (wide paper flag) in installation Exit 4 because it is only used when cut-sheet emulation is active.

Chapter 29. Supporting AFP output on microfilm

PSF supports sending Advanced Function Presentation (AFP) data to microfilm[18] devices in deferred-printing mode. PSF does not support microfilm for z/OS users who are using:

- PSF direct
- Direct Print mode
- Distributed print function (DPF)

After you establish a PSF startup procedure for microfilm, the microfilm device appears to PSF to be a continuous-forms N_UP printer operating in simplex mode.

Note: This information assumes that you know how to configure PSF so a PSF user can print AFP data on paper output.[19] Based on this assumption, additional prerequisites and information are presented so that you can set up PSF users to also print AFP data to microfilm in the z/OS environment. Just as you might need to refer to other documentation for printer-specific information, you might need to refer to documentation supplied by the manufacturer of the microfilm device.

PSF hardware attachment requirements for microfilm devices

PSF supports a microfilm device attached to the host by specific protocols:

- Parallel channel
  - Block multiplexer
  - Data streaming
  - Two-channel switch—optional
- Enterprise Systems Connection (ESCON)

PSF system programmer tasks for microfilm devices

This information presents a visual representation of how PSF and a microfilm device work together to print AFP information to microfilm. The rest of this information describes the process by for setting up a user of PSF to print AFP jobs to microfilm.

Figure 99 on page 342 shows a possible configuration for a microfilm system. This figure represents the flow of an AFP document and related resources through PSF into the microfilm device.

---

[18] *Microfilm* can mean either microfiche or 16 mm film.

[19] If you are not familiar with configuring PSF, IBM recommends that you read Chapter 1, "Understanding PSF for z/OS," on page 1 and the information about defining channel-attached printers, ESCON-attached printers, and SNA-attached printers that use IBM token rings, and the chapters that describe working with libraries, changing the system environment, using PSF parameters, using deferred-printing mode, managing resources and selecting special processing parameters, understanding PSF data protection and security, using installation exits, and understanding AFP performance. If you are not familiar with PSF vocabulary terms, you might want to check the glossary.
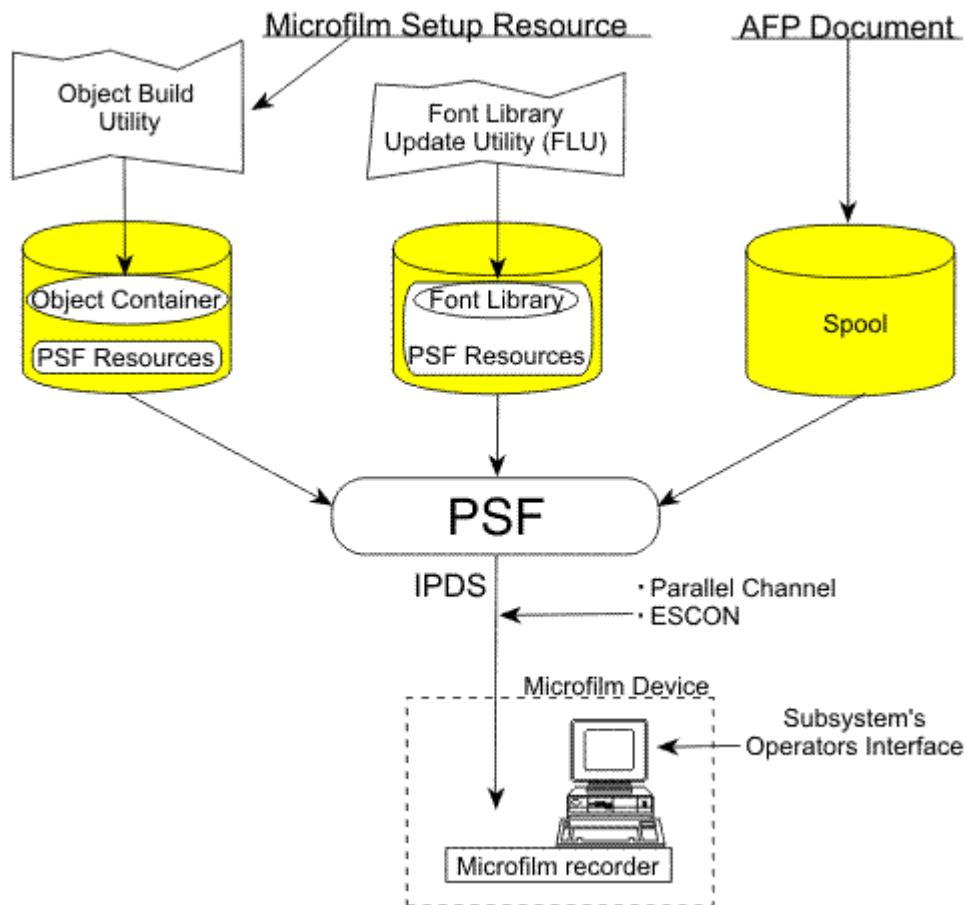
Figure 99. Example of AFP data being transmitted to a microfilm device

Even though your system is configured and installed as instructed by the manufacturer of the microfilm device, you still have some tasks to do before the microfilm device and PSF can work together so PSF users can print AFP jobs on microfilm:

- Attach and define the microfilm device to z/OS or VTAM.
- Define the microfilm device to JES2 or JES3.
- Define the microfilm device to PSF.
- Establish a PSF startup procedure.
- Work with the libraries required for microfilm jobs.
- Manage PSF resources and selecting special processing parameters for microfilm jobs.
- Manage PSF data protection and security for microfilm jobs.
- Use PSF installation exits for microfilm jobs.
- Understand PSF performance issues as they pertain to microfilm.
- Respond to PSF errors reported by a microfilm device.
- Redirect PSF messages for microfilm jobs.
- Recover or restart AFP print jobs for microfilm output.

Defining the microfilm device as channel-attached or ESCON-attached

The microfilm device can be channel-attached or ESCON-attached. An ESCON-attached microfilm device can be locally attached to an ESCON channel or a FICON Bridge channel in an ESCON Director. For more information about the FICON Bridge feature, see *Planning for 9032 Model 5 Director*.

With this in mind, you must attach the microfilm device and define it to z/OS, JES2 or JES3, and PSF.

Defining the microfilm device to z/OS

Whether the microfilm device is channel-attached or ESCON-attached, follow the procedures for attaching and defining channel-attached printers to z/OS (as explained in Chapter 3, "Using channel-attached printers with PSF," on page 9) by entering AFP1 as the unit type. An ESCON-attached microfilm device can be locally attached to an ESCON channel or a FICON Bridge channel in an ESCON Director. For more information about the FICON Bridge feature, see *Planning for 9032 Model 5 Director*.

Defining the microfilm device to JES2 and JES3

Just as you coded JES initialization statements to define PSF channel-attached and SNA-attached printers, you must also code JES initialization statements to define the microfilm device.

The microfilm device requires that you run with checkpointing turned OFF when printing. Therefore, avoid using the CKPTPAGE and CKPTSEC JCL parameters for microfilm jobs:

- For JES2, specify CKPTMODE=SEC and CKPTSEC=0.
- For JES3, specify CKPNTPG=32767.

To avoid printing unneeded microfilm frames and to improve performance, consider the following suggestions:

- Specify the Non-Process Run-Out parameter (NPRO).

  Doing the NPRO might take significantly longer on a microfilm device than on a printer; it depends on the microfilm device used.

  When you are specifying a nonzero NPRO value, consider these:

  - While PSF is doing an NPRO, no jobs are selected for printing. Selection of jobs for printing resumes after the NPRO is complete.
  - The Missing Interrupt Handler timeout value needs to be set to a value twice as long as the time to do an NPRO.
  - PSF issues an NPRO before the device is drained.
  - If a printer setup, such as a forms change, is requested through JES, the setup message is not issued to the operator until the NPRO is complete.

  When you are specifying an NPRO value of zero, consider these:

  - When PSF receives a request to drain the microfilm device, PSF does not make sure that all pages are stacked. It marks as complete all jobs sent to the microfilm device. If an error occurs after the device is drained, and the microfilm is lost, the job must be run again.
  - PSF does not query the microfilm device when it has no work to do. Therefore, a job that is stacked might not be released to JES as complete (and purged off the queue) until a subsequent job begins processing, or the microfilm device is drained.
  - If the microfilm device is powered off or the PSF address space is canceled, PSF releases as incomplete all jobs that were not stacked at the time PSF last queried the device. Unless they are manually purged from the queue, all these jobs are reprinted from the beginning (because no checkpoints exist).

- – An NPRO is done even if a printer setup is requested through JES (for example, a forms change). The setup message is not issued to the operator until the NPRO is complete.
- Recommend that printer operators not initiate system changes for different resolutions from the microfilm operator panel.
- Understand that users who are sending AFP data to a microfilm device might not want to print header, trailer, or separator pages.
- Specify NONE for the **Print error reporting** parameter in the Printer Inventory or specify DATACK=BLOCK in your PRINTDEV statement to prevent data checks from being reported (see Print error reporting).

For more information about defining printers to JES, see Chapter 9, "Using deferred-printing mode," on page 123.

Defining the microfilm device to PSF

You create two separate PRINTDEVs: one for the jobs with 120-pel and 240-pel resolution, and one for those jobs with 300-pel resolution.[20] The 240-pel resolution PRINTDEV uses 240-pel resources, [21] and the 300-pel resolution PRINTDEV uses 300-pel resources. Both PRINTDEVs are defined to drive the same microfilm device; for a specific microfilm device, however, only one PRINTDEV can be active at a time. You define separate classes or destinations for each PRINTDEV. Your installation must route the appropriate jobs to the appropriate class or destination. The two PRINTDEVs can be defined in the same (or separate) startup procedures.

Figure 100 on page 346–Figure 101 on page 347 show the sample startup procedure supplied with PSF for microfilm. It contains all the JCL parameters that you need to produce a startup procedure for print jobs with 240-pel or 300-pel resolution, but all the references to 300-pel resolution are commented out. To create separate startup procedures, make two copies of the sample startup procedure for microfilm, and keep one unchanged copy as your startup procedure for 120-pel and 240-pel resolution. In the second copy, delete or comment out all the references to 240-pel resolution, and make all the references to 300-pel resolution active. This changed copy is your 300-pel resolution startup procedure.

PRINTDEV parameters for microfilm devices

In Figure 101 on page 347, notice how the PSF PRINTDEV control statements are used:

```
COMSETUP=H1SETUPD
FONTDD=*.FONT01 or FONTDD=*.FONT02
OBJCONDD=*.OBJCON01
```

PSF requires the following keywords for all microfilm devices:

COMSETUP=*membername*
COMSETUP=*membername* is a microfilm setup resource parameter that specifies a member name. The manufacturer of the microfilm device provides a utility for you to use when you are generating member names. The microfilm setup resource parameter is used only for sending output to a microfilm device. You can also specify this parameter in the Printer Inventory. See Com setup member.

---

[20] Although doing so is not recommended, you can drive the microfilm device with just one startup procedure, if either of the following conditions is met:

- All the resources in the resource libraries are resolution-independent, and all the jobs submitted that use inline resources or user libraries contain resources that are resolution-independent.
- All submitted jobs that request a resolution that is different from the resolution-dependent resources in the PSF system and security libraries must contain the needed resources at the requested resolution, either as inline resources or in the user's library.

[21] Even if a PSF user requests 120-pel resolution for an AFP print job being sent to a microfilm device, PSF requires 240-pel resources from the library. Do not store 120-pel resources in the library for microfilm devices.

Note: IBM recommends that you use H1 as a prefix to every member name for the microfilm setup resource parameter. In Figure 101 on page 347, for example, H1SETUPD is specified as a default member name for the microfilm setup resource parameter. If you want the member name to be prefixed with H1, you must include H1 as part of the name you specify; PSF does not add a prefix.

As Figure 100 on page 346 and Figure 101 on page 347 show, the COMSETUP=*membername* is also used in the sample OUTPUT statement for microfilm. Although you are advised to define the microfilm device to JES without job headers, job trailers, and data set separator pages, and to turn off messages to save microfilm frames, you might elect to print this information to microfilm. Without a COMSETUP, the microfilm device cannot print the headers, trailers, data set separators, or messages to microfilm.

OBJCONDD=*.*label
OBJCONDD=*.*label* identifies the DD statement that specifies the object container library.[22] This keyword follows the rules of any other PSF resource library reference on the PRINTDEV statement; for the syntax, see Table 8 on page 102.

Before libraries are referenced by the FONTDD, the Font Conversion Utility needs to be run against them. For more information, see "Using the Font Library Update (FLU) utility" on page 348.

---

[22] Microfilm setup is a type of object container resource. PSF uses a single object container system resource library, designed to contain various types of object container resource.

Sample JCL startup procedure for microfilm devices

```
//APSWPROM PROC
//**************** THE PSF STARTUP PROCEDURE ******************
//*
//*01* MODULE-NAME = APSWPROM
//* $MOD(APSWPROM) COMP(APS) PROD(PSF) : RELEASE 4.6.0
//*
//*01* DESCRIPTIVE-NAME = START PROCEDURE FOR MICROFILM
//*                       DEVICES
//*
//*01* STATUS = VERSION 4, RELEASE 6, LEVEL 0
//*
//*01* FUNCTION = SET PSF OPTIONS AND RESOURCES
//*
//*01* NOTES = THE FULL NAME OF THE DEFAULT PAGEDEF IS
//*            P1A06462.
//*            THE FULL NAME OF THE DEFAULT FORMDEF IS
//*            F1A10110.
//*            THE FULL NAME OF THE DEFAULT CORE FONT IS
//*            X060D8. THE FULL NAME OF THE EQUIVALENT
//*            COMPATIBILITY FONT IS XOGT15.
//*            THE FULL NAME OF THE SEPARATOR PAGE PAGEDEF IS
//*            P1V06483.
//*            THE FULL NAME OF THE SEPARATOR PAGE CORE FONT IS
//*            X060D8. THE FULL NAME OF THE EQUIVALENT
//*            SEPARATOR PAGE COMPATABILITY FONT IS X0GT15.
//*
//*01* REQUIRED ACTIONS =
//*      1) DEFAULT COMSETUP - DEFAULT COMSETUP FILES ARE SPECIFIED
//*            IN THIS STARTUP PROCEDURE BUT ARE NOT PROVIDED WITH
//*            PSF.  DEFAULT COMSETUP FILES MUST BE GENERATED
//*            WITH SOFTWARE PROVIDED WITH YOUR MICROFILM DEVICE AND
//*            PLACED IN THE DATA SET SPECIFIED ON THE OBJCON01 DD
//*            IN THIS START PROCEDURE.
//*
//*            THE FULL NAME OF THE DEFAULT COMSETUP IS H1SETUPD.
//*            THE FULL NAME OF THE SEPARATOR PAGE COMSETUP IS
//*            H1SETUPS.
//*
//*      2) RESOLUTION - THIS START PROCEDURE IS SET UP FOR DRIVING
//*            THE MICROFILM DEVICE AT 240-PEL RESOLUTION.  TO CHANGE IT
//*            TO DRIVE 300-PEL RESOLUTION MICROFILM DEVICES, COMMENT
//*            OUT THE FIRST PRINTDEV STATEMENT AND UNCOMMENT THE
//*            SECOND PRINTDEV STATEMENT.
//*
//*01* CHANGE-ACTIVITY :
//* $00=OW?????,  HPRF220, 102494 BJ13JPG : Microfilm Support
//*$DR=LAPS0007, HPRF310,050698 BDKUMAL : Default to core font
//*
//**** END OF SPECIFICATIONS ***/
```

Figure 100. Sample JCL startup procedure for a microfilm device (APSWPROM) (Part 1 of 2)

```
//STEP01   EXEC PGM=APSPPIEP,REGION=4M
//JOBHDR OUTPUT PAGEDEF=V06483,   /* JOB SEPARATOR PAGEDEF      */
//        FORMDEF=A10110,CHARS=60D8, /* JOB SEPARATOR FORMDEF    */
//        COMSETUP=H1SETUPS         /* JOB SEPARATOR COMSETUP     */
//JOBTLR OUTPUT PAGEDEF=V06483,   /* JOB SEPARATOR PAGEDEF      */
//        FORMDEF=A10110,CHARS=60D8, /* JOB SEPARATOR FORMDEF    */
//        COMSETUP=H1SETUPS         /* JOB SEPARATOR COMSETUP     */
//DSHDR  OUTPUT PAGEDEF=V06483,   /* DATA SET SEPARATOR PAGEDEF  */
//        FORMDEF=A10110,CHARS=60D8, /* DATA SET SEPARATOR FORMDEF */
//        COMSETUP=H1SETUPS         /* DATA SET SEPARATOR COMSETUP */
//MSGDS  OUTPUT PAGEDEF=A06462,   /* MESSAGE DATA SET PAGEDEF    */
//        FORMDEF=A10110,          /* MESSAGE DATA SET FORMDEF    */
//        PIMSG=NO,CLASS=A         /* SEND MSG DATA SET TO CLASS A */
//FONT01   DD   DSN=SYS1.FONTLIBB, /* SYSTEM FONTS - 240 PEL     */
//         DISP=SHR
//FONT02   DD   DSN=SYS1.FONT300,  /* SYSTEM FONTS - 300 PEL     */
//         DISP=SHR
//PSEG01   DD   DSN=SYS1.PSEGLIB,  /* SYSTEM PAGE SEGMENTS       */
//         DISP=SHR
//OLAY01   DD   DSN=SYS1.OVERLIB,  /* SYSTEM OVERLAYS            */
//         DISP=SHR
//PDEF01   DD   DSN=SYS1.PDEFLIB,  /* SYSTEM PAGEDEFS            */
//         DISP=SHR
//FDEF01   DD   DSN=SYS1.FDEFLIB,  /* SYSTEM FORMDEFS            */
//         DISP=SHR
//OBJCON01 DD   DSN=SYS1.APSOCLIB, /* SYSTEM OBJECT CONTAINERS   */
//         DISP=SHR                /*                            */
//PRT1     CNTL
//PRT1     PRINTDEV FONTDD=*.FONT01, /* 240 PEL FONT LIBRARY DD  */
//*                                /* <-- SEE REQUIRED ACTIONS   */
//*                                /*     ABOVE                  */
//*PRT1    PRINTDEV FONTDD=*.FONT02, /* 300 PEL FONT LIBRARY DD  */
//*                                /* <-- SEE REQUIRED ACTIONS   */
//*                                /*     ABOVE                  */
//         OVLYDD=*.OLAY01,         /* OVERLAY  LIBRARY DD        */
//         PSEGDD=*.PSEG01,         /* SEGMENT  LIBRARY DD        */
//         PDEFDD=*.PDEF01,         /* PAGEDEF  LIBRARY DD        */
//         FDEFDD=*.FDEF01,         /* FORMDEF  LIBRARY DD        */
//         OBJCONDD=*.OBJCON01,     /* OBJECT   LIBRARY DD        */
//         JOBHDR=*.JOBHDR,         /* JOB HEADER SEPARATOR OUTPUT */
//         JOBTRLR=*.JOBTLR,        /* JOB TRAILER SEPARATOR OUTPUT */
//         DSHDR=*.DSHDR,           /* DATA SET HEADER OUTPUT      */
//         MESSAGE=*.MSGDS,         /* MESSAGE DATA SET OUTPUT     */
//         BUFNO=5,                 /* NUMBER OF WRITE DATA BUFFERS */
//         PAGEDEF=A06462,          /* DEVICE PAGEDEF DEFAULT      */
//         FORMDEF=A10110,          /* DEVICE FORMDEF DEFAULT      */
//         COMSETUP=H1SETUPD,       /* DEFAULT COMSETUP MEMBER     */
//         CHARS=(60D8),            /* DEVICE DEFAULT FONT         */
//         PIMSG=YES,               /* ACCUMULATE DATA SET MESSAGES */
//         DATACK=BLOCK             /* UNBLOCK DATA CHECKS         */
```

Figure 101. Sample JCL startup procedure for a microfilm device (APSWPROM) (Part 2 of 2)

The startup procedures specify PSF initialization parameters and libraries that contain system and installation resources for microfilm. For more information, see "PSF startup procedures for AFP printers" on page 135.

Redirecting message data sets for microfilm jobs

IBM suggests that users not print PSF message data sets on microfilm. For more information, see "Redirecting the message data set" on page 328. In the APSWPROM sample that is provided, the message data sets are redirected to class A.

Working with resource libraries required by microfilm devices

The resources that you use and store in PSF libraries for microfilm devices are the same resources as other printers use, plus microfilm setup resource object containers.

Working with the microfilm setup resource object container library

Microfilm setup is a category of resource called an *object container*,[23] which is stored in a PSF-specified host object container library, in a user library, or inline in the AFP data. You specify a default microfilm

setup resource parameter in the Printer Inventory (see Com setup member) or the PRINTDEV statement (see COMSETUP); PSF users can also specify the COMSETUP parameter with JCL. The microfilm setup resource parameter (COMSETUP) is described in *PSF for z/OS: User's Guide*. If no default parameter is specified in the Printer Inventory or the PRINTDEV statement, a COMSETUP keyword must be specified in every separator and user data set. If any are encountered without a COMSETUP keyword, PSF stops processing.

You must already install or create and name resource libraries that contain resources that are needed for printing to paper media. You must also create and name a resource library to contain the object containers, which are required for printing to microfilm media. No default library is provided with PSF.

Note: Because a microfilm setup resource is one type of object container resource, and PSF does not enforce a prefix for the eight-character name of the microfilm setup resource, you define a naming convention to avoid future conflicts with additional object container resources. IBM recommends a prefix of H1 for all microfilm setup resources.

Working with FOCA font libraries

Every time a FOCA font is added, deleted, or changed in a library, you must run an IBM command-line utility called the Font Library Update (FLU). For more information, see "Using the Font Library Update (FLU) utility" on page 348.

Another requirement for printing to microfilm is that font libraries must be available for the resolutions specified. The fonts with 240-pel resolution are used whenever a user specifies 120-pel or 240-pel resolution for microfilm.

The sample startup procedure provided in SYS1.SAMPLIB(APSWPROM), represented by Figure 100 on page 346 - Figure 101 on page 347, identifies both FONT01 (to support 120-pel and 240-pel resolution) and FONT02 (to support 300-pel resolution) for printing to microfilm.

Managing resources and selecting special processing parameters

Microfilm support requires the following tasks, in addition to the tasks for setting up printers:

- Using a utility provided with the microfilm device, create microfilm setup resource object containers.
- Using the Font Library Update (FLU) utility, update the font libraries.

Generating microfilm setup resource object containers

Run the microfilm utility that is provided by the manufacturer of your microfilm device to generate microfilm setup resource object containers. For more information about using a utility to generate object containers, see the instructions provided with your microfilm device.

The utility that is provided with the microfilm device checks the syntax of the setup parameters and places the setup information into a microfilm setup resource object container,[24] which is associated through JCL keywords with the AFP print job data.

A PSF system library is required for storing the object containers. For more information, see Chapter 6, "Working with libraries," on page 49.

Using the Font Library Update (FLU) utility

Microfilm devices require that all FOCA fonts that are downloaded from the host contain global resource ID (GRID) information. If a requested font does not contain GRID information, the page or data set that uses that font is not printed.

---

[23] Microfilm setup resources are generated by running a utility, provided with the microfilm device. For more information, see "Generating microfilm setup resource object containers" on page 348.

[24] The object container is not an AFP-managed resource; it is a MO:DCA architecture resource.

GRID information uniquely describes code page and character set characteristics, such as style, size, set of characters, and valid code points. Font resolution does not affect the GRID value.

IBM and non-IBM fonts typically do not contain GRID information, although space for that information is reserved. The FLU uses system tables and user-modifiable tables to place needed GRID information in fonts that do not contain complete GRID information. The FLU uses the same Name-to-GRID user mapping table (APSRNTGM) that PSF uses (for more information about how to make a Name-to-GRID user mapping table, see "Using the font mapping tables" on page 237). The user mapping table is searched first when insufficient GRID information is found in a code page or character set. The FLU also generates GRID values from the user range of GRID values and from information in the font, if no entry is found for the font in the system or user mapping table and no font member is found that has the same name and contains complete GRID information.

Considerations when using the FLU

Before you run the FLU, read the following key information:

- The FLU does not use user mapping table entries with an FGID or a CPGID that have a value of either X'0000' or X'FFFF'.
- To prevent the FLU from assigning duplicate GRID information to different fonts, you must use a single invocation to run the FLU against all fonts that are to be used with the microfilm device[25]—including libraries of different resolutions, font types (for example, raster fonts and outline fonts), system font libraries, security font libraries, font libraries that are to be used by ACIF, and font libraries accessed with the USERLIB keyword.
- If code pages and character sets in different libraries have the same member names, PSF assumes that they are the same code page or character set and are updated with identical GRID information. If one member contains complete GRID information, and another member with the same name does not, the incomplete member is updated with the additional information.
- When fonts are added or changed on the system after the FLU runs successfully, the FLU must be rerun with all the existing fonts, plus any new fonts. The old fonts are not changed by the FLU on successive runs, but they are used both to propagate GRID information when members with the same name are encountered and to determine what GRID values in the user range are available for assignment.
- The following members of font data sets are not updated by the FLU:
  - Coded fonts
  - Unbounded box fonts (for the 3800 printer)
  - Code pages with complete GRID information
  - Character sets with complete GRID information
  - Non-font resources
  - Non-AFP resource data
- Because the FLU updates fonts in place, no programs attempt to access the fonts and no PSF FSA are ACTIVE while the FLU is running.
- The attributes of all font data sets must have these requirements (the same as for PSF):
  - Partitioned (not sequential)
  - Variable (not fixed)
  - Non-spanned
  - Contain carriage control (ANSI or MACHINE)
- If the user-modifiable name-to-GRID mapping table, APSRNTGM, is used, it must be link-edited with AMODE=24, RMODE=24.

---

[25] The number of font data sets that can be input to the FLU on a single invocation is limited only by the operating system.

Running the FLU

A sample job, APSWRFLU, is provided in SYS1.SAMPLIB (see). A DD statement for each font library on your system that is to be used by the microfilm device must be added to the JCL. Each font DD name must begin with FONT, to identify it to the FLU as the name of a font library. The remaining characters in the DD have no significance to the program. Concatenated data sets for a single DD are not permitted.

```
//APSWRFLU JOB 'account #','name',MSGLEVEL=(1,1)
//* ****************************************************************
//*                                                               *
//*  USE THIS JOB TO UPDATE ALL FONT LIBRARIES TO BE USED WITH    *
//*  A MICROFILM DEVICE.                                          *
//*                                                               *
//* ****************************************************************
//STEP1    EXEC PGM=APSRFLU
//SYSPRINT  DD SYSOUT=*
//FONT01    DD DSN=SYS1.FONTLIBB,DISP=OLD
//FONT02    DD DSN=SYS1.FONT300,DISP=OLD
//
```

Figure 102. APSWRFLU sample job provided in SYS1.SAMPLIB

For example, if fonts also existed in COMMON.FONTLIB, PSF.V130.FONTLIB, USER.MY.FONTLIB, and SYS1.SECURE.FONTLIB, the DDs in are added to the JCL:

```
//FONT03    DD DSN=COMMON.FONTLIB,DISP=OLD
//FONT130   DD DSN=PSF.V130.FONTLIB,DISP=OLD
//FONTUSR1  DD DSN=USER.MY.FONTLIB,DISP=OLD
//FONTSEC   DD DSN=SYS1.SECURE.FONTLIB,DISP=OLD
```

Figure 103. DDs added to the APSWRFLU sample job for existing fonts

FLU return codes

The FLU returns only three valid return codes:

0

The FLU runs successfully, and the appropriate font members are updated. For information about the updates that are performed, see the informational messages in the SYSPRINT data set.

4

The FLU updates all the members that it can, but found fonts that are not valid and cannot be updated. The not valid fonts cannot be used with the microfilm device until they are fixed and the FLU is run again. For information about the font members in error and the updates that are performed, see the informational messages in the SYSPRINT data set.

16

The FLU updated some or no members. One or more font data sets input to the FLU is not a valid font data set. See the informational messages in the SYSPRINT data set for information about the data set in error.

If any other return code is returned, the FLU encountered a logic error. Consult your service representative in the IBM support center. The FLU has error checking and data isolation to avoid contamination of font data if a logic error occurs and it cancels processing. Fix the error and rerun the FLU.

Printing to microfilm

After you attached the microfilm device, provided a startup procedure, run the utility that is provided by the manufacturer of your microfilm device, and run the FLU utility, users can send AFP data sets to the microfilm device to be printed on microfilm.

Operating a microfilm device

If you have two microfilm devices, one with 240-pel resolution and one with 300-pel resolution, you can assign a different destination or class to each of them. You must tell the PSF users what DEST statements to use to direct their output to the destination with the resolution they want.

To switch from one startup procedure (or PRINTDEV) to the other, use this procedure:

- Drain the active PRINTDEV by using a normal printer stop command: *$P PRTxxx* in JES2, or **V,PRTxxx,OFFLINE* in JES3.
- Start the other PRINTDEV by using a normal printer start command: *$S PRTyyy* in JES2, or **V,PRTyyy,ONLINE* in JES3.

Note: To turn off the microfilm device for normal completion of processing (for example, at the end of the day or for planned maintenance), drain the active PRINTDEV by using a normal printer stop command before you power off the printer.

Some operator commands are not supported for correct operation with microfilm. Tell the operator not to use the following:

- JES $I (Interrupt) command
- JES $F (Space Forward) command
- JES $B (Space Backward) command (unless instructed to do so by the microfilm device)

At some point, the microfilm device instructs you to do operator actions; correct operation is dependent upon your doing so.

Microfilm data protection and security considerations for PSF

No changes in the procedures for data protection and security considerations are made for printing to microfilm. The microfilm device is handled in the same way as an AFP1 family printer. For information, see Chapter 18, "Understanding PSF data protection and security," on page 219.

Using PSF installation exits for microfilm devices

One of the tasks of system programmers is writing, changing, and installing exit routines. For more information, see Chapter 25, "Using installation exits," on page 257.

Understanding PSF performance issues that pertain to microfilm devices

IBM recommends that you define no more than two active microfilm devices (PRINTDEVs) in a single startup procedure because of the large number of pages these devices produce.

Responding to PSF errors reported by microfilm devices

Handle PSF errors reported by the microfilm device in the same way as you handle PSF errors reported by other printers. For more information, see Chapter 26, "Recovering from print errors," on page 325.

Recovering or restarting AFP print jobs for microfilm

The restart and rerun processes have joint PSF and microfilm device interactions. See the documentation provided with your microfilm device. The microfilm device guides you through the restart and rerun process, and instructs the operator to issue commands to PSF when needed.

Appendix A. Migrating PSF parameters to the Printer Inventory

This information describes how to use the Printer Inventory migration program, **aopmig**, to copy existing PRINTDEV parameters, FSS EXEC PARM statements, and AFPPARMS control statement parameters to new FSA and FSS definitions in the Printer Inventory.

The **aopmig** migration program and sample JCL in SYS1.SAMPLIB members, AOPMIGRX and AOPPIDU, are included with Infoprint Server.

Notes:

1. You must manually migrate parameters from Exit 7.

2. The **aopmig** migration program does not support AFPPARMS data set names that include JCL symbols, such as &SYSNAME. For example, parameters in a data set with the name **FSS.PDS.&SYSNAME.AFPPARMS** are not migrated.

3. Because **aopmig** is part of Infoprint Server, **aopmig** fixes are included with that product. Service issues with **aopmig** are directed to Infoprint Server.

Migrating existing printer information is a multi-step process:

1. Run the migration program to read existing printer information and generate Printer Inventory definition statements. The migration program also generates a report of its processing, which you can examine for possible error conditions.

2. Run the Printer Inventory Definition Utility (PIDU) program to create FSA and FSS definitions in the Printer Inventory. The PIDU program uses the Printer Inventory definition statements generated by the migration program as input.

3. Edit Infoprint Server ISPF panels to add options for Exit 7 INIT calls to the FSA definitions in the Printer Inventory.

Because the migration program generates only text output files and does not modify input files or create any entries in the Printer Inventory, you can rerun the migration program as often as you want before you run the PIDU program to create the Printer Inventory from the output of the migration program.

You can also rerun the migration program after you run the PIDU program; however, you must first delete the entries in the Printer Inventory that are created the first time that you run the PIDU program (see for more information).

After you run the PIDU program to create entries in the Printer Inventory, you can use Infoprint Server ISPF panels to browse and optionally edit the new FSA and FSS definitions.

Running the migration program

These sources are input to the migration program:

- PRINTDEV statements in the PSF startup procedure
- EXEC statement PARM parameters in the PSF startup procedure
- AFPPARMS control statement if you have the AFP Download Plus feature installed
- JES initialization statements

Before you run the migration program, you must install Infoprint Server, but you do not need to customize it or have it running.

You can run the migration program from the z/OS UNIX System Services shell by using the **aopmig** command or from TSO by using the AOPBATCH utility program or the BPXBATCH utility program.

Running the migration program using the aopmig command

Format

```
aopmig [-i 'JES_initialization_dataset']… [-x 'PSF_procedure_dataset']…
```

Description

aopmig reads existing PRINTDEV parameters, FSS EXEC PARM parameters, and AFPPARMS control statement parameters and generates Printer Inventory definition statements that the Printer Inventory Definition Utility (PIDU) program can use to create FSA and FSS definitions in the Printer Inventory. **aopmig** writes the Printer Inventory definition statements to `stdout` and writes a report of its processing to `stderr`.

Options

-i '*JES_initialization_dataset*'
> The name of the data set that contains JES initialization statements for PSF. Enclose the name in single quotation marks if the data set name contains special characters, such as ( ) $, that might be interpreted by the shell. If you have more than one initialization data set, specify this option and value more than once. If all definitions for PSF are in one data set, you need to only specify this option and value one time.
>
> The migration program uses JES initialization statements to determine FSS names and the relationship between FSS and FSA entities. If you do not specify this option, the migration program assigns the PROC name to each FSS and creates an FSA definition for each PRINTDEV statement.

-x '*PSF_procedure_dataset*'
> The name of the data set that contains the PSF startup procedure. Enclose the name in single quotation marks if the data set name contains special characters, such as ( ) $, that might be interpreted by the shell. If you have more than one startup procedure, specify this option and value more than once. If all parameters you are migrating are in one startup procedure, you need to only specify this option and value one time.

Usage notes

1. The migration program writes the Printer Inventory definition statements that it generates to its `stdout` file. It writes its report to its `stderr` file. By default, both `stdout` and `stderr` are directed to the terminal. In most cases, you redirect the output to files by using the UNIX redirection symbol (> for `stdout` and 2> for `stderr`).

2. You can rerun the migration program as often as you want before you run the PIDU program to create the Printer Inventory because the migration program generates only text output files and does not modify input files or create any entries in the Printer Inventory.

3. Save the `stdout` file from the migration program until you are satisfied with the entries in the Printer Inventory. If you ran the PIDU, you might need to edit the `stdout` file before you can rerun the migration program. See for more information.

Example

Enter this command to migrate the PRINTDEV and EXEC PARM statements in two PSF startup procedures:

```
aopmig -x 'SYS1.PROCLIB(PSFPROC1)' -x 'SYS1.PROCLIB(PSFPROC2)' \
       -i 'SYS1.PARMLIB(PRTRS)'
```

Note: The backslash (\) indicates that this is a multi-line command. You must start a new line immediately following the backslash; do not type any spaces or characters between the backslash and the new line.

Running the migration program from TSO

You can run the migration program from TSO by using the AOPBATCH utility, which IBM recommends, or the BPXBATCH utility. If you use the AOPBATCH utility, specify the PATH environment variable; specify the LIBPATH and NLSPATH environment variables only if your installation did not install Infoprint Server files in the default directories. If you use the BPXBATCH utility, specify the LIBPATH, NLSPATH, and PATH environment variables.

When you use the AOPBATCH or BPXBATCH utility, you can specify command line arguments by using the PARM parameter on the EXEC statement. However, because the value for the PARM parameter cannot exceed 100 characters, in most cases you use the environment variables that are shown in Table 38 on page 355 to specify input parameters to the migration program.

Table 38. Environment variables for running the migration program from TSO

Environment Variable	Comparable Command Line Option
AOPMIG_JES_PARMS	-i
AOPMIG_PSF_PRINTDEVS	-x

Note: The environment variables can have only one value. If you need to specify multiple input data sets for a migration program option, run the migration program from the z/OS UNIX System Services shell. Alternatively, you can combine all your JES initialization data sets into one data set and all your PSF procedure data sets into one data set. Then, you can specify the combined data sets when you run the migration program.

IBM provides sample JCL in the AOPMIGRX member of SYS1.SAMPLIB for running the migration program by using the AOPBATCH utility. Figure 104 on page 355 shows member AOPMIGRX.

```
//AOPMIGRX JOB MSGLEVEL=(1,1)
//*                                                              *
//*  THIS JCL WILL USE THE AOPBATCH UTILITY TO RUN THE aopmig    *
//*  PROGRAM TO MIGRATE PSF DEFINITIONS TO INFOPRINT SERVER.     *
//*  REPLACE hlq AND volume WITH VALUES APPROPRIATE FOR YOUR     *
//*  SYSTEM.  DO NOT MODIFY PARM='aopmig'.  ALSO SET THE VARIABLES *
//*  IN STDENV TO INDICATE THE DATA SET NAMES ON YOUR SYSTEM.    *
//*                                                              *
//**********************************************************************
//*
//MIGRT    EXEC PGM=AOPBATCH,PARM='aopmig',REGION=8M
//*
//STDOUT   DD  DSN=hlq.INVDEFS,
//             DISP=(NEW,CATLG,DELETE),
//             DCB=(RECFM=VB,LRECL=4096),
//             UNIT=SYSALLDA,VOL=SER=volume,SPACE=(TRK,(5,15,0))
//*
//STDERR   DD  SYSOUT=*
//*
//STDENV   DD  *
PATH=/usr/lpp/Printsrv/bin:/bin:
AOPMIG_PSF_PRINTDEVS=SYS1.PROCLIB(PSFPROC)
AOPMIG_JES_PARMS=SYS1.PARMLIB(PRTRS)
/*
```

Figure 104. Sample JCL for running the migration program from TSO — AOPMIGRX in SYS1.SAMPLIB

Using the migration program report

The migration program writes a report of its processing to `stderr`. The report is divided into sections. Each section corresponds to a processing step that the migration program does; the sections are written in chronological order. Each section begins with a line of asterisks (*). A line of dashes (-) separates entries within a section.

You can read this report to understand the processing that the migration does, or you can choose to scan it online to detect possible error conditions. These are some hints for reading or scanning the error report:

- Try reading or scanning the report online because it can be quite large.

- To find errors that are found by the migration program, search for the word `Error` in the report.
- To find warnings that are found by the migration program, search for the word `Warning` in the report.

Running the Printer Inventory Definition Utility (PIDU)

After you run the migration program and reviewing the report of its processing, run the Printer Inventory Definition Utility (PIDU) program to create FSA and FSS entries in the Infoprint Server Printer Inventory. Use the definition statements that the migration program generates and writes to `stdout` as input to the PIDU program.

Before you run the PIDU program, ensure that Infoprint Server is running; see *z/OS Infoprint Server Operation and Administration* or *z/OS Infoprint Server Printer Inventory for PSF* for information about how to start Infoprint Server.

You can run the PIDU program from the z/OS UNIX System Services shell by using the **pidu** command or from TSO by using the BPXBATCH or AOPBATCH utility program.

Running the PIDU program with the pidu command

To run the PIDU program, you can enter the **pidu** command from the z/OS UNIX Systems Services command line. Name the file that contains the Printer Inventory definition statements created by the migration program as input to the **pidu** command.

The PIDU program writes a report of its processing to `stderr`. By default, `stderr` is directed to the terminal. In most cases, you redirect this output to a file by using the UNIX redirection symbol (2> for `stderr`).

For example, if you entered the following command from the z/OS UNIX System Services command line to direct the Printer Inventory definition statements to a file named `defs`:

```
aopmig ... >defs 2>aopmig.rept
```

Then, enter the following **pidu** command and specify the `defs` file as input:

```
pidu <defs 2>pidu.rept
```

Running the PIDU program from TSO

You can run the PIDU program from TSO by using the AOPBATCH utility, which IBM recommends, or the BPXBATCH utility program. Name the file that contains the Printer Inventory definition statements created by the migration program in the STDIN DD statement.

If you use the AOPBATCH utility, specify the PATH environment variable; specify the LIBPATH and NLSPATH environment variables only if your installation did not install Infoprint Server files in the default directories. If you use the BPXBATCH utility, specify the LIBPATH, NLSPATH, and PATH environment variables.

When you use either the AOPBATCH or the BPXBATCH utility, also specify the AOPCONF environment variable if your Infoprint Server configuration file, `aopd.conf`, is not in the default directory; see *z/OS Infoprint Server Customization* for information about how to set these environment variables.

IBM provides sample JCL in the AOPPIDU member of SYS1.SAMPLIB for running the PIDU program by using the AOPBATCH utility. shows member AOPPIDU.

```
//AOPPIDU  JOB ,'pidu'
//*----------------------------------------------------------------
//*
//*  Sample JCL for running pidu in batch.
//*
//*  This JCL will use the AOPBATCH utility to run the pidu program
//*  to process Infoprint Server inventory definitions.
//*
//*  Replace hlq with the appropriate value.  Use the aopmigrt or
//*  aopmigrx job to create the hlq.invdefs data set.  You may
//*  edit the definitions in the data set as needed.
//*
//*  STDIN may also point to a sequential dataset containing other
//*  Printer Inventory Definition Utility (PIDU) statements.
//*
//*----------------------------------------------------------------
//*
//PIDU     EXEC PGM=AOPBATCH,PARM='pidu'
//*
//STDIN    DD  DSN=hlq.INVDEFS,DISP=SHR
//*
//STDOUT   DD  SYSOUT=*
//STDERR   DD  SYSOUT=*
//*
//* STDENV may point to a dataset containing environment variables.
//*       Built in values will work for the default installation.
//*STDENV  DD  DSN=environment,DISP=SHR
```

Figure 105. Sample JCL for running the PIDU program from TSO — AOPPIDU in SYS1.SAMPLIB

Rerunning the migration program and the PIDU program

You can rerun the migration program as often as you want before you run the PIDU program to create the Printer Inventory. However, if you need to rerun the migration program after you have already run the PIDU program, you must first delete the entries in the Printer Inventory that were created when you previously ran the PIDU program. To do this, follow these steps:

1. Edit the file that you used as input to the PIDU program. Change all **create** statements in this file to **delete** statements. For example, change this sample **create** statement:

   ```
   create fsa PRT1
       page-definition = A08682
     ;
   ```

   To this **delete** statement:

   ```
   delete fsa PRT1;
   ```

   You can use an **awk** command to create a file with **delete** statements. For example, if your input file is named defs, enter the following line from the z/OS UNIX System Services command line:

   ```
   awk '/^create/ { print "delete " $2 " " $3 " ;" }' <defs >delete.defs
   ```

   This program creates output file delete.defs with a **delete** statement for each **create** statement in the input file. See *z/OS UNIX System Services Command Reference* for information about the **awk** command.

2. Run the PIDU program by using the file with the **delete** statements as the input file. This step deletes the entries from the Printer Inventory that were created when you previously ran the PIDU program, but preserves any entries that your installation might have created with Infoprint Server ISPF panels. For example, enter:

   ```
        pidu <delete.defs 2>pidu.rept
   ```

3. Rerun the migration program **aopmig**.
4. Rerun the PIDU program.

Viewing and editing the Printer Inventory

After you run the PIDU program, you can use Infoprint Server ISPF panels to browse the new FSA and FSS definitions in the Printer Inventory. You can also edit the ISPF panels to add options for Exit 7 INIT calls to the FSA definitions in the Printer Inventory. See Figure 32 on page 63 for a sample ISPF panel, or see *z/OS Infoprint Server Operation and Administration* or *z/OS Infoprint Server Printer Inventory for PSF* for information about how to use the ISPF panels.

When you create new entries in the Printer Inventory, you can customize PSF to use the Printer Inventory definitions. See Chapter 7, "Using the Infoprint Server Printer Inventory," on page 53.

Appendix B. Printer information reports and system log output

You can save printer information in a data set specified in the PRTINFO DD statement in the PSF startup procedure (see "Requesting printer information with the PSF startup procedure" on page 201), or you can use the MVS MODIFY command to display printer information (see "Requesting printer information with the MODIFY command" on page 203).

When you save printer information in a data set, you can view the source data for the softcopy printer information report or format and print it as a hardcopy report. When you use the MODIFY command to display printer information, you view the information directly in the system log.

This information shows these examples:

- "Hardcopy report" on page 359
- "Softcopy report" on page 373

The sections you see in a printer information report depend on the printer; therefore, you might see sections in your printer information report that are different from the sections depicted in the examples.

"Softcopy record format" on page 386 shows all possible record formats for the softcopy report.

Hardcopy report

The hardcopy printer information report contains the same information as the softcopy report, but it is formatted so it is easier to read and it has page numbers. Figure 106 on page 360–Figure 119 on page 373 show an example of a hardcopy printer information report from an InfoPrint 4100 printer.

```
PSF Printer Information Report-4.7.0 Level 0000
Printer Information from PSFMVS.WTRES600.DEV6.PRTINFO(PRT619)
```

STM Data

```
FF4100C0 0000000D C2C3FF10 10014022
4400A004 0014C3C6 FF10A004 B001B002
B003C005 C100C101 0014C3C6 FF30A004
B001B002 B003C01E C01FC101 006EC4C3
FF106001 60026003 61016201 70017008
702E7034 706B707B 707E70CE 8008800A
800C80F2 80F480F6 90019002 90039004
90059007 9009900A 900D900E 90139015
901690F3 90F5E000 E001E002 E003F200
F201F202 F203F204 F205F206 F401F601
F602F804 F902FB00 FF01000C C9D4FF10
10014022 A0040026 C9D6FF10 10011202
40224401 50015003 50085080 50815082
51015204 5505A004 F300F301 0024C9D6
FF111001 12024022 44015001 50035008
500A5082 50835101 52045505 A004F301
0024C9D6 FF401001 12024020 44015001
50035008 50805081 50825101 52045505
A004F301 0022C9D6 FF421001 12024020
44015001 50035008 50205082 51015204
5505A004 F3010018 C9D6FF45 10011202
40204401 50205101 5505A004 F301000C
D6C30000 12015800 F301000C D6D3FF10
11021505 A0040008 D7E2FF10 1101000C
D7E3FF30 10014022 50FF0014 E5C7FF20
10014022 41004101 41024106 A004----
```

```
X'FF'    System/370 convention
X'4100'  Device type of the printer, or of the printer that
         is being emulated or mimicked
X'C0'    Model Number
```

BAR CODE COMMAND-SET VECTOR
```
X'000EC2C3FF10100140224400A004'
X'000E' 14      Length of the bar code command-set vector
X'C2C3' BC1 subset of the bar code command-set
X'FF10' BCOCA BCD1 data
X'1001' Bar code objects may be sent in any order
X'4022' Standard OCA color-support property ID
```

```
Date - 12/15/2019   Time - 12:41:2                              Page - 1
```

Figure 106. Hardcopy Printer Information Report (Page 1 of 14)

```
   Printer Information from PSFMVS.WTRES600.DEV6.PRTINFO(PRT619)

X'4400' Extended bar code color support; when this
        property pair is present, the printer supports the
        Color Specification (X'4E') triplet on WBCC-BCDD
X'A004' Object area orientation-support property ID

LOADED-FONT COMMAND-SET VECTOR
X'0014C3C6FF10A004B001B002B003C005C100C101'
X'0014' 20      Length of the loaded-font command-set
                vector
X'C3C6' Loaded-font command-set ID
X'FF10' LF1 subset ID - fully described font plus font
        index
X'A004' Orientation-support property ID
X'B001' Double-byte coded fonts supported
X'B002' Underscore width and position parameters in the
        LFI command are used by the printer
X'B003' GRID-parts fields allowed in the LFC, LFCSC, and
        LCPC commands
X'C005' Coded-font pattern-technology ID, Bounded-box
        raster-font technology
X'C100' Coded-font metric-technology ID, Fixed metrics
X'C101' Coded-font metric-technology ID, Relative metrics

LOADED-FONT COMMAND-SET VECTOR
X'0014C3C6FF30A004B001B002B003C01EC01FC101'
X'0014' 20      Length of the loaded-font command-set
                vector
X'C3C6' Loaded-font command-set ID
X'FF30' LF3 subset ID - code page plus font character set
X'A004' Orientation-support property ID
X'B001' Double-byte coded fonts supported
X'B002' Underscore width and position parameters in the
        LFI command are used by the printer
X'B003' GRID-parts fields allowed in the LFC, LFCSC, and
        LCPC commands
X'C01E' Coded-font pattern-technology ID, CID-keyed
        outline-font technology
X'C01F' Coded-font pattern-technology ID, Type 1 PFB
        outline-font technology
X'C101' Coded-font metric-technology ID, Relative metrics

DEVICE-CONTROL COMMAND-SET VECTOR
X'006EC4C3FF1060016002600361016201700017008702E7034
  706B707B707E70CE8008800A800C80F280F480F69001900290
  039004900590079009900A900D900E90139015901690F390F5
  E000E001E002E003F200F201F202F203F204F205F206F401F6
  01F602F804F902FB00FF01'
X'006E' 110     Length of the device-control command-set
                vector
X'C4C3' Device-control command-set ID
X'FF10' DC1 Subset ID
X'6001' Multiple copy & copy-subgroup support in LCC
X'6002' Media-source-selection support in LCC
X'6003' Media-destination-selection support in LCC
X'6101' Explicit page placement and orientation support in
        the LPP command


Date - 12/15/2019   Time - 12:41:2                    Page - 2
```

Figure 107. Hardcopy Printer Information Report (Page 2 of 14)

```
Printer Information from PSFMVS.WTRES600.DEV6.PRTINFO(PRT619)

X'6201' Logical page and object area coloring support
X'7001' Manage IPDS Dialog (MID) command support
X'7008' Set Presentation Environment (SPE) command support
X'702E' Activate Resource command support
X'7034' Presentation Fidelity Control Command Support
X'706B' Invoke CMR (ICMR) command support
X'707B' Rasterize Presentation Object (RPO) command
        support
X'707E' Include Saved Pages (ISP) command support
X'70CE' DUA command-support property ID
X'8008' Mark Form
X'800A' Alternate Offset Stacker
X'800C' Control Edge Marks
X'80F2' Discard Buffered Data
X'80F4' Request Resource List
X'80F6' Exception-Handling Control
X'9001' Print Buffered Data
X'9002' Deactivate Save Page Group
X'9003' Specify Group Operation
X'9004' Define Group Boundary
X'9005' Erase Residual Print Data
X'9007' Erase Residual Font Data
X'9009' Separate Continuous Forms
X'900A' Remove Saved Page Group
X'900D' Stack Received Pages
X'900E' Select Medium Modifications
X'9013' Eject to Front Facing
X'9015' Select Input Media Source
X'9016' Set Media Origin
X'90F3' Obtain Printer Characteristics
X'90F5' Page Counters Control
X'E000' CMRs can be captured
X'E001' Host-activated link color conversion CMRs
        supported
X'E002' Host-activated, non-generic halftone CMRs
        supported
X'E003' Host-activated, non-generic tone transfer curve
        CMRs supported
X'F200' Local Date and Time Stamp triplets supported in AR
        commands
X'F201' Activation-failed NACK support
X'F202' Font resolution and Metric Technology triplets
        supported in AR commands
X'F203' Metric Adjustment triplets supported in AR
        commands
X'F204' Data-object font support
X'F205' Color Management triplet support in the IDO, RPO,
        LPD, SPE, WBCC, WIC2, WGC, and WOCC commands
X'F206' Device Appearance (X'97') triplet support
X'F401' XOA RRL Multiple Entry Query Support. The printer
        supports multiple-entry queries of query type
        X'05', activation query.
X'F601' Position-Check Highlighting Support in XOA EHC
X'F602' Independent Exception Page-Print in XOA EHC
X'F804' Simplex and duplex N-up supported in the LCC
        command


Date - 12/15/2017   Time - 12:41:2                                    Page - 3
```

Figure 108. Hardcopy Printer Information Report (Page 3 of 14)

```
    Printer Information from PSFMVS.WTRES600.DEV6.PRTINFO(PRT619)

X'F902' Basic cut-sheet emulation mode supported
X'FB00' All architected units of measure supported
X'FF01' Positioning Exception Sense Format Supported

IM-IMAGE COMMAND-SET VECTOR
X'000CC9D4FF1010014022A004'
X'000C' 12      Length of the IM-image command-set vector
X'C9D4' IM1 subset of the IM-image command-set
X'FF10' IMD1 data
X'1001' IM-image objects may be sent in any order
X'4022' Standard OCA color-support property ID
X'A004' Orientation-Support property ID

IO-IMAGE COMMAND-SET VECTOR
X'0026C9D6FF10100112024022440150015003500850805081
   50825101520455505A004F300F301'
X'0026' 38      Length of the IO-image command-set vector
X'C9D6' IO1 subset of the IO-image command-set
X'FF10' IOCA FS10 data
X'1001' IO-image objects may be sent in any order
X'1202' IO-image objects can be downloaded in home state
        as resources
X'4022' Standard OCA color-support property ID
X'4401' Extended IOCA bi-level color support; when this
        property pair is present, the printer supports Set
        Extended Bi-level Image Color (X'F4')
        self-defining field on the WIC2-IDD
X'5001' Modified ITU-TSS Modified READ Algorithm (IBM MMR)
X'5003' Uncompressed image
X'5008' ABIC (bilevel Q-coded) Compression Algorithm
        (ABIC)
X'5080' ITU-TSS T.4 Facsimile Coding Scheme (G3 MH, one
        dimensional
X'5081' ITU-TSS T.4 Facsimile Coding Scheme (G3 MR, two
        dimensional)
X'5082' ITU-TSS T.6 Facsimile Coding Scheme (G4 MMR)
X'5101' Bit ordering supported in the IOCA Image Encoding
        Parameter
X'5204' Unpadded RIDIC recording algorithm supported
X'5505' Multiple Image content support
X'A004' Object area orientation-support property ID
X'F300' Replicate-and-trim mapping supported
X'F301' Scale-to-fill mapping supported

IO-IMAGE COMMAND-SET VECTOR
X'0024C9D6FF11100112024022440150015003500850005A5082
   50835101520455505A004F301'
X'0024' 36      Length of the IO-image command-set vector
X'C9D6' IO1 subset of the IO-image command-set
X'FF11' IOCA FS11 data, implies FS10 is also supported
X'1001' IO-image objects may be sent in any order
X'1202' IO-image objects can be downloaded in home state
        as resources
X'4022' Standard OCA color-support property ID
X'4401' Extended IOCA bi-level color support; when this
        property pair is present, the printer supports Set


Date - 12/15/2019   Time - 12:41:2                              Page - 4
```

Figure 109. Hardcopy Printer Information Report (Page 4 of 14)

```
Printer Information from PSFMVS.WTRES600.DEV6.PRTINFO(PRT619)

         Extended Bi-level Image Color (X'F4')
         self-defining field on the WIC2-IDD
X'5001'  Modified ITU-TSS Modified READ Algorithm (IBM MMR)
X'5003'  Uncompressed image
X'5008'  ABIC (bilevel Q-coded) Compression Algorithm
         (ABIC)
X'500A'  Concatenated ABIC
X'5082'  ITU-TSS T.6 Facsimile Coding Scheme (G4 MMR)
X'5083'  ISO/ITU-TSS JPEG algorithms
X'5101'  Bit ordering supported in the IOCA Image Encoding
         Parameter
X'5204'  Unpadded RIDIC recording algorithm supported
X'5505'  Multiple Image content support
X'A004'  Object area orientation-support property ID
X'F301'  Scale-to-fill mapping supported

IO-IMAGE COMMAND-SET VECTOR
X'0024C9D6FF4010011202402044015001500350085080508 1
   5082510152045505A004F301'
X'0024'  36      Length of the IO-image command-set vector
X'C9D6'  IO1 subset of the IO-image command-set
X'FF40'  IOCA FS40 data
X'1001'  IO-image objects may be sent in any order
X'1202'  IO-image objects can be downloaded in home state
         as resources
X'4020'  Standard OCA color-support property ID
X'4401'  Extended IOCA bi-level color support; when this
         property pair is present, the printer supports Set
         Extended Bi-level Image Color (X'F4')
         self-defining field on the WIC2-IDD
X'5001'  Modified ITU-TSS Modified READ Algorithm (IBM MMR)
X'5003'  Uncompressed image
X'5008'  ABIC (bilevel Q-coded) Compression Algorithm
         (ABIC)
X'5080'  ITU-TSS T.4 Facsimile Coding Scheme (G3 MH, one
         dimensional)
X'5081'  ITU-TSS T.4 Facsimile Coding Scheme (G3 MR, two
         dimensional)
X'5082'  ITU-TSS T.6 Facsimile Coding Scheme (G4 MMR)
X'5101'  Bit ordering supported in the IOCA Image Encoding
         Parameter
X'5204'  Unpadded RIDIC recording algorithm supported
X'5505'  Multiple Image content support
X'A004'  Object area orientation-support property ID
X'F301'  Scale-to-fill mapping supported

IO-IMAGE COMMAND-SET VECTOR
X'0022C9D6FF421001120240204401500150035008502 05082
   510152045505A004F301'
X'0022'  34      Length of the IO-image command-set vector
X'C9D6'  IO1 subset of the IO-image command-set
X'FF42'  IOCA FS42 data
X'1001'  IO-image objects may be sent in any order
X'1202'  IO-image objects can be downloaded in home state
         as resources
X'4020'  Standard OCA color-support property ID


Date - 12/15/2017   Time - 12:41:2                              Page - 5
```

Figure 110. Hardcopy Printer Information Report (Page 5 of 14)

```
       Printer Information from PSFMVS.WTRES600.DEV6.PRTINFO(PRT619)

X'4401' Extended IOCA bi-level color support; when this
        property pair is present, the printer supports Set
        Extended Bi-level Image Color (X'F4')
        self-defining field on the WIC2-IDD
X'5001' Modified ITU-TSS Modified READ Algorithm (IBM MMR)
X'5003' Uncompressed image
X'5008' ABIC (bilevel Q-coded) Compression Algorithm
        (ABIC)
X'5020' Solid Fill Rectangle
X'5082' ITU-TSS T.6 Facsimile Coding Scheme (G4 MMR)
X'5101' Bit ordering supported in the IOCA Image Encoding
        Parameter
X'5204' Unpadded RIDIC recording algorithm supported
X'5505' Multiple Image content support
X'A004' Object area orientation-support property ID
X'F301' Scale-to-fill mapping supported

IO-IMAGE COMMAND-SET VECTOR
X'0018C9D6FF45100112024020440150205101505A004F301'
X'0018' 24      Length of the IO-image command-set vector
X'C9D6' IO1 subset of the IO-image command-set
X'FF45' IOCA FS45 data, implies FS42 is also supported
X'1001' IO-image objects may be sent in any order
X'1202' IO-image objects can be downloaded in home state
        as resources
X'4020' Standard OCA color-support property ID
X'4401' Extended IOCA bi-level color support; when this
        property pair is present, the printer supports Set
        Extended Bi-level Image Color (X'F4')
        self-defining field on the WIC2-IDD
X'5020' Solid Fill Rectangle
X'5101' Bit ordering supported in the IOCA Image Encoding
        Parameter
X'5505' Multiple Image content support
X'A004' Object area orientation-support property ID
X'F301' Scale-to-fill mapping supported

OBJECT CONTAINER COMMAND-SET VECTOR
X'000CD6C3000012015800F301'
X'000C' 12      Length of the object container command-set
                vector
X'D6C3' OC1 subset of the object container command-set
X'0000' No levels defined
X'1201' Data-object-resource support
X'5800' Image Resolution (X'9A') triplet supported in IDO,
        RPO, and WOCC commands
X'F301' Scale-to-fill mapping supported

OVERLAY COMMAND-SET VECTOR
X'000CD6D3FF1011021505A004'
X'000C' 12      Length of the overlay command-set vector
X'D6D3' Overlay command-set ID
X'FF10' OL1 subset ID
X'1102' Extended overlay support, up to 32,511 overlays
        can be activated at one time
X'1505' x'15nn'- Overlay nesting up to nn levels is


Date - 12/15/2019   Time - 12:41:2                          Page - 6
```

Figure 111. Hardcopy Printer Information Report (Page 6 of 14)

Printer Information from PSFMVS.WTRES600.DEV6.PRTINFO(PRT619)

```
        supported
X'A004' Page-overlay-rotation support, all 4 orientations
        supported in the IO command
```

PAGE SEGMENT COMMAND-SET VECTOR
```
X'0008D7E2FF101101'
X'0008' 8       Length of the page segment command-set
                vector
X'D7E2' Page segment command-set ID
X'FF10' PS1 subset ID
X'1101' Extended page segment support, up to 32,511 page
        segments can be activated at one time.
```

TEXT COMMAND-SET VECTOR
```
X'000CD7E3FF301001402250FF'
X'000C' 12      Length of the text command-set vector
X'D7E3' TX1 subset of the text command-set
X'FF30' PTOCA PT3 data
X'1001' Unordered text supported
X'4022' Standard OCA color-support property ID
X'50FF' Multiple text-orientation support for all
        supported media origins
```

GRAPHICS COMMAND-SET VECTOR
```
X'0014E5C7FF2010014022410041014102410 6A004'
X'0014' 20      Length of the graphics command-set vector
X'E5C7' GR1 subset of the graphics command-set
X'FF20' GOCA DR/2V0 data
X'1001' Graphics objects may be sent in any order
X'4022' Standard OCA color-support property ID
X'4100' Set Process Color drawing order support
X'4101' Box drawing orders supported
X'4102' Partial Arc drawing orders supported
X'4106' Set Fractional Line Width drawing order supported
X'A004' Object area orientation-support property ID
```

```
Date - 12/15/2017   Time - 12:41:2                                    Page - 7
```

Figure 112. Hardcopy Printer Information Report (Page 7 of 14)

```
Printer Information from PSFMVS.WTRES600.DEV6.PRTINFO(PRT619)

OPC Data

00180001 00000000 38405FA0 3DE00000
00003DE0 5FA0B000 000A0003 00FF1770
1770000A 00100001 00010001 004B0004
1301010E E6B28000 00000000 11001200
1300141B 01020EE6 B2800000 00000021
00220023 00240031 00320033 00341901
03008000 00000000 00004000 42004800
4A005000 60007000 08000601 000B0000
08000701 000B0000 34000A01 00010302
00020303 00040005 00060006 03070007
03080009 00100010 03110312 00120320
08400040 09410042 00420900 20000B01
03010603 03030606 03070308 03080609
06100310 07400941 0A420900 18000E0D
11181A1B 1C1D1E1F 20212286 87919296
97989A00 06001201 03005800 13310001
F0F0F4F1 F0F0D4C4 F1C9C2D4 F1F0F0F0
F0F0F0F0 F0F0F0F0 F0F00000 F0F0F0F0
F0F0F0F0 F0F1F54B F0F44BF2 F1F823O0
03C99586 96D79989 95A340F4 F1F0F000
00000000 00000000 00000000 00000000
0000D800 14420106 072B1200 0401010E
00000000 00000006 072B1200 0401013C
00000000 00000006 072B1200 04010116
00000000 00000006 072B1200 04010117
00000000 00000092 0206072B 12000401
01140000 00000000 0006072B 12000401
012F0000 00000000 0006072B 12000401
01330000 00000000 0006072B 12000401
01350000 00000000 0006072B 12000401
01390000 00000000 0006072B 12000401
010E0000 00000000 0006072B 12000401
013C0000 00000000 0006072B 12000401
01160000 00000000 0006072B 12000401
01170000 00000000 00000A00 15225051
5D5E5F00 06001675 96------ --------
```

Date - 12/15/2019   Time - 12:41:2   Page - 8

Figure 113. Hardcopy Printer Information Report (Page 8 of 14)

```
Printer Information from PSFMVS.WTRES600.DEV6.PRTINFO(PRT619)

PRINTABLE AREA SELF-DEFINING FIELD
X'001800010000000038405FA03DE0000000003DE05FA0B000'
X'0018' 24      Length of this SDF, including itself
X'0001' Printable Area Self-Defining Field
X'00'   Media-source ID
X'00'   Reserved
X'00'   Unit base equals 10 inches
X'00'   Reserved
X'3840' 14,400  Units per unit base value for this
                self-defining field
X'5FA0' 24,480  Actual width of the medium presentation
                space in L-units
X'3DE0' 15,840  Actual length of the medium presentation
                space in L-units
X'0000' 0       Xm offset of the physical printable area in
                L-units
X'0000' 0       Ym offset of the physical printable area in
                L-units
X'3DE0' 15,840  Xm extent of the physical printable area in
                L-units
X'5FA0' 24,480  Ym extent of the physical printable area in
                L-units
X'B000' B'1011000000000000' Input Media Source
                            Characteristics
1... .... Duplex
.01. .... Continuous forms
...1 .... Media source available
.... 0... Retired item 119
.... .0.. Not envelope media
.... ..0. Automatic media feed
.... ...0 Not computer output microfilm media
0... .... Continuous forms media with carrier strips
.0.. .... Not an inserter bin
..00 0000 Reserved

IM-IMAGE AND CODED-FONT RESOLUTION SELF-DEFINING FIELD
X'000A000300FF17701770'
X'000A' 10      Length of this SDF, including itself
X'0003' IM-Image and Coded-Font Resolution Self-Defining
        Field
X'00'   Ten-inch increments
X'FF'   All resolutions in the range X'0001' - X'7FFF'
X'1770' 6,000   X pels per unit base
X'1770' 6,000   Y pels per unit base

MEDIA-DESTINATIONS SELF-DEFINING FIELD
X'000A0010000100010001'
X'000A' 10      Length of this SDF, including itself
X'0010' Media-Destinations Self-Defining Field
X'0001' Default media-destination ID
X'0001' First number in a range of available, contiguous
        media-destination IDs
X'0001' Last number in a range of available, contiguous
        media destination IDs

STORAGE POOLS SELF-DEFINING FIELD


Date - 12/15/2019   Time - 12:41:2                                      Page - 9
```

Figure 114. Hardcopy Printer Information Report (Page 9 of 14)

```
Printer Information from PSFMVS.WTRES600.DEV6.PRTINFO(PRT619)

X'004B000041301010EE6B280000000000000110012001300141B
  01020EE6B28000000000000210022002300240031003200330
  3419010300080000000000000004000420048004A0050006000
  70'
X'004B' 75     Length of this SDF, including itself
X'0004' Storage Pools Self-Defining Field
X'13'   19     Length of the entry, including itself
X'01'   Entry ID
X'01'   Storage pool ID
X'0EE6B280'  250,000,000  Size of the storage pool, in
                          bytes, when empty
X'00000000'  Reserved
X'0011' Page graphics data
X'0012' Page image data
X'0013' Page text data
X'0014' Page bar code data
X'1B'   27     Length of the entry, including itself
X'01'   Entry ID
X'02'   Storage pool ID
X'0EE6B280'  250,000,000  Size of the storage pool, in
                          bytes, when empty
X'00000000'  Reserved
X'0021' Overlay graphics data
X'0022' Overlay image data
X'0023' Overlay text data
X'0024' Overlay bar code data
X'0031' Page segment graphics data
X'0032' Page segment image data
X'0033' Page segment text data
X'0034' Page segment bar code data
X'19'   25     Length of the entry, including itself
X'01'   Entry ID
X'03'   Storage pool ID
X'00800000' 8,388,608    Size of the storage pool, in
                          bytes, when empty
X'00000000' Reserved
X'0040' Single-byte coded-font index tables
X'0042' Single-byte coded-font patterns
X'0048' Double-byte coded-font index tables
X'004A' Double-byte coded-font patterns
X'0050' Code pages
X'0060' Font character sets
X'0070' Coded fonts

INSTALLED FEATURES SELF-DEFINING FIELD
X'0008000601000B00'
X'0008' 8      Length of this SDF, including itself
X'0006' Installed Features Self-Defining Field
X'0100' Duplex
X'0B00' Continuous-Forms Output

AVAILABLE FEATURES SELF-DEFINING FIELD
X'0008000701000B00'
X'0008' 8      Length of this SDF, including itself
X'0007' Available Features Self-Defining Field
X'0100' Duplex available from at least one media source


Date - 12/15/2019   Time - 12:41:2                              Page - 10
```

Figure 115. Hardcopy Printer Information Report (Page 10 of 14)

```
    Printer Information from PSFMVS.WTRES600.DEV6.PRTINFO(PRT619)

X'0B00' Continuous-Forms Output

XOA RRL RT & RIDF SUPPORT SELF-DEFINING FIELD
X'0034000A010000103020002030300040005000600060307000
    07030800009001000100311031200120320008400004009410042
    004209'
X'0034' 52      Length of this SDF, including itself
X'000A' XOA RRL RT & RIDF Support Self-Defining Field
X'01'   Single-byte LF1-type and LF2-type coded font
X'00'   Host-Assigned Resource ID
X'01'   Single-byte LF1-type and LF2-type coded font
X'03'   GRID-parts format
X'02'   Double-byte LF1-type coded font
X'00'   Host-Assigned Resource ID
X'02'   Double-byte LF1-type coded font
X'03'   GRID-parts format
X'03'   Double-byte LF1-type coded-font section
X'00'   Host-Assigned Resource ID
X'04'   Page segment
X'00'   Host-Assigned Resource ID
X'05'   Overlay
X'00'   Host-Assigned Resource ID
X'06'   Device-version code page
X'00'   Host-Assigned Resource ID
X'06'   Device-version code page
X'03'   GRID-parts format
X'07'   Font character set
X'00'   Host-Assigned Resource ID
X'07'   Font character set
X'03'   GRID-parts format
X'08'   Single-byte coded font index
X'00'   Host-Assigned Resource ID
X'09'   Double-byte coded font section index
X'00'   Host-Assigned Resource ID
X'10'   Coded font
X'00'   Host-Assigned Resource ID
X'10'   Coded font
X'03'   GRID-parts format
X'11'   Graphic character set supported in a font
        character set
X'03'   GRID-parts format
X'12'   Specific code page
X'00'   Host-Assigned Resource ID
X'12'   Specific code page
X'03'   GRID-parts format
X'20'   Saved page group
X'08'   Variable-length group ID triplet
X'40'   Data object resource
X'00'   Host-Assigned Resource ID
X'40'   Data object resource
X'09'   Object-OID format
X'41'   Data-object font
X'00'   Host-Assigned Resource ID
X'42'   Data-object-font components
X'00'   Host-Assigned Resource ID
X'42'   Data-object-font components


Date - 12/15/2019   Time - 12:41:2                                    Page - 11
```

Figure 116. Hardcopy Printer Information Report (Page 11 of 14)

```
Printer Information from PSFMVS.WTRES600.DEV6.PRTINFO(PRT619)

X'09'   Object-OID format

ACTIVATE RESOURCE RT & RIDF SUPPORT SELF-DEFINING FIELD
X'0020000B0103010603030306060307030803080609061003
  10074009410A4209'
X'0020' 32      Length of this SDF, including itself
X'000B' Activate Resource RT & RIDF Support Self-Defining
        Field
X'01'   Single-byte LF1-type and LF2-type coded font
X'03'   GRID-parts format
X'01'   Single-byte LF1-type and LF2-type coded font
X'06'   MVS Host Unalterable Remote Font Environment
X'03'   Double-byte LF1-type coded-font section
X'03'   GRID-parts format
X'03'   Double-byte LF1-type coded-font section
X'06'   MVS Host Unalterable Remote Font Environment
X'06'   Code page
X'03'   GRID-parts format
X'07'   Font character set
X'03'   GRID-parts format
X'08'   Single-byte LF1-type coded-font index
X'03'   GRID-parts format
X'08'   Single-byte LF1-type coded-font index
X'06'   MVS Host Unalterable Remote Font Environment
X'09'   Double-byte LF1-type coded-font section index
X'06'   MVS Host Unalterable Remote Font Environment
X'10'   Coded font
X'03'   GRID-parts format
X'10'   Coded font
X'07'   Coded-font format
X'40'   Data object resource
X'09'   Object-OID format
X'41'   Data-object font
X'0A'   Data-object-font format
X'42'   Data-object-font components
X'09'   Object-OID format

COMMON BAR CODE TYPE/MODIFIER SELF-DEFINING FIELD
X'0018000E0D11181A1B1C1D1E1F2021228687919296979B9A'
X'0018' 24      Length of this SDF, including itself
X'000E' Common Bar Code Type/Modifier Self-Defining Field
X'0D'   Codabar - modifier-byte options X'01' and X'02'
X'11'   Code 128 - modifier-byte option X'02'
X'18'   POSTNET - modifier-byte options X'00' through
        X'03'
X'1A'   RM4SCC - modifier-byte option X'00'
X'1B'   Japan Postal Bar Code - modifier-byte options
        X'00' and X'01'
X'1C'   Data Matrix. Modifier Byte option X'00'.
X'1D'   MaxiCode. Modifier Byte option X'00'.
X'1E'   PDF417 - modifier Byte options X'00' and X'01'
X'1F'   Australia Post Bar Code - modifier-byte options
        X'01'-X'08'
X'20'   QR Code, modifier-byte option X'02'
X'21'   Code 93, modifier-byte option X'00'
X'22'   USPS Four-State, modifier-byte options X'00'


Date - 12/15/2019   Time - 12:41:2                                    Page - 12
```

Figure 117. Hardcopy Printer Information Report (Page 12 of 14)

```
        through X'03'
X'86'   UPC-Two-digit Supplemental - modifier-byte options
        X'01'and X'02'
X'87'   UPC-Five-digit Supplemental - modifier-byte
        options X'01' and X'02'
X'91'   Code 128, modifier-byte option X'03'
X'92'   Code 128, modifier-byte option X'04'
X'96'   EAN Two-digit Supplemental - modifer-byte option
        X'01'
X'97'   EAN Five-digit Supplemental - modifier-byte option
        X'01'
X'98'   POSTNET, modifier-byte option X'04'
X'9A'   RM4SCC, modifier-byte option X'01'

SUPPORTED GROUP OPERATIONS SELF-DEFINING FIELD
X'000600120103'
X'0006' 6       Length of this SDF, including itself
X'0012' Supported Group Operations Self-Defining Field
X'01'   Keep group together as a print unit
X'03'   Save pages

PRODUCT IDENTIFIER SELF-DEFINING FIELD
X'00580013310001F0F0F4F1F0F0D4C4F1C9C2D4F1F0F0F0F0
  F0F0F0F0F0F0F0F0F0F0OOOOOF0F0F0F0F0F0F0F0F0F0F1F54BF0F4
  4BF2F1F8230003C9958696D7998995A340F4F1F0F000000000
  000000000000000000000000000000'
X'0058' 88      Length of this SDF, including itself
X'0013' Product Identifier Self-Defining Field
X'31'   49      Product-identifier parameter length,
                including itself
X'0001' Unique Product Identifier
004100          Device Type
MD1             Model Number
IBM             Manufacturer
10              Plant of manufacture
000000000000    Sequence number
X'0000' Tag
000000000       Engineering Change level
X'F1F54BF0F44BF2F1F8'
        Device-specific information
X'23'   35      Product-identifier parameter length,
                including itself
X'0003' Printer Name Identifier
InfoPrint 4100
        External name of the printer

OBJECT CONTAINER TYPE SUPPORT SELF-DEFINING FIELD
X'00D80014420106072B12000401010E000000000000000607
  2B12000401013C0000000000000000006072B1200040101160000
  000000000000006072B120004010117000000000000000092020607
  2B1200040101140000000000000000006072B12000401012F0000
  000000000006072B120004010133000000000000000006072B12
  0004010135000000000000000006072B12000401013900000000
  000000006072B12000401010E0000000000000000006072B120004
  01013C0000000000000000006072B120004010116000000000000
  0006072B1200040101170000000000000000'
X'0008' 216     Length of this SDF, including itself


Date - 12/15/2019   Time - 12:41:2                                    Page - 13
```

Figure 118. Hardcopy Printer Information Report (Page 13 of 14)

```
Printer Information from PSFMVS.WTRES600.DEV6.PRTINFO(PRT619)

X'0014'  Object Container Type Support Self-Defining Field
X'42'    66      Type Record Length
X'01'    Page or overlay state
X'06072B12000401010E00000000000000' Tag Image File Format
                                     (TIFF)
X'06072B12000401013C00000000000000' Tag Image File Format
                                     (TIFF) without
                                     Transparency
X'06072B12000401011600000000000000' Graphics-Interchange
                                     Format (GIF)
X'06072B12000401011700000000000000' AFPC JPEG Subset (JPEG)
X'92'    146     Type Record Length
X'02'    Home state
X'06072B12000401011400000000000000' Color Mapping Table
                                     Setup File
X'06072B12000401012F00000000000000' IOCA Tile Resource
X'06072B12000401013300000000000000' TrueType/OpenType Font
X'06072B12000401013500000000000000' TrueType/OpenType
                                     Collection
X'06072B12000401013900000000000000' Color Management
                                     Resource (CMR)
X'06072B12000401010E00000000000000' Tag Image File Format
                                     (TIFF)
X'06072B12000401013C00000000000000' Tage Image File Format
                                     (TIFF) without
                                     Transparency
X'06072B12000401011600000000000000' Graphics Interchange
                                     Format (GIF)
X'06072B12000401011700000000000000' AFPC JPEG Subset (JPEG)

DF DEACTIVATION TYPES SUPPORTED SELF-DEFINING FIELD
X'000A00152250515D5E5F'
X'000A' 10      Length of this SDF, including itself
X'0015' DF Deactivation Types Supported Self-Defining
        Field
X'22'   Deactivate a font index for a double-byte coded
        font section
X'50'   Deactivate a coded font
X'51'   Deactivate a coded font and all associated
        components
X'5D'   Deactivate all resident coded fonts and all
        associated components
X'5E'   Deactivate all coded fonts
X'5F'   Deactivate all coded fonts and all associated
        components

PFC TRIPLETS SUPPORTED SELF-DEFINING FIELD
X'000600167596'
X'0006' 6       Length of this SDF, including itself
X'0016' PFC Triplets Supported Self-Defining Field
X'75'   Color Fidelity triplet
X'96'   CMR Tag Fidelity triplet


Date - 12/15/2019   Time - 12:41:2                          Page - 14
```

Figure 119. Hardcopy Printer Information Report (Page 14 of 14)

Softcopy report

– show an example of the softcopy report from an InfoPrint 4100 printer. You can view this softcopy report or format and print it as a hardcopy report. You can also display this same report in the system log.

```
REPORTLVL   PRTINFO  4.7.0   0000 PSF Printer Information Report
HEADING     Printer Information from SAMPROC1.PRTINFO(PRT619)
DATETIME    12/15/2019 11:11:20.57
COMMENT     ********************************************************
TITLE       STM Data
COMMENT     ********************************************************
STMDUMP     FF4100C0 0000000E C2C3FF10 10014022
STMDUMP     4400A004 0014C3C6 FF10A004 B001B002
STMDUMP     B003C005 C100C101 0014C3C6 FF30A004
STMDUMP     B001B002 B003C01E C01FC101 006EC4C3
STMDUMP     FF106001 60026003 61016201 70017008
STMDUMP     702E7034 706B707B 707E70CE 8008800A
STMDUMP     800C80F2 80F480F6 90019002 90039004
STMDUMP     90059007 9009900A 900D900E 90139015
STMDUMP     901690F3 90F5E000 E001E002 E003F200
STMDUMP     F201F202 F203F204 F205F206 F401F601
STMDUMP     F602F804 F902FB00 FF01000C C9D4FF10
STMDUMP     10014022 A0040026 C9D6FF10 10011202
STMDUMP     40224401 50015003 50085080 50815082
STMDUMP     51015204 5505A004 F300F301 0024C9D6
STMDUMP     FF111001 12024022 44015001 50035008
STMDUMP     500A5082 50835101 52045505 A004F301
STMDUMP     0024C9D6 FF401001 12024020 44015001
STMDUMP     50035008 50805081 50825101 52045505
STMDUMP     A004F301 0022C9D6 FF421001 12024020
STMDUMP     44015001 50035008 50205082 51015204
STMDUMP     5505A004 F3010018 C9D6FF45 10011202
STMDUMP     40204401 50205101 5505A004 F301000C
STMDUMP     D6C30000 12015800 F301000C D6D3FF10
STMDUMP     11021505 A0040008 D7E2FF10 1101000C
STMDUMP     D7E3FF30 10014022 50FF0014 E5C7FF20
STMDUMP     10014022 41004101 41024106 A004----
STMBEGIN    X'FF'    "System/370 convention"
STMTYPE     X'4100'  "Device type of the printer, or of the printer that"
STMTCONT             "is being emulated or mimicked"
STMMODEL    X'C0'    "Model Number"
STMRSRV     X'0000'  "Reserved"
STMTITLE
COMMENT     ********************************************************
STMTITLE    "BAR CODE COMMAND-SET VECTOR"
COMMENT     ********************************************************
STMVCTR     X'000EC2C3FF10100140224400A004'
STMLENGTH   X'000E' 14      "Length of the bar code command-set vector"
STMSUBSET   X'C2C3' "BC1 subset of the bar code command-set"
STMLEVEL    X'FF10' "BCOCA BCD1 data"
STMPPAIR    X'1001' "Bar code objects may be sent in any order"
STMPPAIR    X'4022' "Standard OCA color-support property ID"
STMPPAIR    X'4400' "Extended bar code color support; when this"
STMPCONT            "property pair is present, the printer supports the"
STMPCONT            "Color Specification (X'4E') triplet on WBCC-BCDD"
STMPPAIR    X'A004' "Object area orientation-support property ID"
STMTITLE
```

Figure 120. Softcopy Printer Information Report (Page 1 of 13)

```
COMMENT      ********************************************************
STMTITLE     "LOADED-FONT COMMAND-SET VECTOR"
COMMENT      ********************************************************
STMVCTR      X'0014C3C6FF10A004B001B002B003C005C100C101'
STMLENGTH    X'0014' 20      "Length of the loaded-font command-set"
STMLENCONT                   "vector"
STMCMDSET    X'C3C6' "Loaded-font command-set ID"
STMSUBSET    X'FF10' "LF1 subset ID - fully described font plus font"
STMSUBCONT           "index"
STMPPAIR     X'A004' "Orientation-support property ID"
STMPPAIR     X'B001' "Double-byte coded fonts supported"
STMPPAIR     X'B002' "Underscore width and position parameters in the"
STMPCONT             "LFI command are used by the printer"
STMPPAIR     X'B003' "GRID-parts fields allowed in the LFC, LFCSC, and"
STMPCONT             "LCPC commands"
STMPPAIR     X'C005' "Coded-font pattern-technology ID, Bounded-box"
STMPCONT             "raster-font technology"
STMPPAIR     X'C100' "Coded-font metric-technology ID, Fixed metrics"
STMPPAIR     X'C101' "Coded-font metric-technology ID, Relative metrics"
STMTITLE
COMMENT      ********************************************************
STMTITLE     "LOADED-FONT COMMAND-SET VECTOR"
COMMENT      ********************************************************
STMVCTR      X'0014C3C6FF30A004B001B002B003C01EC01FC101'
STMLENGTH    X'0014' 20      "Length of the loaded-font command-set"
STMLENCONT                   "vector"
STMCMDSET    X'C3C6' "Loaded-font command-set ID"
STMSUBSET    X'FF30' "LF3 subset ID - code page plus font character set"
STMPPAIR     X'A004' "Orientation-support property ID"
STMPPAIR     X'B001' "Double-byte coded fonts supported"
STMPPAIR     X'B002' "Underscore width and position parameters in the"
STMPCONT             "LFI command are used by the printer"
STMPPAIR     X'B003' "GRID-parts fields allowed in the LFC, LFCSC, and"
STMPCONT             "LCPC commands"
STMPPAIR     X'C01E' "Coded-font pattern-technology ID, CID-keyed"
STMPCONT             "outline-font technology"
STMPPAIR     X'C01F' "Coded-font pattern-technology ID, Type 1 PFB"
STMPCONT             "outline-font technology"
STMPPAIR     X'C101' "Coded-font metric-technology ID, Relative metrics"
STMTITLE
COMMENT      ********************************************************
STMTITLE     "DEVICE-CONTROL COMMAND-SET VECTOR"
COMMENT      ********************************************************
STMVCTR      X'006EC4C3FF10600160026003610162017001700870870870TE7034
STMVCONT     706B707B707E70CE8008800A800C80F280F480F69001900290
STMVCONT     0390049005900790099009A9000D900E90139015901690F390F5
STMVCONT     E000E001E002E003F200F201F202F203F204F205F206F401F6
STMVCONT     01F602F804F902FB00FF01'
STMLENGTH    X'006E' 110     "Length of the device-control command-set"
STMLENCONT                   "vector"
STMCMDSET    X'C4C3' "Device-control command-set ID"
STMSUBSET    X'FF10' "DC1 Subset ID"
STMPPAIR     X'6001' "Multiple copy & copy-subgroup support in LCC"
STMPPAIR     X'6002' "Media-source-selection support in LCC"
STMPPAIR     X'6003' "Media-destination-selection support in LCC"
STMPPAIR     X'6101' "Explicit page placement and orientation support in"
STMPCONT             "the LPP command"
STMPPAIR     X'6201' "Logical page and object area coloring support"
STMPPAIR     X'7001' "Manage IPDS Dialog (MID) command support"
STMPPAIR     X'7008' "Set Presentation Environment (SPE) command support"
STMPPAIR     X'702E' "Activate Resource command support"
STMPPAIR     X'7034' "Presentation Fidelity Control Command Support"
STMPPAIR     X'706B' "Invoke CMR (ICMR) command support"
```

Figure 121. Softcopy Printer Information Report (Page 2 of 13)

```
STMPPAIR   X'707B'  "Rasterize Presentation Object (RPO) command"
STMPCONT            "support"
STMPPAIR   X'707E'  "Include Saved Pages (ISP) command support"
STMPPAIR   X'70CE'  "DUA command-support property ID"
STMPPAIR   X'8008'  "Mark Form"
STMPPAIR   X'800A'  "Alternate Offset Stacker"
STMPPAIR   X'800C'  "Control Edge Marks"
STMPPAIR   X'80F2'  "Discard Buffered Data"
STMPPAIR   X'80F4'  "Request Resource List"
STMPPAIR   X'80F6'  "Exception-Handling Control"
STMPPAIR   X'9001'  "Print Buffered Data"
STMPPAIR   X'9002'  "Deactivate Save Page Group"
STMPPAIR   X'9003'  "Specify Group Operation"
STMPPAIR   X'9004'  "Define Group Boundary"
STMPPAIR   X'9005'  "Erase Residual Print Data"
STMPPAIR   X'9007'  "Erase Residual Font Data"
STMPPAIR   X'9009'  "Separate Continuous Forms"
STMPPAIR   X'900A'  "Remove Saved Page Group"
STMPPAIR   X'900D'  "Stack Received Pages"
STMPPAIR   X'900E'  "Select Medium Modifications"
STMPPAIR   X'9013'  "Eject to Front Facing"
STMPPAIR   X'9015'  "Select Input Media Source"
STMPPAIR   X'9016'  "Set Media Origin"
STMPPAIR   X'90F3'  "Obtain Printer Characteristics"
STMPPAIR   X'90F5'  "Page Counters Control"
STMPPAIR   X'E000'  "CMRs can be captured"
STMPPAIR   X'E001'  "Host-activated link color conversion CMRs"
STMPCONT            "supported"
STMPPAIR   X'E002'  "Host-activated, non-generic halftone CMRs"
STMPCONT            "supported"
STMPPAIR   X'E003'  "Host-activated, non-generic tone transfer curve"
STMPCONT            "CMRs supported"
STMPPAIR   X'F200'  "Local Date and Time Stamp triplets supported in AR"
STMPCONT            "commands"
STMPPAIR   X'F201'  "Activation-failed NACK support"
STMPPAIR   X'F202'  "Font Resolution and Metric Technology triplets"
STMPCONT            "supported in AR commands"
STMPPAIR   X'F203'  "Metric Adjustment triplets supported in AR"
STMPCONT            "commands"
STMPPAIR   X'F204'  "Data-object font support"
STMPPAIR   X'F205'  "Color Management triplet support in the IDO, RPO,"
STMPCONT            "LPD, SPE, WBCC, WIC2, WGC, and WOCC commands"
STMPPAIR   X'F206'  "Device Appearance (X'97') triplet support"
STMPPAIR   X'F401'  "XOA RRL Multiple Entry Query Support. The printer"
STMPCONT            "supports multiple-entry queries of query type"
STMPCONT            "X'05', activation query."
STMPPAIR   X'F601'  "Position-Check Highlighting Support in XOA EHC"
STMPPAIR   X'F602'  "Independent Exception Page-Print in XOA EHC"
STMPPAIR   X'F804'  "Simplex and duplex N-up supported in the LCC"
STMPCONT            "command"
STMPPAIR   X'F902'  "Basic cut-sheet emulation mode supported"
STMPPAIR   X'FB00'  "All architected units of measure supported"
STMPPAIR   X'FF01'  "Positioning Exception Sense Format Supported"
STMTITLE
COMMENT    ********************************************************
STMTITLE   "IM-IMAGE COMMAND-SET VECTOR"
COMMENT    ********************************************************
STMVCTR    X'000CC9D4FF1010014022A004'
STMLENGTH  X'000C' 12      "Length of the IM-image command-set vector"
STMSUBSET  X'C9D4' "IM1 subset of the IM-image command-set"
STMLEVEL   X'FF10' "IMD1 data"
STMPPAIR   X'1001' "IM-image objects may be sent in any order"
```

Figure 122. Softcopy Printer Information Report (Page 3 of 13)

```
STMPPAIR   X'4022' "Standard OCA color-support property ID"
STMPPAIR   X'A004' "Orientation-Support property ID"
STMTITLE
COMMENT    ********************************************************
STMTITLE   "IO-IMAGE COMMAND-SET VECTOR"
COMMENT    ********************************************************
STMVCTR    X'0026C9D6FF1010011202402244015001500350085080805081
STMVCONT   5082510152045505A004F300F301'
STMLENGTH  X'0026' 38      "Length of the IO-image command-set vector"
STMSUBSET  X'C9D6' "IO1 subset of the IO-image command-set"
STMLEVEL   X'FF10' "IOCA FS10 data"
STMPPAIR   X'1001' "IO-image objects may be sent in any order"
STMPPAIR   X'1202' "IO-image objects can be downloaded in home state"
STMPCONT           "as resources"
STMPPAIR   X'4022' "Standard OCA color-support property ID"
STMPPAIR   X'4401' "Extended IOCA bi-level color support; when this"
STMPCONT           "property pair is present, the printer supports Set"
STMPCONT           "Extended Bi-level Image Color (X'F4')"
STMPCONT           "self-defining field on the WIC2-IDD"
STMPPAIR   X'5001' "Modified ITU-TSS Modified READ Algorithm (IBM MMR)"
STMPPAIR   X'5003' "Uncompressed image"
STMPPAIR   X'5008' "ABIC (bilevel Q-coded) Compression Algorithm"
STMPCONT           "(ABIC)"
STMPPAIR   X'5080' "ITU-TSS T.4 Facsimile Coding Scheme (G3 MH, one"
STMPCONT           "dimensional)"
STMPPAIR   X'5081' "ITU-TSS T.4 Facsimile Coding Scheme (G3 MR, two"
STMPCONT           "dimensional)"
STMPPAIR   X'5082' "ITU-TSS T.6 Facsimile Coding Scheme (G4 MMR)"
STMPPAIR   X'5101' "Bit ordering supported in the IOCA Image Encoding"
STMPCONT           "Parameter"
STMPPAIR   X'5204' "Unpadded RIDIC recording algorithm supported"
STMPPAIR   X'5505' "Multiple Image content support"
STMPPAIR   X'A004' "Object area orientation-support property ID"
STMPPAIR   X'F300' "Replicate-and-trim mapping supported"
STMPPAIR   X'F301' "Scale-to-fill mapping supported"
STMTITLE
COMMENT    ********************************************************
STMTITLE   "IO-IMAGE COMMAND-SET VECTOR"
COMMENT    ********************************************************
STMVCTR    X'0024C9D6FF1110011202402244015001500350085000A5082
STMVCONT   5083510152045505A004F301'
STMLENGTH  X'0024' 36      "Length of the IO-image command-set vector"
STMSUBSET  X'C9D6' "IO1 subset of the IO-image command-set"
STMLEVEL   X'FF11' "IOCA FS11 data, implies FS10 is also supported"
STMPPAIR   X'1001' "IO-image objects may be sent in any order"
STMPPAIR   X'1202' "IO-image objects can be downloaded in home state"
STMPCONT           "as resources"
STMPPAIR   X'4022' "Standard OCA color-support property ID"
STMPPAIR   X'4401' "Extended IOCA bi-level color support; when this"
STMPCONT           "property pair is present, the printer supports Set"
STMPCONT           "Extended Bi-level Image Color (X'F4')"
STMPCONT           "self-defining field on the WIC2-IDD"
STMPPAIR   X'5001' "Modified ITU-TSS Modified READ Algorithm (IBM MMR)"
STMPPAIR   X'5003' "Uncompressed image"
STMPPAIR   X'5008' "ABIC (bilevel Q-coded) Compression Algorithm"
STMPCONT           "(ABIC)"
STMPPAIR   X'500A' "Concatenated ABIC"
STMPPAIR   X'5082' "ITU-TSS T.6 Facsimile Coding Scheme (G4 MMR)"
STMPPAIR   X'5083' "ISO/ITU-TSS JPEG algorithms"
STMPPAIR   X'5101' "Bit ordering supported in the IOCA Image Encoding"
STMPCONT           "Parameter"
STMPPAIR   X'5204' "Unpadded RIDIC recording algorithm supported"
```

Figure 123. Softcopy Printer Information Report (Page 4 of 13)

```
STMPPAIR   X'5505' "Multiple Image content support"
STMPPAIR   X'A004' "Object area orientation-support property ID"
STMPPAIR   X'F301' "Scale-to-fill mapping supported"
STMTITLE
COMMENT    ********************************************************
STMTITLE   "IO-IMAGE COMMAND-SET VECTOR"
COMMENT    ********************************************************
STMVCTR    X'0024C9D6FF401001120240204401500150035008508050 81
STMVCONT   5082510152045505A004F301'
STMLENGTH  X'0024' 36     "Length of the IO-image command-set vector"
STMSUBSET  X'C9D6' "IO1 subset of the IO-image command-set"
STMLEVEL   X'FF40' "IOCA FS40 data"
STMPPAIR   X'1001' "IO-image objects may be sent in any order"
STMPPAIR   X'1202' "IO-image objects can be downloaded in home state"
STMPCONT          "as resources"
STMPPAIR   X'4020' "Standard OCA color-support property ID"
STMPPAIR   X'4401' "Extended IOCA bi-level color support; when this"
STMPCONT          "property pair is present, the printer supports Set"
STMPCONT          "Extended Bi-level Image Color (X'F4')"
STMPCONT          "self-defining field on the WIC2-IDD"
STMPPAIR   X'5001' "Modified ITU-TSS Modified READ Algorithm (IBM MMR)"
STMPPAIR   X'5003' "Uncompressed image"
STMPPAIR   X'5008' "ABIC (bilevel Q-coded) Compression Algorithm"
STMPCONT          "(ABIC)"
STMPPAIR   X'5080' "ITU-TSS T.4 Facsimile Coding Scheme (G3 MH, one"
STMPCONT          "dimensional)"
STMPPAIR   X'5081' "ITU-TSS T.4 Facsimile Coding Scheme (G3 MR, two"
STMPCONT          "dimensional)"
STMPPAIR   X'5082' "ITU-TSS T.6 Facsimile Coding Scheme (G4 MMR)"
STMPPAIR   X'5101' "Bit ordering supported in the IOCA Image Encoding"
STMPCONT          "Parameter"
STMPPAIR   X'5204' "Unpadded RIDIC recording algorithm supported"
STMPPAIR   X'5505' "Multiple Image content support"
STMPPAIR   X'A004' "Object area orientation-support property ID"
STMPPAIR   X'F301' "Scale-to-fill mapping supported"
STMTITLE
COMMENT    ********************************************************
STMTITLE   "IO-IMAGE COMMAND-SET VECTOR"
COMMENT    ********************************************************
STMVCTR    X'0022C9D6FF4210011202402044015001500350085020 5082
STMVCONT   510152045505A004F301'
STMLENGTH  X'0022' 34     "Length of the IO-image command-set vector"
STMSUBSET  X'C9D6' "IO1 subset of the IO-image command-set"
STMLEVEL   X'FF42' "IOCA FS42 data"
STMPPAIR   X'1001' "IO-image objects may be sent in any order"
STMPPAIR   X'1202' "IO-image objects can be downloaded in home state"
STMPCONT          "as resources"
STMPPAIR   X'4020' "Standard OCA color-support property ID"
STMPPAIR   X'4401' "Extended IOCA bi-level color support; when this"
STMPCONT          "property pair is present, the printer supports Set"
STMPCONT          "Extended Bi-level Image Color (X'F4')"
STMPCONT          "self-defining field on the WIC2-IDD"
STMPPAIR   X'5001' "Modified ITU-TSS Modified READ Algorithm (IBM MMR)"
STMPPAIR   X'5003' "Uncompressed image"
STMPPAIR   X'5008' "ABIC (bilevel Q-coded) Compression Algorithm"
STMPCONT          "(ABIC)"
STMPPAIR   X'5020' "Solid Fill Rectangle"
STMPPAIR   X'5082' "ITU-TSS T.6 Facsimile Coding Scheme (G4 MMR)"
STMPPAIR   X'5101' "Bit ordering supported in the IOCA Image Encoding"
STMPCONT          "Parameter"
STMPPAIR   X'5204' "Unpadded RIDIC recording algorithm supported"
STMPPAIR   X'5505' "Multiple Image content support"
```

Figure 124. Softcopy Printer Information Report (Page 5 of 13)

```
STMPPAIR   X'A004' "Object area orientation-support property ID"
STMPPAIR   X'F301' "Scale-to-fill mapping supported"
STMTITLE
COMMENT    *******************************************************
STMTITLE   "IO-IMAGE COMMAND-SET VECTOR"
COMMENT    *******************************************************
STMVCTR    X'0018C9D6FF4510011202402044015020510155050A004F301'
STMLENGTH  X'0018' 24     "Length of the IO-image command-set vector"
STMSUBSET  X'C9D6' "IO1 subset of the IO-image command-set"
STMLEVEL   X'FF45' "IOCA FS45 data, implies FS42 is also supported"
STMPPAIR   X'1001' "IO-image objects may be sent in any order"
STMPPAIR   X'1202' "IO-image objects can be downloaded in home state"
STMPCONT          "as resources"
STMPPAIR   X'4020' "Standard OCA color-support property ID"
STMPPAIR   X'4401' "Extended IOCA bi-level color support; when this"
STMPCONT          "property pair is present, the printer supports Set"
STMPCONT          "Extended Bi-level Image Color (X'F4')"
STMPCONT          "self-defining field on the WIC2-IDD"
STMPPAIR   X'5020' "Solid Fill Rectangle"
STMPPAIR   X'5101' "Bit ordering supported in the IOCA Image Encoding"
STMPCONT          "Parameter"
STMPPAIR   X'5505' "Multiple Image content support"
STMPPAIR   X'A004' "Object area orientation-support property ID"
STMPPAIR   X'F301' "Scale-to-fill mapping supported"
STMTITLE
COMMENT    *******************************************************
STMTITLE   "OBJECT CONTAINER COMMAND-SET VECTOR"
COMMENT    *******************************************************
STMVCTR    X'000CD6C3000012015800F301'
STMLENGTH  X'000C' 12     "Length of the object container command-set"
STMLENCONT               "vector"
STMSUBSET  X'D6C3' "OC1 subset of the object container command-set"
STMLEVEL   X'0000' "No levels defined"
STMPPAIR   X'1201' "Data-object-resource support"
STMPPAIR   X'5800' "Image Resolution (X'9A') triplet supported in IDO,"
STMPCONT          "RPO, and WOCC commands"
STMPPAIR   X'F301' "Scale-to-fill mapping supported"
STMTITLE
COMMENT    *******************************************************
STMTITLE   "OVERLAY COMMAND-SET VECTOR"
COMMENT    *******************************************************
STMVCTR    X'000CD6D3FF1011021505A004'
STMLENGTH  X'000C' 12     "Length of the overlay command-set vector"
STMCMDSET  X'D6D3' "Overlay command-set ID"
STMSUBSET  X'FF10' "OL1 subset ID"
STMPPAIR   X'1102' "Extended overlay support, up to 32,511 overlays"
STMPCONT          "can be activated at one time"
STMPPAIR   X'1505' "x'15nn' - Overlay nesting up to nn levels is"
STMPCONT          "supported"
STMPPAIR   X'A004' "Page-overlay-rotation support, all 4 orientations"
STMPCONT          "supported in the IO command"
STMTITLE
COMMENT    *******************************************************
STMTITLE   "PAGE SEGMENT COMMAND-SET VECTOR"
COMMENT    *******************************************************
STMVCTR    X'0008D7E2FF101101'
STMLENGTH  X'0008' 8      "Length of the page segment command-set"
STMLENCONT               "vector"
STMCMDSET  X'D7E2' "Page segment command-set ID"
STMSUBSET  X'FF10' "PS1 subset ID"
```

Figure 125. Softcopy Printer Information Report (Page 6 of 13)

```
STMPPAIR   X'1101' "Extended page segment support, up to 32,511 page"
STMPCONT           "segments can be activated at one time."
STMTITLE
COMMENT    ********************************************************
STMTITLE   "TEXT COMMAND-SET VECTOR"
COMMENT    ********************************************************
STMVCTR    X'000CD7E3FF301001402250FF'
STMLENGTH  X'000C' 12      "Length of the text command-set vector"
STMSUBSET  X'D7E3' "TX1 subset of the text command-set"
STMLEVEL   X'FF30' "PTOCA PT3 data"
STMPPAIR   X'1001' "Unordered text supported"
STMPPAIR   X'4022' "Standard OCA color-support property ID"
STMPPAIR   X'50FF' "Multiple text-orientation support for all"
STMPCONT           "supported media origins"
STMTITLE
COMMENT    ********************************************************
STMTITLE   "GRAPHICS COMMAND-SET VECTOR"
COMMENT    ********************************************************
STMVCTR    X'0014E5C7FF2010014022410041010141024106A004'
STMLENGTH  X'0014' 20      "Length of the graphics command-set vector"
STMSUBSET  X'E5C7' "GR1 subset of the graphics command-set"
STMLEVEL   X'FF20' "GOCA DR/2V0 data"
STMPPAIR   X'1001' "Graphics objects may be sent in any order"
STMPPAIR   X'4022' "Standard OCA color-support property ID"
STMPPAIR   X'4100' "Set Process Color drawing order support"
STMPPAIR   X'4101' "Box drawing orders supported"
STMPPAIR   X'4102' "Partial Arc drawing orders supported"
STMPPAIR   X'4106' "Set Fractional Line Width drawing order supported"
STMPPAIR   X'A004' "Object area orientation-support property ID"
STMEND
COMMENT    ********************************************************
TITLE      OPC Data
COMMENT    ********************************************************
OPCDUMP    00180001 00000000 38405FA0 3DE00000
OPCDUMP    00003DE0 5FA0B000 000A0003 00FF1770
OPCDUMP    1770000A 00100001 00010001 004B0004
OPCDUMP    1301010E E6B28000 00000000 11001200
OPCDUMP    1300141B 01020EE6 B2800000 00000021
OPCDUMP    00220023 00240031 00320033 00341901
OPCDUMP    03008000 00000000 00004000 42004800
OPCDUMP    4A005000 60007000 08000601 000B0000
OPCDUMP    08000701 000B0000 34000A01 00010302
OPCDUMP    00020303 00040005 00060006 03070007
OPCDUMP    03080009 00100010 03110312 00120320
OPCDUMP    08400040 09410042 00420900 20000B01
OPCDUMP    03010603 03030606 03070308 03080609
OPCDUMP    06100310 07400941 0A420900 18000E0D
OPCDUMP    11181A1B 1C1D1E1F 20212286 87919296
OPCDUMP    97989A00 06001201 03005800 13310001
OPCDUMP    F0F0F4F1 F0F0D4C4 F1C9C2D4 F1F0F0F0
OPCDUMP    F0F0F0F0 F0F0F0F0 F0F00000 F0F0F0F0
OPCDUMP    F0F0F0F0 F0F1F54B F0F44BF2 F1F82300
OPCDUMP    03C99586 96D79989 95A340F4 F1F0F000
OPCDUMP    00000000 00000000 00000000 00000000
OPCDUMP    0000D800 14420106 072B1200 0401010E
OPCDUMP    00000000 00000006 072B1200 0401013C
OPCDUMP    00000000 00000006 072B1200 04010116
OPCDUMP    00000000 00000006 072B1200 04010117
OPCDUMP    00000000 00000092 0206072B 12000401
OPCDUMP    01140000 00000000 0006072B 12000401
OPCDUMP    012F0000 00000000 0006072B 12000401
OPCDUMP    01330000 00000000 0006072B 12000401
OPCDUMP    01350000 00000000 0006072B 12000401
OPCDUMP    01390000 00000000 0006072B 12000401
```

Figure 126. Softcopy Printer Information Report (Page 7 of 13)

```
OPCDUMP    010E0000 00000000 0006072B 12000401
OPCDUMP    013C0000 00000000 0006072B 12000401
OPCDUMP    01160000 00000000 0006072B 12000401
OPCDUMP    01170000 00000000 00000A00 15225051
OPCDUMP    5D5E5F00 06001675 96------ --------
OPCTITLE
COMMENT     ********************************************************
OPCTITLE    "PRINTABLE AREA SELF-DEFINING FIELD"
COMMENT     ********************************************************
OPCSDF      X'001800010000000038405FA03DE0000000003DE05FA0B000'
OPCLENGTH   X'0018' 24       "Length of this SDF, including itself"
OPCSDFID    X'0001' "Printable Area Self-Defining Field"
OPCCODE     X'00'    "Media-source ID"
OPCCODE     X'00'    "Reserved"
OPCCODE     X'00'    "Unit base equals 10 inches"
OPCCODE     X'00'    "Reserved"
OPCVAL      X'3840' 14,400 "Units per unit base value for this"
OPCVALCONT           "self-defining field"
OPCVAL      X'5FA0' 24,480 "Actual width of the medium presentation"
OPCVALCONT           "space in L-units"
OPCVAL      X'3DE0' 15,840 "Actual length of the medium presentation"
OPCVALCONT           "space in L-units"
OPCVAL      X'0000' 0      "Xm offset of the physical printable area in"
OPCVALCONT           "L-units"
OPCVAL      X'0000' 0      "Ym offset of the physical printable area in"
OPCVALCONT           "L-units"
OPCVAL      X'3DE0' 15,840 "Xm extent of the physical printable area in"
OPCVALCONT           "L-units"
OPCVAL      X'5FA0' 24,480 "Ym extent of the physical printable area in"
OPCVALCONT           "L-units"
OPCBITS     X'B000' B'1011000000000000' "Input Media Source"
OPCBCONT                              "Characteristics"
OPCBIT      1... .... "Duplex"
OPCBIT      .01. .... "Continuous forms"
OPCBIT      ...1 .... "Media source available"
OPCBIT      .... 0... "Retired item 119"
OPCBIT      .... .0.. "Not envelope media"
OPCBIT      .... ..0. "Automatic media feed"
OPCBIT      .... ...0 "Not computer output microfilm media"
OPCBIT      0... .... "Continuous forms media with carrier strips"
OPCBIT      .0.. .... "Not an inserter bin"
OPCBIT      ..00 0000 "Reserved"
OPCTITLE
COMMENT     ********************************************************
OPCTITLE    "IM-IMAGE AND CODED-FONT RESOLUTION SELF-DEFINING FIELD"
COMMENT     ********************************************************
OPCSDF      X'000A000300FF17701770'
OPCLENGTH   X'000A' 10       "Length of this SDF, including itself"
OPCSDFID    X'0003' "IM-Image and Coded-Font Resolution Self-Defining"
OPCSIDCONT           "Field"
OPCCODE     X'00'    "Ten-inch increments"
OPCCODE     X'FF'    "All resolutions in the range X'0001' - X'7FFF'"
OPCVAL      X'1770' 6,000   "X pels per unit base"
OPCVAL      X'1770' 6,000   "Y pels per unit base"
OPCTITLE
COMMENT     ********************************************************
OPCTITLE    "MEDIA-DESTINATIONS SELF-DEFINING FIELD"
COMMENT     ********************************************************
OPCSDF      X'000A00100001000010001'
OPCLENGTH   X'000A' 10       "Length of this SDF, including itself"
OPCSDFID    X'0010' "Media-Destinations Self-Defining Field"
```

Figure 127. Softcopy Printer Information Report (Page 8 of 13)

```
OPCCODE    X'0001' "Default media-destination ID"
OPCCODE    X'0001' "First number in a range of available, contiguous"
OPCCDCONT          "media-destination IDs"
OPCCODE    X'0001' "Last number in a range of available, contiguous"
OPCCDCONT          "media-destination IDs"
OPCTITLE
COMMENT    **********************************************************
OPCTITLE   "STORAGE POOLS SELF-DEFINING FIELD"
COMMENT    **********************************************************
OPCSDF     X'004B00041301010EE6B280000000000001100120013000141B
OPCSDFCONT 01020EE6B2800000000000210022002300240031003200330D
OPCSDFCONT 341901030080000000000000004000420048004A0050006000
OPCSDFCONT 70'
OPCLENGTH  X'004B' 75     "Length of this SDF, including itself"
OPCSDFID   X'0004' "Storage Pools Self-Defining Field"
OPCLENGTH  X'13'   19     "Length of the entry, including itself"
OPCCODE    X'01'   "Entry ID"
OPCCODE    X'01'   "Storage pool ID"
OPCVAL4    X'0EE6B280' 250,000,000   "Size of the storage pool, in"
OPCV4CONT                            "bytes, when empty"
OPCDATA    X'00000000' "Reserved"
OPCCODE    X'0011' "Page graphics data"
OPCCODE    X'0012' "Page image data"
OPCCODE    X'0013' "Page text data"
OPCCODE    X'0014' "Page bar code data"
OPCLENGTH  X'1B'   27     "Length of the entry, including itself"
OPCCODE    X'01'   "Entry ID"
OPCCODE    X'02'   "Storage pool ID"
OPCVAL4    X'0EE6B280' 250,000,000   "Size of the storage pool, in"
OPCV4CONT                            "bytes, when empty"
OPCDATA    X'00000000' "Reserved"
OPCCODE    X'0021' "Overlay graphics data"
OPCCODE    X'0022' "Overlay image data"
OPCCODE    X'0023' "Overlay text data"
OPCCODE    X'0024' "Overlay bar code data"
OPCCODE    X'0031' "Page segment graphics data"
OPCCODE    X'0032' "Page segment image data"
OPCCODE    X'0033' "Page segment text data"
OPCCODE    X'0034' "Page segment bar code data"
OPCLENGTH  X'19'   25     "Length of the entry, including itself"
OPCCODE    X'01'   "Entry ID"
OPCCODE    X'03'   "Storage pool ID"
OPCVAL4    X'00800000' 8,388,608    "Size of the storage pool, in"
OPCV4CONT                            "bytes, when empty"
OPCDATA    X'00000000' "Reserved"
OPCCODE    X'0040' "Single-byte coded-font index tables"
OPCCODE    X'0042' "Single-byte coded-font patterns"
OPCCODE    X'0048' "Double-byte coded-font index tables"
OPCCODE    X'004A' "Double-byte coded-font patterns"
OPCCODE    X'0050' "Code pages"
OPCCODE    X'0060' "Font character sets"
OPCCODE    X'0070' "Coded fonts"
OPCTITLE
COMMENT    **********************************************************
OPCTITLE   "INSTALLED FEATURES SELF-DEFINING FIELD"
COMMENT    **********************************************************
OPCSDF     X'0008000601000B00'
OPCLENGTH  X'0008' 8       "Length of this SDF, including itself"
OPCSDFID   X'0006' "Installed Features Self-Defining Field"
OPCCODE    X'0100' "Duplex"
OPCCODE    X'0B00' "Continuous-Forms Output"
OPCTITLE
```

Figure 128. Softcopy Printer Information Report (Page 9 of 13)

```
COMMENT      ***********************************************************
OPCTITLE     "AVAILABLE FEATURES SELF-DEFINING FIELD"
COMMENT      ***********************************************************
OPCSDF       X'0008000701000B00'
OPCLENGTH    X'0008' 8        "Length of this SDF, including itself"
OPCSDFID     X'0007' "Available Features Self-Defining Field"
OPCCODE      X'0100' "Duplex available from at least one media source"
OPCCODE      X'0B00' "Continuous-Forms Output"
OPCTITLE
COMMENT      ***********************************************************
OPCTITLE     "XOA RRL RT & RIDF SUPPORT SELF-DEFINING FIELD"
COMMENT      ***********************************************************
OPCSDF       X'0034000A0100010302000203030004000500060006030700
OPCSDFCONT   0703080009001000100311031200120320084000400941 0042
OPCSDFCONT   004209'
OPCLENGTH    X'0034' 52       "Length of this SDF, including itself"
OPCSDFID     X'000A' "XOA RRL RT & RIDF Support Self-Defining Field"
OPCCODE      X'01'    "Single-byte LF1-type and LF2-type coded font"
OPCCODE      X'00'    "Host-Assigned Resource ID"
OPCCODE      X'01'    "Single-byte LF1-type and LF2-type coded font"
OPCCODE      X'03'    "GRID-parts format"
OPCCODE      X'02'    "Double-byte LF1-type coded font"
OPCCODE      X'00'    "Host-Assigned Resource ID"
OPCCODE      X'02'    "Double-byte LF1-type coded font"
OPCCODE      X'03'    "GRID-parts format"
OPCCODE      X'03'    "Double-byte LF1-type coded-font section"
OPCCODE      X'00'    "Host-Assigned Resource ID"
OPCCODE      X'04'    "Page segment"
OPCCODE      X'00'    "Host-Assigned Resource ID"
OPCCODE      X'05'    "Overlay"
OPCCODE      X'00'    "Host-Assigned Resource ID"
OPCCODE      X'06'    "Device-version code page"
OPCCODE      X'00'    "Host-Assigned Resource ID"
OPCCODE      X'06'    "Device-version code page"
OPCCODE      X'03'    "GRID-parts format"
OPCCODE      X'07'    "Font character set"
OPCCODE      X'00'    "Host-Assigned Resource ID"
OPCCODE      X'07'    "Font character set"
OPCCODE      X'03'    "GRID-parts format"
OPCCODE      X'08'    "Single-byte coded font index"
OPCCODE      X'00'    "Host-Assigned Resource ID"
OPCCODE      X'09'    "Double-byte coded font section index"
OPCCODE      X'00'    "Host-Assigned Resource ID"
OPCCODE      X'10'    "Coded font"
OPCCODE      X'00'    "Host-Assigned Resource ID"
OPCCODE      X'10'    "Coded font"
OPCCODE      X'03'    "GRID-parts format"
OPCCODE      X'11'    "Graphic character set supported in a font"
OPCCDCONT             "character set"
OPCCODE      X'03'    "GRID-parts format"
OPCCODE      X'12'    "Specific code page"
OPCCODE      X'00'    "Host-Assigned Resource ID"
OPCCODE      X'12'    "Specific code page"
OPCCODE      X'03'    "GRID-parts format"
OPCCODE      X'20'    "Saved page group"
OPCCODE      X'08'    "Variable-length group ID triplet"
OPCCODE      X'40'    "Data object resource"
OPCCODE      X'00'    "Host-Assigned Resource ID"
OPCCODE      X'40'    "Data object resource"
OPCCODE      X'09'    "Object-OID format"
OPCCODE      X'41'    "Data-object font"
OPCCODE      X'00'    "Host-Assigned Resource ID"
```

Figure 129. Softcopy Printer Information Report (Page 10 of 13)

```
OPCCODE      X'42'    "Data-object-font components"
OPCCODE      X'00'    "Host-Assigned Resource ID"
OPCCODE      X'42'    "Data-object-font components"
OPCCODE      X'09'    "Object-OID format"
OPCTITLE
COMMENT      *********************************************************
OPCTITLE     "ACTIVATE RESOURCE RT & RIDF SUPPORT SELF-DEFINING FIELD"
COMMENT      *********************************************************
OPCSDF       X'0020000B0103010603030306060307030803080609061003
OPCSDFCONT   10074009410A4209'
OPCLENGTH    X'0020' 32      "Length of this SDF, including itself"
OPCSDFID     X'000B' "Activate Resource RT & RIDF Support Self-Defining"
OPCSIDCONT           "Field"
OPCCODE      X'01'    "Single-byte LF1-type and LF2-type coded font"
OPCCODE      X'03'    "GRID-parts format"
OPCCODE      X'01'    "Single-byte LF1-type and LF2-type coded font"
OPCCODE      X'06'    "MVS Host Unalterable Remote Font Environment"
OPCCODE      X'03'    "Double-byte LF1-type coded-font section"
OPCCODE      X'03'    "GRID-parts format"
OPCCODE      X'03'    "Double-byte LF1-type coded-font section"
OPCCODE      X'06'    "MVS Host Unalterable Remote Font Environment"
OPCCODE      X'06'    "Code page"
OPCCODE      X'03'    "GRID-parts format"
OPCCODE      X'07'    "Font character set"
OPCCODE      X'03'    "GRID-parts format"
OPCCODE      X'08'    "Single-byte LF1-type coded-font index"
OPCCODE      X'03'    "GRID-parts format"
OPCCODE      X'08'    "Single-byte LF1-type coded-font index"
OPCCODE      X'06'    "MVS Host Unalterable Remote Font Environment"
OPCCODE      X'09'    "Double-byte LF1-type coded-font section index"
OPCCODE      X'06'    "MVS Host Unalterable Remote Font Environment"
OPCCODE      X'10'    "Coded font"
OPCCODE      X'03'    "GRID-parts format"
OPCCODE      X'10'    "Coded font"
OPCCODE      X'07'    "Coded-font format"
OPCCODE      X'40'    "Data object resource"
OPCCODE      X'09'    "Object-OID format"
OPCCODE      X'41'    "Data-object font"
OPCCODE      X'0A'    "Data-object-font format"
OPCCODE      X'42'    "Data-object-font components"
OPCCODE      X'09'    "Object-OID format"
OPCTITLE
COMMENT      *********************************************************
OPCTITLE     "COMMON BAR CODE TYPE/MODIFIER SELF-DEFINING FIELD"
COMMENT      *********************************************************
OPCSDF       X'0018000E0D11181A1B1C1D1E1F202122866879192966979 89A'
OPCLENGTH    X'0018' 24      "Length of this SDF, including itself"
OPCSDFID     X'000E' "Common Bar Code Type/Modifier Self-Defining Field"
OPCCODE      X'0D'    "Codabar - modifier-byte options X'01' and X'02'"
OPCCODE      X'11'    "Code 128 - modifier-byte option X'02'"
OPCCODE      X'18'    "POSTNET - modifier-byte options X'00' through"
OPCCDCONT            "X'03'"
OPCCODE      X'1A'    "RM4SCC - modifier-byte option X'00'"
OPCCODE      X'1B'    "Japan Postal Bar Code - modifier-byte options"
OPCCDCONT            "X'00' and X'01'"
OPCCODE      X'1C'    "Data Matrix. Modifier Byte option X'00'."
OPCCODE      X'1D'    "MaxiCode. Modifier Byte option X'00'."
OPCCODE      X'1E'    "PDF417 - modifier Byte options X'00' and X'01'"
OPCCODE      X'1F'    "Australia Post Bar Code - modifier-byte options"
OPCCDCONT            "X'01'-X'08'"
OPCCODE      X'20'    "QR Code, modifier-byte option X'02'"
OPCCODE      X'21'    "Code 93, modifier-byte option X'00'"
```

Figure 130. Softcopy Printer Information Report (Page 11 of 13)

```
OPCCODE    X'22'   "USPS Four-State, modifier-byte options X'00'"
OPCCDCONT          "through X'03'"
OPCCODE    X'86'   "UPC-Two-digit Supplemental - modifier-byte options"
OPCCDCONT          "X'01' and X'02'"
OPCCODE    X'87'   "UPC-Five-digit Supplemental - modifier-byte"
OPCCDCONT          "options X'01' and X'02'"
OPCCODE    X'91'   "Code 128, modifier-byte option X'03'"
OPCCODE    X'92'   "Code 128, modifier-byte option X'04'"
OPCCODE    X'96'   "EAN Two-digit Supplemental - modifer-byte option"
OPCCDCONT          "X'01'"
OPCCODE    X'97'   "EAN Five-digit Supplemental - modifier-byte option"
OPCCDCONT          "X'01'"
OPCCODE    X'98'   "POSTNET, modifier-byte option X'04'"
OPCCODE    X'9A'   "RM4SCC, modifier-byte option X'01'"
OPCTITLE
COMMENT    ********************************************************
OPCTITLE   "SUPPORTED GROUP OPERATIONS SELF-DEFINING FIELD"
COMMENT    ********************************************************
OPCSDF     X'000600120103'
OPCLENGTH  X'0006' 6       "Length of this SDF, including itself"
OPCSDFID   X'0012' "Supported Group Operations Self-Defining Field"
OPCCODE    X'01'   "Keep group together as a print unit"
OPCCODE    X'03'   "Save pages"
OPCTITLE
COMMENT    ********************************************************
OPCTITLE   "PRODUCT IDENTIFIER SELF-DEFINING FIELD"
COMMENT    ********************************************************
OPCSDF     X'00580013310001F0F0F4F1F0F0D4C4F1C9C2D4F1F0F0F0F0
OPCSDFCONT F0F0F0F0F0F0F0F0F0F00000F0F0F0F0F0F0F0F0F0F1F54BF0F4
OPCSDFCONT 4BF2F1F8230003C9958696D7998995A340F4F1F0F000000000
OPCSDFCONT 00000000000000000000000000000000'
OPCLENGTH  X'0058' 88      "Length of this SDF, including itself"
OPCSDFID   X'0013' "Product Identifier Self-Defining Field"
OPCLENGTH  X'31'   49      "Product-identifier parameter length,"
OPCLENCONT                 "including itself"
OPCCODE    X'0001' "Unique Product Identifier"
OPCCHAR    004100          "Device Type"
OPCCHAR    MD1             "Model Number"
OPCCHAR    IBM             "Manufacturer"
OPCCHAR    10              "Plant of manufacture"
OPCCHAR    000000000000 "Sequence number"
OPCCODE    X'0000' "Tag"
OPCCHAR    000000000     "Engineering Change level"
OPCSDF     X'F1F54BF0F44BF2F1F8'
OPCDATA            "Device-specific information"
OPCLENGTH  X'23'   35    "Product-identifier parameter length,"
OPCLENCONT                "including itself"
OPCCODE    X'0003' "Printer Name Identifier"
OPCSDF     InfoPrint 4100
OPCCHAR            "External name of the printer"
OPCTITLE
COMMENT    ********************************************************
OPCTITLE   "OBJECT CONTAINER TYPE SUPPORT SELF-DEFINING FIELD"
COMMENT    ********************************************************
OPCSDF     X'00D800144420106072B12000401010E000000000000000607
OPCSDFCONT 2B12000401013C0000000000000000006072B1200040101160000
OPCSDFCONT 000000000006072B120004010117000000000000000092020607
OPCSDFCONT 2B12000401010114000000000000000006072B12000401012F0000
OPCSDFCONT 00000000006072B120004010133000000000000000006072B12
OPCSDFCONT 000401013500000000000000006072B120004010139000000000
OPCSDFCONT 00000006072B12000401010E000000000000000006072B120004
OPCSDFCONT 01013C0000000000000000006072B12000401011600000000000000
```

Figure 131. Softcopy Printer Information Report (Page 12 of 13)

```
OPCSDFCONT 0006072B12000040101170000000000000000'
OPCLENGTH  X'00D8' 216    "Length of this SDF, including itself"
OPCSDFID   X'0014' "Object Container Type Support Self-Defining Field"
OPCLENGTH  X'42'   66     "Type Record Length"
OPCCODE    X'01'    "Page or overlay state"
OPCCODE16  X'06072B12000401010E00000000000000' "Tag Image File Format"
OPCC16CONT                                      "(TIFF)"
OPCCODE16  X'06072B12000401013C00000000000000' "Tag Image File Format"
OPCC16CONT                                      "(TIFF) without"
OPCC16CONT                                      "Transparency"
OPCCODE16  X'06072B12000040101160000000000000' "Graphics Interchange"
OPCC16CONT                                      "Format (GIF)"
OPCCODE16  X'06072B12000040101170000000000000' "AFPC JPEG Subset (JPEG)"
OPCLENGTH  X'92'   146    "Type Record Length"
OPCCODE    X'02'    "Home state"
OPCCODE16  X'06072B12000040101140000000000000' "Color Mapping Table"
OPCC16CONT                                      "Setup File"
OPCCODE16  X'06072B12000401012F00000000000000' "IOCA Tile Resource"
OPCCODE16  X'06072B12000040101330000000000000' "TrueType/OpenType Font"
OPCCODE16  X'06072B12000040101350000000000000' "TrueType/OpenType"
OPCC16CONT                                      "Collection"
OPCCODE16  X'06072B12000040101390000000000000' "Color Management"
OPCC16CONT                                      "Resource (CMR)"
OPCCODE16  X'06072B12000401010E00000000000000' "Tag Image File Format"
OPCC16CONT                                      "(TIFF)"
OPCCODE16  X'06072B12000401013C00000000000000' "Tag Image File Format"
OPCC16CONT                                      "(TIFF) without"
OPCC16CONT                                      "Transparency"
OPCCODE16  X'06072B12000040101160000000000000' "Graphics Interchange"
OPCC16CONT                                      "Format (GIF)"
OPCCODE16  X'06072B12000040101170000000000000' "AFPC JPEG Subset (JPEG)"
OPCTITLE
COMMENT    *******************************************************
OPCTITLE   "DF DEACTIVATION TYPES SUPPORTED SELF-DEFINING FIELD"
COMMENT    *******************************************************
OPCSDF     X'000A00152250515D5E5F'
OPCLENGTH  X'000A' 10     "Length of this SDF, including itself"
OPCSDFID   X'0015' "DF Deactivation Types Supported Self-Defining"
OPCSIDCONT         "Field"
OPCCODE    X'22'   "Deactivate a font index for a double-byte coded"
OPCCDCONT          "font section"
OPCCODE    X'50'   "Deactivate a coded font"
OPCCODE    X'51'   "Deactivate a coded font and all associated"
OPCCDCONT          "components"
OPCCODE    X'5D'   "Deactivate all resident coded fonts and all"
OPCCDCONT          "associated components"
OPCCODE    X'5E'   "Deactivate all coded fonts"
OPCCODE    X'5F'   "Deactivate all coded fonts and all associated"
OPCCDCONT          "components"
OPCTITLE
COMMENT    *******************************************************
OPCTITLE   "PFC TRIPLETS SUPPORTED SELF-DEFINING FIELD"
COMMENT    *******************************************************
OPCSDF     X'000600167596'
OPCLENGTH  X'0006' 6      "Length of this SDF, including itself"
OPCSDFID   X'0016' "PFC Triplets Supported Self-Defining Field"
OPCCODE    X'75'   "Color Fidelity triplet"
OPCCODE    X'96'   "CMR Tag Fidelity triplet"
OPCEND
```

Figure 132. Softcopy Printer Information Report (Page 13 of 13)

Softcopy record format

- give detailed information about the format of the softcopy printer information report in - . It lists each record that can be in the report, along with a detailed description. You can use this information to format your own hardcopy printer information report.

```
================================================================================
This is the generic mapping used by all records in the report.

   OFFSET    OFFSET
  DECIMAL     HEX   TYPE       LENGTH                 DESCRIPTION
  ======== ======== ========= ========               ==============================
         0      (0) CHARACTER       10                The Record Label (ID)
        10      (A) CHARACTER        1                Column Separator
        11      (B) CHARACTER      501                The variable part of the rec.


================================================================================
The HEADING record contains text with the name of the PRTINFO member.

   OFFSET    OFFSET
  DECIMAL     HEX   TYPE       LENGTH                 DESCRIPTION
  ======== ======== ========= ========               ==============================
         0      (0) CHARACTER       10                Layout ID Value: HEADING
        10      (A) CHARACTER        1                Column Separator
        11      (B) CHARACTER      501                Text with "Printer Information
                                                      from " followed by PDS
                                                      name(member name)


================================================================================
The REPORT LEVEL record contains the information relevant to uniquely identify the
report's format.

   OFFSET    OFFSET
  DECIMAL     HEX   TYPE       LENGTH                 DESCRIPTION
  ======== ======== ========= ========               ==============================
         0      (0) CHARACTER       10                Layout ID value: REPORTLVL
        10      (A) CHARACTER        1                Column Separator
        11      (B) CHARACTER        8                Report Identifier
        19     (13) CHARACTER        1                Column Separator
        20     (14) CHARACTER        8                PSF version, release, and
                                                      modification
        28     (1C) CHARACTER        1                Column Separator
        29     (1D) CHARACTER        4                Report Level Identifier
        33     (21) CHARACTER        1                Column Separator
        34     (22) CHARACTER       30                Report Description
        64     (40) CHARACTER        1                Column Separator
        65     (41) CHARACTER        8                FSA name
        73     (49) CHARACTER        1                Column Separator
        74     (4A) CHARACTER       12                CPU ID
        86     (56) CHARACTER        1                Column separator
        87     (57) CHARACTER        8                System name
        95     (5F) CHARACTER        1                Column separator
        96     (60) CHARACTER      250                IP address or LU name
```

Figure 133. Record formats for the Softcopy Printer Information Report (Page 1 of 12)

```
================================================================================
The DATETIME record has the date mm/dd/yyyy and the time hh:mm:ss.tu.

  OFFSET    OFFSET
 DECIMAL     HEX   TYPE        LENGTH                   DESCRIPTION
======== ======== ========= ========         ===============================
       0      (0) CHARACTER       10                   Layout ID Value: DATETIME
      10      (A) CHARACTER        1                   Column Separator
      11      (B) CHARACTER       10                   Date mm/dd/yyyy
      21     (15) CHARACTER        1                   Column separator
      22     (16) CHARACTER       11                   Time hh:mm:ss.tu

================================================================================
The COMMENT record contains asterisks to make the softcopy report more readable.
Comment records are not printed in the hardcopy report.

  OFFSET    OFFSET
 DECIMAL     HEX   TYPE        LENGTH                   DESCRIPTION
======== ======== ========= ========         ===============================
       0      (0) CHARACTER       10                   Layout ID Value: COMMENT
      10      (A) CHARACTER        1                   Column Separator
      11      (B) CHARACTER      501                   One line of *****

 ================================================================================
The TITLE record contains text that will be printed as a section heading.

  OFFSET    OFFSET
 DECIMAL     HEX   TYPE        LENGTH                   DESCRIPTION
======== ======== ========= ========         ===============================
       0      (0) CHARACTER       10                   Layout ID Value: TITLE
      10      (A) CHARACTER        1                   Column Separator
      11      (B) CHARACTER      501                   One line of text

================================================================================
The STMDUMP record contains 16 bytes of hex data returned in the IPDS STM response.

  OFFSET    OFFSET
 DECIMAL     HEX   TYPE        LENGTH                   DESCRIPTION
======== ======== ========= ========         ===============================
       0      (0) CHARACTER       10                   Layout ID Value: STMDUMP
      10      (A) CHARACTER        1                   Column Separator
      11      (B) STRUCTURE        8                   4 bytes of hex data
      19     (13) CHARACTER        1                   Column Separator
      20     (14) STRUCTURE        8                   4 bytes of hex data
      28     (1C) CHARACTER        1                   Column Separator
      29     (1D) STRUCTURE        8                   4 bytes of hex data
      37     (25) CHARACTER        1                   Column Separator
      38     (26) STRUCTURE        8                   4 bytes of hex data

================================================================================
The STMBEGIN record contains the first byte of the IPDS STM response.

  OFFSET    OFFSET
 DECIMAL     HEX   TYPE        LENGTH                   DESCRIPTION
======== ======== ========= ========         ===============================
       0      (0) CHARACTER       10                   Layout ID Value: STMBEGIN
      10      (A) CHARACTER        1                   Column Separator
      11      (B) CHARACTER        5                   X'FF'
      16     (10) CHARACTER        3                   Column Separator
      19     (13) CHARACTER        *                   The English description
                                                       delimited by quotes ("s)
```

Figure 134. Record formats for the Softcopy Printer Information Report (Page 2 of 12)

```
========================================================================================
The STMTYPE record contains the Device type of the IPDS STM response.

   OFFSET    OFFSET
  DECIMAL     HEX   TYPE        LENGTH                    DESCRIPTION
  ======== ======== ========= ========                   ===============================
        0     (0)   CHARACTER      10                    Layout ID Value: STMTYPE
       10     (A)   CHARACTER       1                    Column Separator
       11     (B)   CHARACTER       7                    Device type X'nnnn'
       18    (12)   CHARACTER       1                    Column Separator
       19    (13)   CHARACTER       *                    The English description
                                                         delimited by quotes ("s)


========================================================================================
The STMTCONT record contains the continuation of English description of STMTYPE.

   OFFSET    OFFSET
  DECIMAL     HEX   TYPE        LENGTH                    DESCRIPTION
  ======== ======== ========= ========                   ===============================
        0     (0)   CHARACTER      10                    Layout ID Value: STMTCONT
       10     (A)   CHARACTER       9                    Column Separator
       19    (13)   CHARACTER       *                    The English description
                                                         delimited by quotes ("s)


========================================================================================
The STMMODEL record contains the model number in the IPDS STM response.

   OFFSET    OFFSET
  DECIMAL     HEX   TYPE        LENGTH                    DESCRIPTION
  ======== ======== ========= ========                   ===============================
        0     (0)   CHARACTER      10                    Layout ID Value: STMMODEL
       10     (A)   CHARACTER       1                    Column Separator
       11     (B)   CHARACTER       5                    Model number X'nn'
       16    (10)   CHARACTER       3                    Column Separator
       19    (13)   CHARACTER       *                    The English description
                                                         delimited by quotes ("s)


========================================================================================
The STMMCONT record contains the continuation of the description for STMMODEL.

   OFFSET    OFFSET
  DECIMAL     HEX   TYPE        LENGTH                    DESCRIPTION
  ======== ======== ========= ========                   ===============================
        0     (0)   CHARACTER      10                    Layout ID Value: STMMCONT
       10     (A)   CHARACTER       9                    Column Separator
       19    (13)   CHARACTER       *                    The English description
                                                         delimited by quotes ("s)


========================================================================================
The STMRSRV record contains 2 bytes of hex data labeled Reserved in the IPDS STM
response.

   OFFSET    OFFSET
  DECIMAL     HEX   TYPE        LENGTH                    DESCRIPTION
  ======== ======== ========= ========                   ===============================
        0     (0)   CHARACTER      10                    Layout ID Value: STMRSRV
       10     (A)   CHARACTER       1                    Column Separator
       11     (B)   CHARACTER       7                    Hex data X'nnnn'
       18    (12)   CHARACTER       1                    Column Separator
       19    (13)   CHARACTER       *                    The English description
                                                         delimited by quotes ("s)
```

Figure 135. Record formats for the Softcopy Printer Information Report (Page 3 of 12)

```
================================================================================
The STMTITLE record contains blanks or text for an STM vector title.

  OFFSET    OFFSET
 DECIMAL     HEX   TYPE       LENGTH                DESCRIPTION
======== ======== ========= ========               ==============================
       0     (0)  CHARACTER      10                 Layout ID Value: STMTITLE
      10     (A)  CHARACTER       1                 Column Separator
      11     (B)  CHARACTER     501                 Blanks or vector title
                                                    delimited by quotes("s)

================================================================================
The STMTLCONT record contains the continuation of the description for STMTITLE.

  OFFSET    OFFSET
 DECIMAL     HEX   TYPE       LENGTH                DESCRIPTION
======== ======== ========= ========               ==============================
       0     (0)  CHARACTER      10                 Layout ID Value: STMTLCONT
      10     (A)  CHARACTER       1                 Column Separator
      11     (B)  CHARACTER     501                 Continuation of vector title
                                                    delimited by quotes("s)

================================================================================
The STMVCTR record contains the hex data for one STM vector.

  OFFSET    OFFSET
 DECIMAL     HEX   TYPE       LENGTH                DESCRIPTION
======== ======== ========= ========               ==============================
       0     (0)  CHARACTER      10                 Layout ID Value: STMVCTR
      10     (A)  CHARACTER       1                 Column Separator
      11     (B)  CHARACTER     501                 Hex vector data

================================================================================
The STMVCONT record contains the continuation of the hex data for STMVCTR.

  OFFSET    OFFSET
 DECIMAL     HEX   TYPE       LENGTH                DESCRIPTION
======== ======== ========= ========               ==============================
       0     (0)  CHARACTER      10                 Layout ID Value: STMVCONT
      10     (A)  CHARACTER       1                 Column Separator
      11     (B)  CHARACTER     501                 Hex vector data

================================================================================
The STMCONT record contains the continuation of the description for all STM records
that don't have a defined continuation record.

  OFFSET    OFFSET
 DECIMAL     HEX   TYPE       LENGTH                DESCRIPTION
======== ======== ========= ========               ==============================
       0     (0)  CHARACTER      10                 Layout ID Value: STMCONT
      10     (A)  CHARACTER       1                 Column Separator
      11     (B)  CHARACTER     501                 The English description
                                                    delimited by quotes ("s)

================================================================================
The STMLENGTH record contains the length of the STM vector in hex and in decimal.

  OFFSET    OFFSET
 DECIMAL     HEX   TYPE       LENGTH                DESCRIPTION
======== ======== ========= ========               ==============================
       0     (0)  CHARACTER      10                 Layout ID Value: STMLENGTH
      10     (A)  CHARACTER       1                 Column Separator
```

Figure 136. Record formats for the Softcopy Printer Information Report (Page 4 of 12)

```
        11         (B) CHARACTER        7              Length in hex
        18        (12) CHARACTER        1              Column Separator
        19        (13) CHARACTER        6              Length in decimal
        25        (19) CHARACTER        1              Column Separator
        26        (1A) CHARACTER        *              The English description
                                                       delimited by quotes ("s)


   =================================================================================
   The STMLENCONT record contains the continuation of the description of STMLENGTH.

    OFFSET    OFFSET
   DECIMAL      HEX  TYPE       LENGTH                  DESCRIPTION
   ========  ======== ========= ========               ================================
         0         (0) CHARACTER       10              Layout ID Value: STMLENCONT
        10         (A) CHARACTER       16              Column Separator
        26        (1A) CHARACTER        *              The English description
                                                       delimited by quotes ("s)


   =================================================================================
   The STMSUBSET record contains the 2-byte subset ID in the IPDS STM response.

    OFFSET    OFFSET
   DECIMAL      HEX  TYPE       LENGTH                  DESCRIPTION
   ========  ======== ========= ========               ================================
         0         (0) CHARACTER       10              Layout ID Value: STMSUBSET
        10         (A) CHARACTER        1              Column Separator
        11         (B) CHARACTER        7              Subset ID X'nnnn'
        18        (12) CHARACTER        1              Column Separator
        19        (13) CHARACTER        *              The English description
                                                       delimited by quotes ("s)


   =================================================================================
   The STMSUBCONT record contains the continuation of the description for STMSUBSET.

    OFFSET    OFFSET
   DECIMAL      HEX  TYPE       LENGTH                  DESCRIPTION
   ========  ======== ========= ========               ================================
         0         (0) CHARACTER       10              Layout ID Value: STMSUBCONT
        10         (A) CHARACTER        9              Column Separator
        19        (13) CHARACTER        *              The English description
                                                       delimited by quotes ("s)


   =================================================================================
   The STMCMDSET record contains the 2-byte command set ID in the IPDS STM response.

    OFFSET    OFFSET
   DECIMAL      HEX  TYPE       LENGTH                  DESCRIPTION
   ========  ======== ========= ========               ================================
         0         (0) CHARACTER       10              Layout ID Value: STMCMDSET
        10         (A) CHARACTER        1              Column Separator
        11         (B) CHARACTER        7              Command Set ID X'nnnn'
        18        (12) CHARACTER        1              Column Separator
        19        (13) CHARACTER        *              The English description
                                                       delimited by quotes ("s)


   =================================================================================
   The STMCMDCONT record contains the continuation of the description for STMCMDSET.

    OFFSET    OFFSET
   DECIMAL      HEX  TYPE       LENGTH                  DESCRIPTION
   ========  ======== ========= ========               ================================
         0         (0) CHARACTER       10              Layout ID Value: STMCMDCONT
        10         (A) CHARACTER        9              Column Separator
        19        (13) CHARACTER        *              The English description
                                                       delimited by quotes ("s)
```

Figure 137. Record formats for the Softcopy Printer Information Report (Page 5 of 12)

```
================================================================================
The STMLEVEL record contains the 2-byte level ID in the IPDS STM response.

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE      LENGTH                  DESCRIPTION
 ======= ======== ========= ========               ==============================
       0      (0) CHARACTER      10                 Layout ID Value: STMLEVEL
      10      (A) CHARACTER       1                 Column Separator
      11      (B) CHARACTER       7                 Level ID X'nnnn'
      18     (12) CHARACTER       1                 Column Separator
      19     (13) CHARACTER       *                 The English description
                                                    delimited by quotes ("s)


================================================================================
The STMLEVCONT record contains the continuation of the description for STMLEVEL.

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE      LENGTH                  DESCRIPTION
 ======= ======== ========= ========               ==============================
       0      (0) CHARACTER      10                 Layout ID Value: STMLEVCONT
      10      (A) CHARACTER       9                 Column Separator
      19     (13) CHARACTER       *                 The English description
                                                    delimited by quotes ("s)


================================================================================
The STMPPAIR record contains the 2-byte property pair ID in the IPDS STM response.

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE      LENGTH                  DESCRIPTION
 ======= ======== ========= ========               ==============================
       0      (0) CHARACTER      10                 Layout ID Value: STMPPAIR
      10      (A) CHARACTER       1                 Column Separator
      11      (B) CHARACTER       7                 Property Pair X'nnnn'
      18     (12) CHARACTER       1                 Column Separator
      19     (13) CHARACTER       *                 The English description
                                                    delimited by quotes ("s)


================================================================================
The STMPCONT record contains the continuation of the description for STMPPAIR.

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE      LENGTH                  DESCRIPTION
 ======= ======== ========= ========               ==============================
       0      (0) CHARACTER      10                 Layout ID Value: STMPCONT
      10      (A) CHARACTER       9                 Column Separator
      19     (13) CHARACTER       *                 The English description
                                                    delimited by quotes ("s)


================================================================================
The STMEND record indicates the end of the STM data.

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE      LENGTH                  DESCRIPTION
 ======= ======== ========= ========               ==============================
       0      (0) CHARACTER      10                 Layout ID Value: STMEND
      10      (A) CHARACTER       1                 Column Separator
      11      (B) CHARACTER     501                 Blanks
```

Figure 138. Record formats for the Softcopy Printer Information Report (Page 6 of 12)

```
=================================================================================
The OPCDUMP record contains 16 bytes of hex data returned in the IPDS XOH OPC.
response.

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE       LENGTH                    DESCRIPTION
 ======== ======== ========= ========                 ===============================
       0     (0)  CHARACTER      10                    Layout ID Value: OPCDUMP
      10     (A)  CHARACTER       1                    Column Separator
      11     (B)  STRUCTURE       8                    4 bytes of hex data
      19    (13)  CHARACTER       1                    Column Separator
      20    (14)  STRUCTURE       8                    4 bytes of hex data
      28    (1C)  CHARACTER       1                    Column Separator
      29    (1D)  STRUCTURE       8                    4 bytes of hex data
      37    (25)  CHARACTER       1                    Column Separator
      38    (26)  STRUCTURE       8                    4 bytes of hex data

=================================================================================
The OPCTITLE record contains blanks or text for an XOH OPC self-defining field
(SDF).

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE       LENGTH                    DESCRIPTION
 ======== ======== ========= ========                 ===============================
       0     (0)  CHARACTER      10                    Layout ID Value: OPCTITLE
      10     (A)  CHARACTER       1                    Column Separator
      11     (B)  CHARACTER     501                    Blanks or SDF title

=================================================================================
The OPCTCONT record contains the continuation of the description for OPCTITLE.

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE       LENGTH                    DESCRIPTION
 ======== ======== ========= ========                 ===============================
       0     (0)  CHARACTER      10                    Layout ID Value: OPCTCONT
      10     (A)  CHARACTER       1                    Column Separator
      11     (B)  CHARACTER     501                    Continuation of SDF title

=================================================================================
The OPCCONT record contains the continuation of the description for other XOH OPC
records that don't have a specific continuation record defined.

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE       LENGTH                    DESCRIPTION
 ======== ======== ========= ========                 ===============================
       0     (0)  CHARACTER      10                    Layout ID Value: OPCCONT
      10     (A)  CHARACTER       1                    Column Separator
      11     (B)  CHARACTER     501                    Continuation of text

=================================================================================
The OPCSDF record contains the hex data for one XOH OPC self-defining field (SDF).

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE       LENGTH                    DESCRIPTION
 ======== ======== ========= ========                 ===============================
       0     (0)  CHARACTER      10                    Layout ID Value: OPCSDF
      10     (A)  CHARACTER       1                    Column Separator
      11     (B)  CHARACTER     501                    Hex SDF data
```

Figure 139. Record formats for the Softcopy Printer Information Report (Page 7 of 12)

```
===================================================================================
The OPCSDFCONT record contains the continuation of the hex data for OPCSDF.

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE       LENGTH              DESCRIPTION
 ======== ======== ========= ========           ==============================
        0     (0) CHARACTER     10               Layout ID Value: OPCSDFCONT
       10     (A) CHARACTER      1               Column Separator
       11     (B) CHARACTER    501               Hex SDF data

===================================================================================
The OPCLENGTH record contains the length of the XOH OPC SDF in hex and in decimal.

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE       LENGTH              DESCRIPTION
 ======== ======== ========= ========           ==============================
        0     (0) CHARACTER     10               Layout ID Value: OPCLENGTH
       10     (A) CHARACTER      1               Column Separator
       11     (B) CHARACTER      7               Length in hex
       18    (12) CHARACTER      1               Column Separator
       19    (13) CHARACTER      6               Length in decimal
       25    (19) CHARACTER      1               Column Separator
       26    (1A) CHARACTER      *               The English description
                                                 delimited by quotes ("s)

===================================================================================
The OPCLENCONT record contains the continuation of the description for OPCLENGTH.

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE       LENGTH              DESCRIPTION
 ======== ======== ========= ========           ==============================
        0     (0) CHARACTER     10               Layout ID Value: OPCLENCONT
       10     (A) CHARACTER     16               Column Separator
       26    (1A) CHARACTER      *               The English description
                                                 delimited by quotes ("s)

===================================================================================
The OPCSDFID record contains the 2-byte self-defining field ID in the XOH OPC
response.

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE       LENGTH              DESCRIPTION
 ======== ======== ========= ========           ==============================
        0     (0) CHARACTER     10               Layout ID Value: OPCSDFID
       10     (A) CHARACTER      1               Column Separator
       11     (B) CHARACTER      7               SDF ID X'nnnn'
       18    (12) CHARACTER      1               Column Separator
       19    (13) CHARACTER      *               The English description
                                                 delimited by quotes ("s)

===================================================================================
The OPCSIDCONT record contains the continuation of the description for OPCSDFID.

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE       LENGTH              DESCRIPTION
 ======== ======== ========= ========           ==============================
        0     (0) CHARACTER     10               Layout ID Value: OPCSIDCONT
       10     (A) CHARACTER      9               Column Separator
       19    (13) CHARACTER      *               The English description
                                                 delimited by quotes ("s)
```

Figure 140. Record formats for the Softcopy Printer Information Report (Page 8 of 12)

```
================================================================================
The OPCCODE record contains 1 or 2 bytes of a hex code value within an XOH OPC SDF.

   OFFSET   OFFSET
 DECIMAL     HEX   TYPE       LENGTH                    DESCRIPTION
 ======== ======== ========= ========                  ===============================
        0      (0) CHARACTER      10                    Layout ID Value: OPCCODE
       10      (A) CHARACTER       1                    Column Separator
       11      (B) CHARACTER       7                    Hex code X'nn' or X'nnnn'
       18     (12) CHARACTER       1                    Column Separator
       19     (13) CHARACTER       *                    The English description
                                                        delimited by quotes ("s)


================================================================================
The OPCCDCONT record contains the continuation of the description for OPCCODE.

   OFFSET   OFFSET
 DECIMAL     HEX   TYPE       LENGTH                    DESCRIPTION
 ======== ======== ========= ========                  ===============================
        0      (0) CHARACTER      10                    Layout ID Value: OPCCDCONT
       10      (A) CHARACTER       9                    Column Separator
       19     (13) CHARACTER       *                    The English description
                                                        delimited by quotes ("s)


================================================================================
The OPCDATA record contains a variable number of hex bytes for XOH OPC SDF entries
where the printer returns printer defined information.

   OFFSET   OFFSET
 DECIMAL     HEX   TYPE       LENGTH                    DESCRIPTION
 ======== ======== ========= ========                  ===============================
        0      (0) CHARACTER      10                    Layout ID Value: OPCDATA
       10      (A) CHARACTER       1                    Column Separator
       11      (B) CHARACTER     501                    Hex code X'nn...' followed by
                                                        the English description
                                                        delimited by quotes ("s)


================================================================================
The OPCDCONT record contains the continuation of OPCDATA.

   OFFSET   OFFSET
 DECIMAL     HEX   TYPE       LENGTH                    DESCRIPTION
 ======== ======== ========= ========                  ===============================
        0      (0) CHARACTER      10                    Layout ID Value: OPCDCONT
       10      (A) CHARACTER       1                    Column Separator
       11      (B) CHARACTER     501                    Continuation of whatever was
                                                        still to be written for OPCDATA
```

Figure 141. Record formats for the Softcopy Printer Information Report (Page 9 of 12)

```
================================================================================
The OPCCHAR record contains a variable length printable text string followed by the
English description of that entry.

   OFFSET    OFFSET
  DECIMAL     HEX  TYPE       LENGTH                   DESCRIPTION
  ======== ======== ========= ========                 ==============================
        0      (0) CHARACTER      10                   Layout ID Value: OPCCHAR
       10      (A) CHARACTER       1                   Column Separator
       11      (B) CHARACTER     501                   Text string without quotes
                                                       followed by
                                                       the English description
                                                       delimited by quotes ("s)


================================================================================
The OPCCCONT record contains the continuation of OPCCHAR.

   OFFSET    OFFSET
  DECIMAL     HEX  TYPE       LENGTH                   DESCRIPTION
  ======== ======== ========= ========                 ==============================
        0      (0) CHARACTER      10                   Layout ID Value: OPCCCONT
       10      (A) CHARACTER       1                   Column Separator
       11      (B) CHARACTER     501                   Continuation of whatever was
                                                       still to be written for OPCCHAR


================================================================================
The OPCVAL record contains 1 or 2 bytes of hex data followed by
the decimal equivalent of the entry and an English description of the value.
in an XOH OPC SDF.

   OFFSET    OFFSET
  DECIMAL     HEX  TYPE       LENGTH                   DESCRIPTION
  ======== ======== ========= ========                 ==============================
        0      (0) CHARACTER      10                   Layout ID Value: OPCVAL
       10      (A) CHARACTER       1                   Column Separator
       11      (B) CHARACTER       7                   Hex data X'nn' or X'nnnn'
       18     (12) CHARACTER       1                   Column Separator
       19     (13) CHARACTER       6                   Value in decimal
       25     (19) CHARACTER       1                   Column Separator
       26     (1A) CHARACTER       *                   The English description
                                                       delimited by quotes ("s)


================================================================================
The OPCVALCONT record contains the continuation of the description of OPCVAL.

   OFFSET    OFFSET
  DECIMAL     HEX  TYPE       LENGTH                   DESCRIPTION
  ======== ======== ========= ========                 ==============================
        0      (0) CHARACTER      10                   Layout ID Value: OPCVALCONT
       10      (A) CHARACTER      16                   Column Separator
       26     (1A) CHARACTER       *                   The English description
                                                       delimited by quotes ("s)
```

Figure 142. Record formats for the Softcopy Printer Information Report (Page 10 of 12)

```
================================================================================
The OPCVAL4 record contains 4 bytes of hex data followed by
the decimal equivalent of the entry and an English description of the value
in an XOH OPC SDF.

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE      LENGTH                DESCRIPTION
 ======== ======== ========= ========            ===============================
       0     (0) CHARACTER      10                Layout ID Value: OPCVAL4
      10     (A) CHARACTER       1                Column Separator
      11     (B) CHARACTER      11                Hex data X'nnnnnnnn'
      22    (16) CHARACTER       1                Column Separator
      23    (17) CHARACTER      13                Value in decimal
      36    (24) CHARACTER       1                Column Separator
      37    (25) CHARACTER       *                The English description
                                                  delimited by quotes ("s)


================================================================================
The OPCV4CONT record contains the continuation of the description for OPCVAL4.

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE      LENGTH                DESCRIPTION
 ======== ======== ========= ========            ===============================
       0     (0) CHARACTER      10                Layout ID Value: OPCV4CONT
      10     (A) CHARACTER      27                Column Separator
      37    (25) CHARACTER       *                The English description
                                                  delimited by quotes ("s)


================================================================================
The OPCBITS record has 1 or 2 bytes of hex data followed by the bit string.
equivalent and the English description.

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE      LENGTH                DESCRIPTION
 ======== ======== ========= ========            ===============================
       0     (0) CHARACTER      10                Layout ID Value: OPCBITS
      10     (A) CHARACTER       1                Column Separator
      11     (B) CHARACTER       7                Hex data X'nn' or X'nnnn'
      18    (12) CHARACTER       1                Column separator
      19    (13) CHARACTER      19                Bit string B'nnn...n'
      38    (26) CHARACTER       1                Column separator
      39    (27) CHARACTER       *                The English description
                                                  delimited by quotes ("s)


================================================================================
The OPCBCONT record contains the continuation of the description for OPCBITS
equivalent and the English description.

  OFFSET   OFFSET
 DECIMAL    HEX   TYPE      LENGTH                DESCRIPTION
 ======== ======== ========= ========            ===============================
       0     (0) CHARACTER      10                Layout ID Value: OPCBCONT
      10     (A) CHARACTER      29                Column Separator
      39    (27) CHARACTER       *                The English description
                                                  delimited by quotes ("s)
```

Figure 143. Record formats for the Softcopy Printer Information Report (Page 11 of 12)

```
================================================================================
The OPCBIT record contains a single bit entry value followed by its description.

  OFFSET   OFFSET
 DECIMAL     HEX  TYPE       LENGTH                   DESCRIPTION
 ======== ======== ========= ========                 ==============================
       0      (0) CHARACTER      10                   Layout ID Value: OPCBIT
      10      (A) CHARACTER       1                   Column Separator
      11      (B) CHARACTER       9                   Dotted bit string (1... ....)
      20     (14) CHARACTER       1                   Column separator
      21     (15) CHARACTER       *                   The English description
                                                      delimited by quotes ("s)


================================================================================
The OPCBITCONT record contains the continuation of the description for OPCBIT.

  OFFSET   OFFSET
 DECIMAL     HEX  TYPE       LENGTH                   DESCRIPTION
 ======== ======== ========= ========                 ==============================
       0      (0) CHARACTER      10                   Layout ID Value: OPCBITCONT
      10      (A) CHARACTER      11                   Column Separator
      21     (15) CHARACTER       *                   The English description
                                                      delimited by quotes ("s)


================================================================================
The OPCCODE16 record has 16 bytes of hex data followed by its English description.

  OFFSET   OFFSET
 DECIMAL     HEX  TYPE       LENGTH                   DESCRIPTION
 ======== ======== ========= ========                 ==============================
       0      (0) CHARACTER      10                   Layout ID Value: OPCCODE16
      10      (A) CHARACTER       1                   Column Separator
      11      (B) CHARACTER      35                   Hex data X'nn...n'
      46     (2E) CHARACTER       1                   Column separator
      47     (2F) CHARACTER       *                   The English description
                                                      delimited by quotes ("s)


================================================================================
The OPCC16CONT record has the continuation of the description for OPCCODE16.

  OFFSET   OFFSET
 DECIMAL     HEX  TYPE       LENGTH                   DESCRIPTION
 ======== ======== ========= ========                 ==============================
       0      (0) CHARACTER      10                   Layout ID Value: OPCC16CONT
      10      (A) CHARACTER      37                   Column Separator
      47     (2F) CHARACTER       *                   The English description
                                                      delimited by quotes ("s)


================================================================================
The OPCEND record indicates the end of the XOH OPC data.

  OFFSET   OFFSET
 DECIMAL     HEX  TYPE       LENGTH                   DESCRIPTION
 ======== ======== ========= ========                 ==============================
       0      (0) CHARACTER      10                   Layout ID Value: OPCEND
      10      (A) CHARACTER       1                   Column Separator
      11      (B) CHARACTER     501                   Blanks
```

Figure 144. Record formats for the Softcopy Printer Information Report (Page 12 of 12)

Appendix C. APSRMARK sample jobs

This information lists the APSRMARK sample jobs that are included with PSF. Table 39 on page 399 lists the SYS1.SAMPLIB member names and associates each sample job with a corresponding printer resource diskette or cartridge.

Table 39. Sample jobs in SYS1.SAMPLIB		
Member name	**Description**	**Printer that might have these resident (some are optional font cards)**
APSWMCPG	AFCCU code pages	Any AFCCU printer, 3930 printer (see the documentation provided with the printer.)
APSWMCR	Core Interchange Courier	AFCCU, 3930 printer
APSWMGML	DCF/GML	3812 and 3816 printers
APSWMGR4	3270/IDS, DCF/GML, Language Group 2, Language Group 3, and Language Group 4	3812 and 3816 printers, and any printer with a 3812/3816 compatible font set
APSWMHLV	Core Interchange Helvetica	AFCCU, 3930 printer
APSWMLG2	Language Group 2	3812 and 3816 printers
APSWMLG3	Language Group 3	3812 and 3816 printers
APSWMLG4	Language Group 4	3812 and 3816 printers
APSWMRES	Generic job to mark individual members in a font library as PUBLIC, and replace them	Any printer with resident fonts
APSWMROM	Kanji fonts	3820 printer with ROM Font RPQ, 3130 printer
APSWMRPM	Generic job to mark all members in a font library as PUBLIC	Any printer attached through DPF
APSWMSTD	3270 IDS diskette	3812 and 3816 printers
APSWMTNR	Core Interchange Times New Roman	AFCCU, 3930 printer
APSW4028	4028 resident fonts	4028 printer and any printer with a 4028 compatible font set
APSWRRID	Job to mark specific members of a raster font library for printer capture; specifies the RRID for each character set and code page	Not applicable
APSWMALL	Job to mark all members of a raster font library for printer capture; uses "MEMBER=ALL" notation	Not applicable

Notes:

1. None of the sample jobs that are listed in Table 39 on page 399 runs as provided. Each sample job requires modification to the control statements or the JCL.
2. Some of the listings of jobs in the tables include recommended procedures. If the procedures are available, they are given in the comment section at the top of each member.

3. Language Group diskettes or cartridges might not be available in your area. Check with your IBM marketing representative for more information.

4. These sample jobs do not include statements for any font simulations that use font substitutions.

5. For a 300-pel printer, you must create a 300-pel font library by using APSRCF30, and then mark it with these APSRMARK jobs. For more information, see .

Appendix D. APSRMARK long report format

[Figure 145 on page 401](#)–[Figure 146 on page 402](#) show an example of a long report format for resources marked with APSRMARK.

```
APS580I       INDD=OUT2,MEMBER=ALL,REPORT,LONG
APS580I CZB50C   RESOURCE TYPE: FONT CHARACTER SET (OUTLINE, RELATIVE METRICS)
                 INTERNAL ID:   FGID=00000(X'0000') WIDTH=00000(X'0000') GCSGID=00000(X'0000')
                 TIME(CREATED): YEAR=2001 MONTH=12 DAY=25 TIME=17:13:16.06
                 TIME(REVISED): NOT SPECIFIED IN RESOURCE
                 TIME(MARKED):  YEAR=2004 MONTH=02 DAY=29 TIME=14:32:56.06
                 CHANGE LEVEL:  NOT SPECIFIED IN RESOURCE
                 MANAGEMENT:    PUBLIC=YES CRC=9763
                 OBJECT ORIGIN: SYSTEM=661200354381 OS=MVS VOLUME=PSFDVT DSN=L1111111.L2222222.L3333333.L4444444.L5555555
                 ENABLED FOR:   CAPTURE(DPF)=NO  CAPTURE(PRINTER)=NO  RESIDENT RESOURCE ACTIVATION=NO
APS580I CZMYNEW1 RESOURCE TYPE: FONT CHARACTER SET (RASTER,  RELATIVE METRICS, PEL=300)
                 INTERNAL ID:   FGID=00040(X'0028') WIDTH=00144(X'0090') GCSGID=00274(X'0112')
                 TIME(CREATED): YEAR=2004 MONTH=02 DAY=29 TIME=14:32:56.06
                 TIME(REVISED): NOT SPECIFIED IN RESOURCE
                 TIME(MARKED):  YEAR=2004 MONTH=04 DAY=30 TIME=09:27:06.06
                 CHANGE LEVEL:  FGID=00040(X'0028') WIDTH=00144(X'0090') YEAR=2004 MONTH=03 DAY=01 TIME=08:30:00.00
                 MANAGEMENT:    PUBLIC=YES CRC=9763
                 OBJECT ORIGIN: SYSTEM=661200354381 OS=MVS VOLUME=PSFDVT DSN=L1111111.L2222222.L3333333.L4444444.L5555555
                 ENABLED FOR:   CAPTURE(DPF)=YES CAPTURE(PRINTER)=YES RESIDENT RESOURCE ACTIVATION=YES
APS580I C0FACTRY RESOURCE TYPE: FONT CHARACTER SET (RASTER, RELATIVE METRICS, PEL=300)
                 INTERNAL ID:   FGID=00040(X'0028') WIDTH=00144(X'0090') GCSGID=00274(X'0112')
                 TIME(CREATED): YEAR=2004 MONTH=02 DAY=29 TIME=14:32:56.06
                 TIME(REVISED): NOT SPECIFIED IN RESOURCE
                 TIME(MARKED):  NOT SPECIFIED IN RESOURCE
                 CHANGE LEVEL:  NOT SPECIFIED IN RESOURCE
                 MANAGEMENT:    PUBLIC=YES CRC=9763
                 OBJECT ORIGIN: NOT SPECIFIED IN RESOURCE
                 ENABLED FOR:   CAPTURE(DPF)=NO  CAPTURE(PRINTER)=YES RESIDENT RESOURCE ACTIVATION=YES
APS580I C0NOMARK RESOURCE TYPE: FONT CHARACTER SET (RASTER,  FIXED METRICS,     PEL=240)
                 INTERNAL ID:   FGID=00040(X'0028') WIDTH=00144(X'0090') GCSGID=00274(X'0112')
                 TIME(CREATED): NOT SPECIFIED IN RESOURCE
                 TIME(REVISED): NOT SPECIFIED IN RESOURCE
                 TIME(MARKED):  NOT SPECIFIED IN RESOURCE
                 CHANGE LEVEL:  NOT SPECIFIED IN RESOURCE
                 MANAGEMENT:    NOT SPECIFIED IN RESOURCE
                 OBJECT ORIGIN: NOT SPECIFIED IN RESOURCE
                 ENABLED FOR:   CAPTURE(DPF)=NO  CAPTURE(PRINTER)=NO  RESIDENT RESOURCE ACTIVATION=NO
APS580I F1NOBOX  RESOURCE TYPE: FORMDEF
                 ENABLED FOR:   CAPTURE(DPF)=NO  CAPTURE(PRINTER)=NO  RESIDENT RESOURCE ACTIVATION=NO
APS580I M1UBLPK  RESOURCE TYPE: OBJECT CONTAINER   (COLOR MAPPING TABLE)
                 ENABLED FOR:   CAPTURE(DPF)=NO  CAPTURE(PRINTER)=NO  RESIDENT RESOURCE ACTIVATION=NO
APS580I I1MYNEW1 RESOURCE TYPE: IOCA IMAGE
                 ENABLED FOR:   CAPTURE(DPF)=NO  CAPTURE(PRINTER)=NO  RESIDENT RESOURCE ACTIVATION=NO
APS580I O1ROI000 RESOURCE TYPE: OVERLAY
                 TIME(MARKED):  YEAR=2004 MONTH=02 DAY=29 TIME=14:32:56.06
                 MANAGEMENT:    PUBLIC=YES CRC=9763
                 OBJECT ORIGIN: SYSTEM=661200354381 OS=MVS VOLUME=PSFDVT DSN=PSFMVS.LISZ.MARKED.RESOURCE
                 ENABLED FOR:   CAPTURE(DPF)=YES CAPTURE(PRINTER)=NO  RESIDENT RESOURCE ACTIVATION=NO
APS580I P1SEC5   RESOURCE TYPE: PAGEDEF
                 ENABLED FOR:   CAPTURE(DPF)=NO  CAPTURE(PRINTER)=NO  RESIDENT RESOURCE ACTIVATION=NO
APS580I S1SCHLAF RESOURCE TYPE: PAGE SEGMENT
                 TIME(MARKED):  YEAR=2004 MONTH=02 DAY=29 TIME=14:32:56.06
                 MANAGEMENT:    PUBLIC=YES CRC=9763
                 OBJECT ORIGIN: SYSTEM=661200354381 OS=MVS VOLUME=PSFDVT DSN=DOSCHE.PRIVATE.MARKED.PSEGMENT.RESOURCE
                 ENABLED FOR:   CAPTURE(DPF)=YES CAPTURE(PRINTER)=NO  RESIDENT RESOURCE ACTIVATION=NO
APS580I T1L000XN RESOURCE TYPE: CODE PAGE          (SINGLE BYTE, EBCDIC)
                 INTERNAL ID:   GCSGID=00000(X'0000') CPGID=00000(X'0000')
                 TIME(CREATED): YEAR=1999 MONTH=11 DAY=30 TIME=10:32:46.06
                 TIME(REVISED): YEAR=2001 MONTH=02 DAY=19 TIME=04:23:36.06
                 TIME(MARKED):  YEAR=2000 MONTH=05 DAY=26 TIME=20:17:56.06
                 CHANGE LEVEL:  NOT SPECIFIED IN RESOURCE
                 MANAGEMENT:    PUBLIC=YES CRC=9763
                 OBJECT ORIGIN: SYSTEM=661200354381 OS=MVS VOLUME=PSFDVT DSN=L1111111.L2222222.L3333333.L4444444.L5555555
                 ENABLED FOR:   CAPTURE(DPF)=YES CAPTURE(PRINTER)=NO  RESIDENT RESOURCE ACTIVATION=NO
```

Figure 145. Printed output of printer-resident long report (Part 1 of 2)

```
APS580I  T1MYNEW1 RESOURCE TYPE: CODE PAGE          (SINGLE BYTE, EBCDIC)
                  INTERNAL ID:  GCSGID=00307(X'0133') CPGID=00297(X'0129')
                  TIME(CREATED): YEAR=1999 MONTH=11 DAY=30 TIME=10:32:46.06
                  TIME(REVISED): YEAR=2001 MONTH=02 DAY=19 TIME=04:23:36.06
                  TIME(MARKED):  YEAR=2000 MONTH=05 DAY=26 TIME=20:17:56.06
                  CHANGE LEVEL:  GCSGID=00307(X'0133') CPGID=00297(X'0129') YEAR=1997 MONTH=03 DAY=01 TIME=08:30:00.00
                  MANAGEMENT:    PUBLIC=YES CRC=9763
                  OBJECT ORIGIN: SYSTEM=661200354381 OS=MVS VOLUME=PSFDVT DSN=L1111111.L2222222.L3333333.L4444444.L5555555
                  ENABLED FOR:   CAPTURE(DPF)=YES CAPTURE(PRINTER)=YES RESIDENT RESOURCE ACTIVATION=YES
APS580I  T1BASIC1 RESOURCE TYPE: CODE PAGE          (SINGLE BYTE, EBCDIC)
                  INTERNAL ID:  GCSGID=00307(X'0133') CPGID=00297(X'0129')
                  TIME(CREATED): YEAR=2004 MONTH=02 DAY=29 TIME=14:32:56.06
                  TIME(REVISED): NOT SPECIFIED IN RESOURCE
                  TIME(MARKED):  NOT SPECIFIED IN RESOURCE
                  CHANGE LEVEL:  NOT SPECIFIED IN RESOURCE
                  MANAGEMENT:    PUBLIC=YES CRC=9763
                  OBJECT ORIGIN: NOT SPECIFIED IN RESOURCE
                  ENABLED FOR:   CAPTURE(DPF)=NO  CAPTURE(PRINTER)=YES RESIDENT RESOURCE ACTIVATION=YES
APS580I  XZH0FN   RESOURCE TYPE: CODED FONT
                  ENABLED FOR:   CAPTURE(DPF)=NO  CAPTURE(PRINTER)=NO  RESIDENT RESOURCE ACTIVATION=NO
APS580I  E1U0S001 RESOURCE TYPE: UNKNOWN
```

Figure 146. Printed output of printer-resident long report (Part 2 of 2)

An APSRMARK long report format contains four areas of data for each resource object:

Message number
The message number is contained in columns 1–7 and defines each new resource object listed in the report. All resource objects that are listed in the long report have the message number APS580I.

Resource name
The resource name is contained in columns 10–17 and identifies the resource object.

Data category
The data category is contained in columns 19–32 and indicates the type of data reported for a resource object. The data categories that are listed for a resource object are determined by its resource type. See "Data categories" on page 403 for information about the different data categories.

Data description
The data description is contained in columns 34–122 and describes the settings for a data category. The settings for a resource object determine resident resource activation and resource capture.

Table 40 on page 402 shows examples of the four data areas on a long report.

Table 40. Data areas on an APSRMARK long report

Message Number	Resource Name	Data Category	Data Description
APS580I	CZB50C	RESOURCE TYPE:	FONT CHARACTER SET (OUTLINE, RELATIVE METRICS)
		INTERNAL ID:	FGID=00000(X'0000') WIDTH=00000(X'0000') GCSGID=00000(X'0000')
		TIME(CREATED):	YEAR=2001 MONTH=12 DAY=25 TIME=17:13:16.06
		TIME(REVISED):	NOT SPECIFIED IN RESOURCE
		TIME(MARKED):	YEAR=2004 MONTH=02 DAY=29 TIME=14:32:56.06
		CHANGE LEVEL:	NOT SPECIFIED IN RESOURCE
		MANAGEMENT:	PUBLIC=YES CRC=9763
		OBJECT ORIGIN:	SYSTEM=661200354381 OS=MVS VOLUME=PSFDVT DSN=L1111111.L2222222.L3333333.L4444444.L5555555
		ENABLED FOR:	CAPTURE(DPF)=NO CAPTURE(PRINTER)=NO RESIDENT RESOURCE ACTIVATION=NO

Table 40. Data areas on an APSRMARK long report *(continued)*			
Message Number	**Resource Name**	**Data Category**	**Data Description**
APS580I	F1NOBOX	RESOURCE TYPE:	FORMDEF
		ENABLED FOR:	CAPTURE(DPF)=NO CAPTURE(PRINTER)=NO RESIDENT RESOURCE ACTIVATION=NO
APS580I	S1SCHLAF	RESOURCE TYPE:	PAGE SEGMENT
		TIME(MARKED):	YEAR=2004 MONTH=02 DAY=29 TIME=14:32:56.06
		MANAGEMENT:	PUBLIC=YES CRC=9763
		OBJECT ORIGIN:	SYSTEM=661200354381 OS=MVS VOLUME=PSFDVT DSN=DOSCHE.PRIVATE.MARKED.PSEGMENT.RESOURCE
		ENABLED FOR:	CAPTURE(DPF)=YES CAPTURE(PRINTER)=NO RESIDENT RESOURCE ACTIVATION=NO

Data categories

The data categories for a resource object can include:

- Resource Type
- Internal ID
- Time(Created)
- Time(Revised)
- Time(Marked)
- Change Level
- Management
- Object Origin
- Enabled For

The Resource Type and Enabled For categories are listed for all recognized MO:DCA resource objects in the data set. The other categories that are listed are determined by the resource type (see "Resource type" on page 403).

Resource type

The Resource Type category always contains one of the following settings in the data description area of the report:

CODED FONT
Only the Resource Type and the Enabled For categories are listed for a resource object when its resource type is CODED FONT.

CODE PAGE (*fonttype, encodingscheme*)

fonttype
Specifies the type of fonts the resource is used for:

- SINGLE BYTE
- DOUBLE BYTE

encodingscheme
Specifies the encoding scheme of the code page:

- ASCII
- EBCDIC

The Resource Type, Internal ID, and Enabled For categories are listed for a resource object when its resource type is CODE PAGE.

FONT CHARACTER SET (*technology, metrics,* PEL=*ppp*)

> ***technology***
> Specifies the technology of the font:
>
> - RASTER
> - OUTLINE
>
> ***metrics***
> Specifies the type of metrics used in the font:
>
> - FIXED METRIC
> - RELATIVE METRIC
>
> **PEL=*ppp***
> Specifies the shape resolution of a raster font. For example, *ppp* is:
>
> - 240
> - 300

The Resource Type, Internal ID, and Enabled For categories are listed for a resource object when its resource type is FONT CHARACTER SET.

Note: In some cases, such as metric only fonts, the shape resolution is not available in the FOCA Font Control (FNC) structured field. If shape resolution is not available, this value is not listed on the report.

FORMDEF
Only the Resource Type and the Enabled For categories are listed for a resource object when its resource type is FORMDEF.

IOCA IMAGE
Only the Resource Type and Enabled For categories are listed for a resource object when its resource type is IOCA IMAGE.

OBJECT CONTAINER (*object container type*)

> ***object container type***
> Specifies the type of object container. For example:
>
> - COLOR MAPPING TABLE
> - COMSETUP
> - ENCAPSULATED POSTSCRIPT
> - IOCA TILE RESOURCE OBJECT
> - PDF SINGLE-PAGE OBJECT
> - PDF RESOURCE OBJECT

Only the Resource Type and the Enabled For categories are listed for a resource object when its resource type is OBJECT CONTAINER.

OVERLAY
The Resource Type and Enabled For categories are listed for a resource object when its resource type is OVERLAY.

PAGEDEF
Only the Resource Type and the Enabled For categories are listed for a resource object when its resource type is PAGEDEF.

PAGE SEGMENT
    The Resource Type and Enabled For categories are listed for a resource object when its resource type is PAGE SEGMENT.

UNKNOWN
    Only the Resource Type category is listed for a resource object when its resource type is UNKNOWN.

Note: The Bxx structured field contains the data for the Resource Type category.

Internal ID

The Internal ID category defines global resource identifier (GRID) parts. Internal IDs are only listed if the resource type is CODE PAGE or FONT CHARACTER SET; the settings that are listed in the data description area depend on the resource type:

- CODE PAGE:

  GCSGID=00000(X'0000')
      Graphic character set global identifier

  CPGID=00000(X'0000')
      Code page global identifier

- FONT CHARACTER SET:

  FGID=00000(X'0000')
      Font typeface global identifier

  WIDTH=00000(X'0000')
      Font width

  GCSGID=00000(X'0000')
      Graphic character set global identifier

Note: The CPD structured field contains the Internal ID data for a code page resource type. The FND structured field contains the Internal ID data for a font character set resource.

Time(Created)

The Time(Created) category specifies the date and time the resource object was manufactured. Time(Created) is only listed if the resource type is CODE PAGE or FONT CHARACTER SET. The settings that are listed in the data description area are:

YEAR=yyyy MONTH=mm DAY=dd
    The date the resource object was manufactured.

TIME=hh:mm:ss.hh
    The hour, minute, second, and hundredth of a second the resource object was manufactured.

Notes:

1. The date and time are listed if they were included in the resource; otherwise, NOT SPECIFIED IN RESOURCE is listed in the data description area.

2. The Time(Created) category is only used in processing outline fonts and for resident resource activation and printer capture.

3. The Time(Created) category is only listed in the report when the Bxx structured field contains an X'62' type X'00' triplet.

Time(Revised)

The Time(Revised) category specifies the date and time the resource object was last revised by the manufacturer. Time(Revised) is only listed if the resource type is CODE PAGE or FONT CHARACTER SET. The settings that are listed in the data description area are:

YEAR=yyyy MONTH=mm DAY=dd
    The date the resource object was last revised.

TIME=hh:mm:ss.hh
  The hour, minute, second, and hundredth of a second the resource object was last revised.

Notes:

1. The date and time are listed if they were included in the resource; otherwise, NOT SPECIFIED IN RESOURCE is listed in the data description area.

2. The Time(Revised) category is only used in processing outline fonts and for resident resource activation and printer capture.

3. The Time(Revised) category is only listed in the report when the Bxx structured field contains an X'62' type X'03' triplet.

Time(Marked)

The Time(Marked) category specifies the date and time the resource object was last marked with APSRMARK. The settings that are listed in the data description area are:

YEAR=yyyy MONTH=mm DAY=dd
  The date the resource object was marked.

TIME=hh:mm:ss.hh
  The hour, minute, second, and hundredth of a second the resource object was marked.

Notes:

1. The date and time are listed if the resource object is marked with APSRMARK; otherwise, NOT SPECIFIED IN RESOURCE is listed in the data description area.

2. The Time(Marked) category is only used for distributed print function (DPF) and Remote Print Manager (RPM) attached printers.

3. The Time(Marked) category is only listed in the report when the Bxx structured field contains an X'62' type X'01' triplet.

Change Level

The Change Level category defines the Engineering Change (EC) level of the resource object, including global resource identifier (GRID) parts and the date and time the resource was marked. The settings that are listed in the data description area depend on whether the resource type is CODE PAGE or FONT CHARACTER SET:

- CODE PAGE:

  GCSGID=00000(X'0000')
    Graphic character set global identifier, which is specified as four HEX digits.

  CPGID=00000(X'0000')
    Code page global identifier, which is specified as four HEX digits.

  YEAR=yyyy MONTH=mm DAY=dd
    The date the resource object was marked.

  TIME=hh:mm:ss.hh
    The hour, minute, second, and hundredth of a second the resource object was marked.

- FONT CHARACTER SET:

  FGID=00000(X'0000')
    Font typeface global identifier, which is specified as four HEX digits.

  WIDTH=00000(X'0000')
    Font width, which is specified as four HEX digits.

  YEAR=yyyy MONTH=mm DAY=dd
    The date the resource object was marked.

  TIME=hh:mm:ss.hh
    The hour, minute, second, and hundredth of a second the resource object was marked.

Notes:

1. The GRID parts, date, and time are listed if the resource object is marked with APSRMARK; otherwise, NOT SPECIFIED IN RESOURCE is listed in the data description area.

2. The GRID parts, date, and time are derived from the RRID, RRDATE, and RRTIME parameters in APSRMARK.

3. The change level is intended to match the change level of the identical resource object included with a printer. The change level can be set in the resource object with APSRMARK.

4. The Change Level category is used for resident resource activation of raster fonts.

5. The Change Level category is only listed in the report when the Bxx structured field contains an X'63' type X'02' triplet.

Management

The Management category specifies resource object attributes. The settings that are listed in the data description area are:

PUBLIC=YES|NO
Indicates whether the resource is public or private.

CRC=nnnn
Specifies the Cyclic Redundancy Check number that APSRMARK computes the last time that the resource is marked.

Notes:

1. The attributes are listed if the resource object is marked with APSRMARK; otherwise, NOT SPECIFIED IN RESOURCE is listed in the data description area.

2. The Management category is only listed in the report when the Bxx structured field contains an X'63' type X'01' triplet.

Object Origin

The Object Origin category indicates where the resource object was residing when it was marked by APSRMARK. The settings that are listed in the data description area are:

SYSTEM=nnnnnnnnnnnn
Identifies the physical system where APSRMARK ran the last time that the resource was marked.

OS=xxx
Identifies the operating system where APSRMARK ran the last time that the resource was marked. MVS indicates z/OS.

VOLUME=xxxxxx
Identifies the physical volume serial number for the device that contains the PDS with this resource when APSRMARK was last used to mark the resource.

DSN=xxxxxxxx.xxxxxxxx.xxxxxxxx.xxxxxxxx.xxxxxxxx
Identifies the partitioned data set where this resource resided the last time APSRMARK was used to mark the resource.

Notes:

1. The location data is listed if the resource object is marked with APSRMARK; otherwise, NOT SPECIFIED IN RESOURCE is listed in the data description area.

2. The Object Origin category is only listed in the report when the Bxx structured field contains an X'64' triplet.

Enabled For

The Enabled For category indicates whether the resource object is enabled for resource capture and resident resource activation. The settings that are listed in the data description area are:

CAPTURE(DPF)=YES|NO
> Indicates whether a resource object is enabled for capture by a DPF-attached printer.

CAPTURE(PRINTER)=YES|NO
> Indicates whether a resource object is enabled for capture by a printer that supports resource capture.

RESIDENT RESOURCE ACTIVATION=YES|NO
> Indicates whether a resource object is enabled for resident resource activation.

APSRMARK determines the settings for the Enabled For category based on the resource type and the settings established for the other data categories. Table 41 on page 408 indicates how the Enabled For settings are determined. All conditions that are listed in the table must be true for the setting to equal YES; otherwise, the setting equals NO.

Table 41. Enabled For settings

Resource Type	CAPTURE(DPF)=YES	CAPTURE(PRINTER)=YES	RESIDENT RESOURCE ACTIVATION=YES
FONT CHARACTER SET	• The font is RASTER. • All Time(Marked) data is present. • All Management data is available and PUBLIC=YES. • All Object Origin data is present.	OUTLINE: • The font is OUTLINE. • FGID data is specified for Internal ID, Change Level, or both, and is not zero. • Date and time data is available for Time(Created), Time(Revised), or Change Level. • PUBLIC=YES for Management. RASTER: • The font is RASTER. • FGID data is specified for Internal ID, Change Level, or both, and is not zero. • WIDTH data is specified for Internal ID, Change Level, or both, and is not zero. • Date and time data is available for Change Level. • PUBLIC=YES for Management.	OUTLINE: • The font is OUTLINE. • FGID data is specified for Internal ID, Change Level, or both, and is not zero. • PUBLIC=YES for Management. RASTER: • The font is RASTER. • FGID data is specified for Internal ID, Change Level, or both, and is not zero. • WIDTH data is specified for Internal ID, Change Level, or both, and is not zero. • PUBLIC=YES for Management.

Resource Type	CAPTURE(DPF)=YES	CAPTURE(PRINTER)=YES	RESIDENT RESOURCE ACTIVATION=YES
Table 41. Enabled For settings (continued)			
CODE PAGE	• All Time(Marked) data is present. • All Management data is available and PUBLIC=YES. • All Object Origin data is present.	• GCSGID data is specified for Internal ID, Change Level, or both, and is not zero. • CPGID data is specified for Internal ID, Change Level, or both, and is not zero. • Date and time data is available for Change Level. • PUBLIC=YES for Management. **Note:** Even if GCSGID data is not present, the CODE PAGE resource might get captured if the FONT CHARACTER SET resource it is paired with has GCSGID data. The CODE PAGE resource inherits the GCSGID from the FONT CHARACTER SET resource in this case.	• GCSGID data is specified for Internal ID, Change Level, or both, and is not zero. • CPGID data is specified for Internal ID, Change Level, or both, and is not zero. • PUBLIC=YES for Management. **Note:** It is possible to have a CODE PAGE that does not have a GCSGID; therefore, it is marked RESIDENT RESOURCE ACTIVATION=NO. However, when the CODE PAGE is used and the GCSGID is found in the FONT CHARACTER SET resource that it was paired with, a valid resource activation can take place.
OVERLAY and PAGE SEGMENT	• All Time(Marked) data is present. • All Management data is available and PUBLIC=YES. • All Object Origin data is present.	No conditions are true; setting always equals NO.	No conditions are true; setting always equals NO.

Note: PSF for z/OS contains support for Font Mapping Tables. With this support, a specified font might get replaced with an entirely different font. The Enabled For data is presented in the APSRMARK report regardless of the effect the Font Mapping Table support has on the actual font processing and font substitution.

Appendix E. FOCA font resource tables supplied by PSF

This information contains information for intended Programming Interfaces.

The following FOCA font resource tables, which are provided with PSF, are intended as samples. The routines as provided might not be suitable for your requirements.

The APSRFTBL font resource table module is provided on the PSF distribution tapes.

APSRFTBL font resource table

"APSRFTBL resident font resource table" on page 411 is a copy of the module APSRFTBL, which is provided in SYS1.SAMPLIB. The host code pages and character sets that are listed in this font resource table are equivalent to the resident symbol sets in the 4224, 4230, 4234, 4247, 64xx, and 65xx printers, and are used only with those printers.

PSF searches the table of resident symbol sets to find table entries that match the member names specified in the Map Coded Font structured fields. If PSF finds a match for both the font character set and the code page member names, PSF builds the Load Font Equivalence (LFE) command by using information from the entry in the table of resident symbol sets. If PSF cannot find resident symbol-set table entries that match the names that are specified in the Map Coded Font structured field, PSF ends the current page and issues messages that say that the requested resource cannot be identified as a resident symbol set.

Unlike other printer-resident font table samples provided, the APSRFTBL font resource table does not represent fonts that can be marked by use of APSRMARK. PSF supports only the resident symbol sets; it does not download fonts to the supported printers.

If you edit any table, do not change its last line (which contains the characters 'FFFFFFFF'); all code for that table must be placed before that last line.

APSRFTBL resident font resource table

```
TITLE:   APSRFTBL RESIDENT FONT RESOURCE TABLES
APSRFTBL START 0
*/****START OF SPECIFICATIONS********************EXTPROC*************/
*/*                                                          */
*/*      MODULE NAME: APSRFTBL                               */
*/*                                                          */
*/*      DESCRIPTIVE NAME: RESIDENT FONT RESOURCE TABLES     */
*/*                                                          */
*/*      FUNCTION: THESE TABLES ALLOW PSF TO DETERMINE THE   */
*/*                FONT CHARACTER SETS AND CODE PAGES THAT MAY BE   */
*/*                RESIDENT IN THE SYMBOL SET PRINTER(S). THE   */
*/*                USER MAY UPDATE THESE TABLES AND CHANGE THE   */
*/*                MAPPINGS. IF ANY CHANGES ARE MADE, THEN THIS   */
*/*                FILE MUST BE ASSEMBLED AND RE-LINKEDITED.    @06C*/
*/*                                                          */
*/*      NOTES:                                              */
*/*         DEPENDENCIES: NONE                               */
*/*         RESTRICTIONS: NONE                               */
*/*         REGISTER CONVENTIONS: NONE                       */
*/*         PATCH LABEL: NONE                                */
*/*                                                          */
*/*      MODULE TYPE:  PROCEDURE                             */
*/*         PROCESSOR: OS ASSEMBLER H                        */
*/*         MODULE SIZE: 900                                 */
*/*         ATTRIBUTES: READ ONLY, NO EXECUTABLE CODE        */
*/*                                                          */
*/*                                                          */
```

Figure 147. APSRFTBL resident font resource table (Part 1 of 10)

```
*/*       ENTRY POINT: APSCS24V                                 */
*/*         PURPOSE:  THIS IS THE TABLE OF RESIDENT FONT CHARACTER    */
*/*                   SETS IN THE 4224 PRINTER. THIS TABLE IS     @02C*/
*/*                   ALSO USED BY THE 4230 AND 4247 PRINTERS WHEN    */
*/*                   RUN IN 4224 EMULATION MODE.                @05A*/
*/*                   IT CAN BE CHANGED BY THE USER.                  */
*/*                   THE TABLE IS TERMINATED BY A STRING OF          */
*/*                   CL8'FFFFFFFF'.                                  */
*/*         LINKAGE:  NONE                                            */
*/*                                                                   */
*/*       ENTRY POINT: APSCP24V                                       */
*/*         PURPOSE:  THIS IS A TABLE OF RESIDENT CODE PAGES IN THE   */
*/*                   4224 PRINTER. THIS TABLE IS ALSO USED BY    @02C*/
*/*                   THE 4230 AND 4247 PRINTERS WHEN RUN IN 4224     */
*/*                   EMULATION MODE.                            @05A*/
*/*                   IT CAN BE CHANGED BY THE USER.                  */
*/*                   THE TABLE IS TERMINATED BY A STRING OF          */
*/*                   CL8'FFFFFFFF'.                                  */
*/*         LINKAGE:  NONE                                            */
*/*                                                                   */
*/*       ENTRY POINT: APSCS34V                                       */
*/*         PURPOSE:  THIS IS THE TABLE OF RESIDENT FONT CHARACTER    */
*/*                   SETS IN THE 4234 PRINTER. THIS TABLE IS ALSO    */
*/*                   USED BY THE 64XX AND 65XX PRINTERS WHEN RUN IN  */
*/*                   4234 EMULATION MODE.  IT CAN BE CHANGED     @08C*/
*/*                   BY THE USER. THE TABLE IS TERMINATED BY A       */
*/*                   A STRING OF CL8'FFFFFFFF'.                      */
*/*         LINKAGE:  NONE                                            */
*/*                                                                   */
*/*       ENTRY POINT: APSCP34V                                       */
*/*         PURPOSE:  THIS IS A TABLE OF RESIDENT CODE PAGES IN THE   */
*/*                   4234 PRINTER. THIS TABLE IS ALSO USED BY THE    */
*/*                   64XX AND 65XX PRINTERS WHEN RUN IN 4234         */
*/*                   EMULATION MODE. IT CAN BE CHANGED BY THE    @08C*/
*/*                   USER.                                      @05A*/
*/*                   THE TABLE IS TERMINATED BY A STRING OF          */
*/*                   CL8'FFFFFFFF'.                                  */
*/*         LINKAGE:  NONE                                            */
*/*                                                                   */
*/*       ENTRY POINT: APSCS47V                                   @04A*/
*/*         PURPOSE:  THIS IS THE TABLE OF RESIDENT FONT CHARACTER    */
*/*                   SETS IN THE 4247 PRINTER. IT CAN BE CHANGED     */
*/*                   BY THE USER. THE TABLE IS TERMINATED BY A       */
*/*                   STRING OF CL8'FFFFFFFF'.                        */
*/*         LINKAGE:  NONE                                            */
*/*                                                                   */
*/*       ENTRY POINT: APSCP47V                                   @04A*/
*/*         PURPOSE:  THIS IS A TABLE OF RESIDENT CODE PAGES IN THE   */
*/*                   4247 PRINTER. IT CAN BE CHANGED BY THE USER.    */
*/*                   THE TABLE IS TERMINATED BY A STRING OF          */
*/*                   CL8'FFFFFFFF'.                                  */
*/*         LINKAGE:  NONE                                            */
*/*                                                                   */
*/*       ENTRY POINT: APSCS64X                                   @03A*/
*/*         PURPOSE:  THIS IS THE TABLE OF RESIDENT FONT CHARACTER    */
*/*                   SETS IN THE 64XX AND 65XX PRINTERS. IT CAN @08C*/
*/*                   BE CHANGED BY THE USER. THE TABLE IS TERMINATED*/
*/*                   BY A STRING OF CL8'FFFFFFFF'.                   */
*/*         LINKAGE:  NONE                                            */
*/*                                                                   */
*/*       ENTRY POINT: APSCP64X                                   @03A*/
*/*         PURPOSE:  THIS IS A TABLE OF RESIDENT CODE PAGES IN THE   */
*/*                   64XX AND 65XX PRINTERS. IT CAN BE CHANGED   @08C*/
*/*                   BY THE USER. THE TABLE IS TERMINATED BY A       */
*/*                   STRING OF CL8'FFFFFFFF'.                        */
*/*         LINKAGE:  NONE                                            */
*/*                                                                   */
```

Figure 148. APSRFTBL resident font resource table (Part 2 of 10)

```
*/*       TABLES:                                                    */
*/*         APSCS24V - 4224 & 4230 RESIDENT CHAR SET MAPPING TABLE   */
*/*         APSCP24V - 4224 & 4230 RESIDENT CODE PAGE MAPPING TABLE  */
*/*         APSCS34V - 4234 RESIDENT CHAR SET MAPPING TABLE          */
*/*         APSCP34V - 4234 RESIDENT CODE PAGE MAPPING TABLE         */
*/*         APSCS47V - 4247 RESIDENT CHAR SET MAPPING TABLE    @04A*/
*/*         APSCP47V - 4247 RESIDENT CODE PAGE MAPPING TABLE   @04A*/
*/*         APSCS64X - 6400/6408/6412 RESIDENT CHAR SET MAPPING TABLE */
*/*                                                             @03A*/
*/*         APSCP64X - 6400/6408/6412 RESIDENT CODE PAGE MAPPING  @03A*/
*/*                                                                  */
*/*                                                                  */
*/*       MESSAGES: NONE                                             */
*/*                                                                  */
*/*       CHANGE ACTIVITY                                            */
*/*                                                                  */
*/* $00=LAPS0005,HAF1237, 022690, B51PATM: RELEASE 2.1.0        @00A*/
*/* $H7=LAPS0005,HPRF102, 040690, B51PATM: REL 2.1.0 STAGE 2    @H7A*/
*/* $01=OY53959, HPRF102, 060292, BJ13KZB: INCORRECT FGID       @01A*/
*/* $02=OY56466, HPRF102, 080492, BJ13KZB: ADD 4230 CODE PAGES 875 & */
*/*                                        1026                 @02A*/
*/* $03=OW04104, HPRF220, 040694, BJ13KPC: 6408/6412 SUPPORT    @03A*/
*/* $04=OW19064, HPRF220, 960307, BDKUMLC: 4247 SUPPORT         @04A*/
*/* $DU=LAPS0007,HPRF310, 980604, BDKURLB: VERSION 3.1.0        @DUA*/
*/* $05=OW36740, HPRF310, 981028, BDKUEAS: Euro support for impact   */
*/*                                        printers             @05A*/
*/* $DX=LAPS0008,HPRF320,991117,BUQ4RLB: Version 3.2.0          @DXA*/
*/* $EV=LAPS0009,HPRF330,010530,BUQ4RLB: Version 3.3.0          @EVA*/
*/* $FA=LAPS0009,HPRF330,011017,BDKULMM: OCRB-KATAKANA font     @FAA*/
*/* $06=OW56680 ,HPRF330,021025,BDKULD : More 4247 support      @06A*/
*/* $FC=LAPS0010,HPRF340,020612,BDKULD : Version 3.4.0          @FCA*/
*/* $FM=LAPS0011,HPRF410,040220,BDKURDD: Version 4.1.0          @FMA*/
*/* $08=OA09521 ,HPRF340,041102,BDKURDD: Infoprint 6500 support @08A*/
*/* $FU=LAPS0012,HPRF420,060630,BDKURDD: Version 4.2.0          @FUA*/
*/* $GD=LAPS0013,HPRF430,080514,BDKURDD: Version 4.3.0          @GDA*/
*/* $GH=LAPS0014,HPRF440,090922,BDKURDD: Version 4.4.0          @GHA*/
*/* $HH=LAPS0015,HPRF450,111202,BDKUGLM: Version 4.5.0          @HHA*/
*/*                                                                  */
*/**********************END OF SPECIFICATIONS************************/
*/*****************************************************************/
*/*                                                                  */
*/*                     BAL CODE FOLLOWS                              */
*/*                                                                  */
*/*****************************************************************/
APSRFTBL CSECT                      ENTRY POINT OF THIS MODULE
APSRFTBL AMODE 31                   DO NOT REMOVE/CHANGE
APSRFTBL RMODE ANY                  DO NOT REMOVE/CHANGE
CS4224@  DC    V(APSCS24V)          DO NOT REMOVE/CHANGE
CP4224@  DC    V(APSCP24V)          DO NOT REMOVE/CHANGE
CS4234@  DC    V(APSCS34V)          DO NOT REMOVE/CHANGE
CP4234@  DC    V(APSCP34V)          DO NOT REMOVE/CHANGE
CS4247@  DC    V(APSCS47V)          DO NOT REMOVE/CHANGE        @04A*
CP4247@  DC    V(APSCP47V)          DO NOT REMOVE/CHANGE        @04A*
CS64XX@  DC    V(APSCS64X)          DO NOT REMOVE/CHANGE        @03A*
CP64XX@  DC    V(APSCP64X)          DO NOT REMOVE/CHANGE        @03A*
************************************************************************
* THE FOLLOWING IS THE CHARACTER SET TABLE FOR THE 4224 PRINTER. IT  *
* IS ALSO USED FOR THE 4230 AND 4247 PRINTERS WHEN RUNNING IN 4224   *
* EMULATION MODE. THE FIRST EIGHT (8) BYTES OF EACH ENTRY IS THE  @05A*
* NAME OF THE FONT CHARACTER SET. THE NEXT TWO (2) BYTE FIELD IS THE *
* GLOBAL ID OF CORRESPONDING RESIDENT FONT. THE NEXT TWO (2) BYTE    *
* FIELD DEFINES THE FONT ATTRIBUTES. THE FIRST BYTE IS RESERVED AND  *
* THE SECOND BYTE DEFINES WHETHER THE FONT IS ITALIC, DOUBLE STRIKE, *
* BOLD, OR DOUBLE WIDE.                                              *
*                                                                    *
* FOR MORE INFORMATION ON THE 4224 RESIDENT FONTS, SEE THE "IBM 4224 *
* PRINTER PRODUCT AND PROGRAMMING DESCRIPTION MANUAL (GC31-2551)."   *
************************************************************************
```

Figure 149. APSRFTBL resident font resource table (Part 3 of 10)

```
        ENTRY APSCS24V                         DO NOT REMOVE/CHANGE
APSCS24V DS    0D                              DO NOT REMOVE/CHANGE
        DC    CL8'C0L00BOA',XL2'0003',XL2'0000'  OCR-B
        DC    CL8'C0S0CR10',XL2'000B',XL2'0000'  COURIER 10
        DC    CL8'C0S0CB10',XL2'000B',XL2'0002'  COURIER BOLD 10
        DC    CL8'C0S0CI10',XL2'000B',XL2'0008'  COURIER ITALIC 10
        DC    CL8'C0L00AOA',XL2'0013',XL2'0000'  OCR-A
        DC    CL8'C0L0KATA',XL2'001A',XL2'0000'  KATAKANA
        DC    CL8'C0S0AE10',XL2'001A',XL2'0000'  APL 10
        DC    CL8'C0D0GT10',XL2'001A',XL2'0000'  GOTHIC TEXT 10
        DC    CL8'C0D0GB10',XL2'001A',XL2'0002'  GOTHIC BOLD 10
        DC    CL8'C0S0CR12',XL2'0055',XL2'0000'  COURIER 12
        DC    CL8'C0S0CB12',XL2'0055',XL2'0002'  COURIER BOLD 12
        DC    CL8'C0S0CI12',XL2'0055',XL2'0008'  COURIER ITALIC 12
        DC    CL8'C0D0GT12',XL2'0057',XL2'0000'  GOTHIC TEXT 12
        DC    CL8'C0D0GB12',XL2'0057',XL2'0002'  GOTHIC BOLD 12
        DC    CL8'C0D0GI12',XL2'0057',XL2'0008'  GOTHIC ITALIC 12
        DC    CL8'C0S0ESTR',XL2'00A0',XL2'0000'  ESSAY
        DC    CL8'C0S0EBTR',XL2'00A0',XL2'0002'  ESSAY BOLD
        DC    CL8'C0S0EITR',XL2'00A0',XL2'0008'  ESSAY ITALIC
        DC    CL8'C0D0GT15',XL2'00DE',XL2'0000'  GOTHIC TEXT 15
        DC    CL8'C0S0CR15',XL2'00DF',XL2'0000'  COURIER 15
        DC    CL8'C0S0CB15',XL2'00DF',XL2'0002'  COURIER BOLD 15
        DC    CL8'C0S0CI15',XL2'00DF',XL2'0008'  COURIER ITALIC 15
        DC    CL8'C0S0CD15',XL2'00DF',XL2'0001'  COURIER DBL-WIDE 15
        DC    CL8'C0S0CW15',XL2'00DF',XL2'0009'  COURIER DBL-WIDE ITAL
*************************************************************************
* THE FOLLOWING CHARACTER SETS ARE USED ONLY WITH THE 4247            *
* PRINTER WHEN RUN IN 4224 EMULATION MODE.                      @06A*
*************************************************************************
        DC    CL8'C0E0CR0R',XL2'000B',XL2'0000'  COURIER 10      @06A
        DC    CL8'C0520060',XL2'000B',XL2'0000'  COURIER 10      @06A
        DC    CL8'FFFFFFFF',XL2'FFFF',XL2'FFFF'  DO NOT REMOVE/CHANGE
*************************************************************************
* THE FOLLOWING IS THE CODE PAGE TABLE FOR THE 4224 PRINTER. IT IS    *
* ALSO USED FOR THE 4230 AND 4247 PRINTERS WHEN RUNNING IN 4224       *
* EMULATION MODE. THE FIRST EIGHT (8) BYTES OF EACH ENTRY IS THE  @05A*
* NAME OF THE CODE PAGE. THE NEXT TWO (2) BYTE FIELD IS THE GLOBAL    *
* ID OF THE CORRESPONDING RESIDENT CODE PAGE.                         *
*                                                                     *
* FOR MORE INFORMATION ON THE 4224 RESIDENT FONTS, SEE THE "IBM 4224  *
* PRINTER PRODUCT AND PROGRAMMING DESCRIPTION MANUAL (GC31-2551)."    *
*************************************************************************
        ENTRY APSCP24V                         DO NOT REMOVE/CHANGE
APSCP24V DS    0D                              DO NOT REMOVE/CHANGE
        DC    CL8'T1D0BASE',XL2'03EA'           CODE PAGE 1002
        DC    CL8'T1L038BA',XL2'03EA'           CODE PAGE 1002
        DC    CL8'T1V10037',XL2'0025'           CODE PAGE 37
        DC    CL8'T1V10273',XL2'0111'           CODE PAGE 273
        DC    CL8'T1V10274',XL2'0112'           CODE PAGE 274
        DC    CL8'T1V10275',XL2'0113'           CODE PAGE 275
        DC    CL8'T1V10277',XL2'0115'           CODE PAGE 277
        DC    CL8'T1V10278',XL2'0116'           CODE PAGE 278
        DC    CL8'T1V10280',XL2'0118'           CODE PAGE 280
        DC    CL8'T1V10281',XL2'0119'           CODE PAGE 281
        DC    CL8'T1V10282',XL2'011A'           CODE PAGE 282
        DC    CL8'T1V10284',XL2'011C'           CODE PAGE 284
        DC    CL8'T1V10285',XL2'011D'           CODE PAGE 285
        DC    CL8'T1000290',XL2'0122'           CODE PAGE 290
        DC    CL8'T1L02773',XL2'0122'           CODE PAGE 290 SUBSET
```

Figure 150. APSRFTBL resident font resource table (Part 4 of 10)

```
        DC    CL8'T1L02774',XL2'0122'          CODE PAGE 290 SUBSET
        DC    CL8'T1S0AE10',XL2'0125'          CODE PAGE 293
        DC    CL8'T1V10297',XL2'0129'          CODE PAGE 297
        DC    CL8'T1L00CR1',XL2'0154'          CODE PAGE 340
        DC    CL8'T1L00CRB',XL2'0154'          CODE PAGE 340
        DC    CL8'T1V10500',XL2'01F4'          CODE PAGE 500
        DC    CL8'T1001140',XL2'0474'          CODE PAGE 1140   @05A
        DC    CL8'T1001141',XL2'0475'          CODE PAGE 1141   @05A
        DC    CL8'T1001142',XL2'0476'          CODE PAGE 1142   @05A
        DC    CL8'T1001143',XL2'0477'          CODE PAGE 1143   @05A
        DC    CL8'T1001144',XL2'0478'          CODE PAGE 1144   @05A
        DC    CL8'T1001145',XL2'0479'          CODE PAGE 1145   @05A
        DC    CL8'T1001146',XL2'047A'          CODE PAGE 1146   @05A
        DC    CL8'T1001147',XL2'047B'          CODE PAGE 1147   @05A
        DC    CL8'T1001148',XL2'047C'          CODE PAGE 1148   @05A
        DC    CL8'T1001149',XL2'047D'          CODE PAGE 1149   @05A
***********************************************************************
* THE FOLLOWING CODE PAGES 875 AND 1026 ARE TO BE USED WITH THE    @02A*
* 4230 PRINTER ONLY.                                              @02A*
***********************************************************************
        DC    CL8'T1000875',XL2'036B'          CODE PAGE 875    @02A
        DC    CL8'T1001026',XL2'0402'          CODE PAGE 1026   @02A
***********************************************************************
* THE FOLLOWING CODE PAGES ARE USED ONLY WITH THE 4247            @06A*
* PRINTER WHEN RUN IN 4224 EMULATION MODE                        @05A*
***********************************************************************
        DC    CL8'T1000260',XL2'0104'          CODE PAGE 260    @06A
        DC    CL8'T1000286',XL2'011E'          CODE PAGE 286    @06A
        DC    CL8'T1000287',XL2'011F'          CODE PAGE 287    @06A
        DC    CL8'T1000288',XL2'0120'          CODE PAGE 288    @06A
        DC    CL8'T1000289',XL2'0121'          CODE PAGE 289    @06A
        DC    CL8'T1000437',XL2'01B5'          CODE PAGE 437    @06A
        DC    CL8'T1V10282',XL2'033F'          CODE PAGE 831    @06A
        DC    CL8'T1H00833',XL2'0341'          CODE PAGE 833    @06A
        DC    CL8'T1000890',XL2'037A'          CODE PAGE 890    @06A
        DC    CL8'T1L00924',XL2'039C'          CODE PAGE 924    @05A
        DC    CL8'T1001112',XL2'0458'          CODE PAGE 1112   @06A
        DC    CL8'T1001122',XL2'0462'          CODE PAGE 1122   @06A
        DC    CL8'FFFFFFFF',XL2'FFFF'          DO NOT REMOVE/CHANGE
***********************************************************************
* THE FOLLOWING IS THE CHARACTER SET TABLE FOR THE 4234 PRINTER. IT   *
* IS ALSO USED FOR THE 64XX AND 65XX PRINTERS WHEN RUNNING IN 4234@08C*
* EMULATION MODE. THE FIRST EIGHT (8) BYTES OF EACH ENTRY IS THE  @05A*
* NAME OF THE FONT CHARACTER SET. THE NEXT TWO (2) BYTE FIELD IS THE  *
* GLOBAL ID OF CORRESPONDING RESIDENT FONT. THE NEXT TWO (2) BYTE     *
* FIELD DEFINES THE FONT ATTRIBUTES. THE FIRST BYTE IS RESERVED AND   *
* THE SECOND BYTE DEFINES WHETHER THE FONT IS ITALIC, DOUBLE STRIKE,  *
* BOLD, OR DOUBLE WIDE.                                               *
*                                                                    *
* FOR MORE INFORMATION ON THE 4234 RESIDENT FONTS, SEE THE "IBM 4234  *
* PRINTER PRODUCT AND PROGRAMMING DESCRIPTION MANUAL (GC31-2554)."    *
***********************************************************************
```

Figure 151. APSRFTBL resident font resource table (Part 5 of 10)

```
        ENTRY APSCS34V                          DO NOT REMOVE/CHANGE
APSCS34V DS    0D                               DO NOT REMOVE/CHANGE
        DC    CL8'C0L00BOA',XL2'0003',XL2'0000'  OCR-B
        DC    CL8'C0S0CR10',XL2'000B',XL2'0000'  COURIER 10
        DC    CL8'C0S0CB10',XL2'000B',XL2'0002'  COURIER BOLD 10
        DC    CL8'C0S0CI10',XL2'000B',XL2'0008'  COURIER ITALIC 10
        DC    CL8'C0L00AOA',XL2'0013',XL2'0000'  OCR-A
        DC    CL8'C0L0KATA',XL2'001A',XL2'0000'  KATAKANA
        DC    CL8'C0S0AE10',XL2'001A',XL2'0000'  APL 10
        DC    CL8'C0D0GT10',XL2'001A',XL2'0000'  GOTHIC TEXT 10
        DC    CL8'C0D0GB10',XL2'001A',XL2'0002'  GOTHIC BOLD 10
        DC    CL8'C0S0CR12',XL2'0055',XL2'0000'  COURIER 12
        DC    CL8'C0S0CB12',XL2'0055',XL2'0002'  COURIER BOLD 12
        DC    CL8'C0S0CI12',XL2'0055',XL2'0008'  COURIER ITALIC 12
        DC    CL8'C0D0GT12',XL2'0057',XL2'0000'  GOTHIC TEXT 12
        DC    CL8'C0D0GB12',XL2'0057',XL2'0002'  GOTHIC BOLD 12
        DC    CL8'C0D0GI12',XL2'0057',XL2'0008'  GOTHIC ITALIC 12
        DC    CL8'C0S0ESTR',XL2'00A0',XL2'0000'  ESSAY
        DC    CL8'C0S0EBTR',XL2'00A0',XL2'0002'  ESSAY BOLD
        DC    CL8'C0S0EITR',XL2'00A0',XL2'0008'  ESSAY ITALIC
        DC    CL8'C0S0D224',XL2'00CC',XL2'0000'  GOTHIC TEXT 13   @01C
        DC    CL8'C0S0D225',XL2'00CC',XL2'0000'  GOTHIC TEXT 13   @01C
        DC    CL8'C0S0D226',XL2'00CC',XL2'0002'  GOTHIC BOLD 13   @01C
        DC    CL8'C0S0D227',XL2'00CC',XL2'0008'  GOTHIC ITAL 13   @01C
        DC    CL8'C0D0GT15',XL2'00DE',XL2'0000'  GOTHIC TEXT 15
        DC    CL8'C0S0CR15',XL2'00DF',XL2'0000'  COURIER 15
        DC    CL8'C0S0CB15',XL2'00DF',XL2'0002'  COURIER BOLD 15
        DC    CL8'C0S0CI15',XL2'00DF',XL2'0008'  COURIER ITALIC 15
        DC    CL8'C0S0CD15',XL2'00DF',XL2'0001'  COURIER DBL-WIDE 15
        DC    CL8'C0S0CW15',XL2'00DF',XL2'0009'  COURIER DBL-WIDE ITAL
        DC    CL8'FFFFFFFF',XL2'FFFF',XL2'FFFF'  DO NOT REMOVE/CHANGE
*************************************************************************
* THE FOLLOWING IS THE CODE PAGE TABLE FOR THE 4234 PRINTER. IT IS    *
* ALSO USED FOR THE 64XX AND 65XX PRINTERS WHEN RUNNING IN 4234   @08C*
* EMULATION MODE. THE FIRST EIGHT (8) BYTES OF EACH ENTRY IS THE NAME *
* OF THE COD PAGE. THE NEXT TWO (2) BYTE FIELD IS THE GLOBAL ID   @05A*
* OF THE CORRESPONDING RESIDENT CODE PAGE.                           *
*                                                                    *
* FOR MORE INFORMATION ON THE 4234 RESIDENT FONTS, SEE THE "IBM 4234 *
* PRINTER PRODUCT AND PROGRAMMING DESCRIPTION MANUAL (GC31-2554)."   *
*************************************************************************
        ENTRY APSCP34V                          DO NOT REMOVE/CHANGE
APSCP34V DS    0D                               DO NOT REMOVE/CHANGE
        DC    CL8'T1D0BASE',XL2'03EA'            CODE PAGE 1002
        DC    CL8'T1L038BA',XL2'03EA'            CODE PAGE 1002
        DC    CL8'T1V10037',XL2'0025'            CODE PAGE 37
        DC    CL8'T1V10273',XL2'0111'            CODE PAGE 273
        DC    CL8'T1V10274',XL2'0112'            CODE PAGE 274
        DC    CL8'T1V10275',XL2'0113'            CODE PAGE 275
        DC    CL8'T1V10277',XL2'0115'            CODE PAGE 277
        DC    CL8'T1V10278',XL2'0116'            CODE PAGE 278
        DC    CL8'T1V10280',XL2'0118'            CODE PAGE 280
        DC    CL8'T1V10281',XL2'0119'            CODE PAGE 281
        DC    CL8'T1V10282',XL2'011A'            CODE PAGE 282
        DC    CL8'T1V10284',XL2'011C'            CODE PAGE 284
        DC    CL8'T1V10285',XL2'011D'            CODE PAGE 285
        DC    CL8'T1000290',XL2'0122'            CODE PAGE 290
        DC    CL8'T1L02773',XL2'0122'            CODE PAGE 290 SUBSET
        DC    CL8'T1L02774',XL2'0122'            CODE PAGE 290 SUBSET
        DC    CL8'T1S0AE10',XL2'0125'            CODE PAGE 293
        DC    CL8'T1V10297',XL2'0129'            CODE PAGE 297
        DC    CL8'T1000310',XL2'0136'            CODE PAGE 310
        DC    CL8'T1000420',XL2'01A4'            CODE PAGE 420
        DC    CL8'T1000423',XL2'01A7'            CODE PAGE 423
        DC    CL8'T1000424',XL2'01A8'            CODE PAGE 424
        DC    CL8'T1V10500',XL2'01F4'            CODE PAGE 500
        DC    CL8'T1000838',XL2'0346'            CODE PAGE 838
        DC    CL8'T1000870',XL2'0366'            CODE PAGE 870
        DC    CL8'T1V10871',XL2'0367'            CODE PAGE 871
        DC    CL8'T1L00OCR1',XL2'037C'           CODE PAGE 892
        DC    CL8'T1L00OCRB',XL2'037D'           CODE PAGE 893
```

Figure 152. APSRFTBL resident font resource table (Part 6 of 10)

```
***********************************************************************
* THE FOLLOWING CODE PAGES ARE TO BE USED ONLY WITH THE 64XX    @05A*
* PRINTERS WHEN RUN IN 4234 EMULATION MODE.                     @05A*
* *********************************************************************
         DC    CL8'T1L00924',XL2'039C'            CODE PAGE 924    @05A
         DC    CL8'T1001140',XL2'0474'            CODE PAGE 1140   @05A
         DC    CL8'T1001141',XL2'0475'            CODE PAGE 1141   @05A
         DC    CL8'T1001142',XL2'0476'            CODE PAGE 1142   @05A
         DC    CL8'T1001143',XL2'0477'            CODE PAGE 1143   @05A
         DC    CL8'T1001144',XL2'0478'            CODE PAGE 1144   @05A
         DC    CL8'T1001145',XL2'0479'            CODE PAGE 1145   @05A
         DC    CL8'T1001146',XL2'047A'            CODE PAGE 1146   @05A
         DC    CL8'T1001147',XL2'047B'            CODE PAGE 1147   @05A
         DC    CL8'T1001148',XL2'047C'            CODE PAGE 1148   @05A
         DC    CL8'T1001149',XL2'047D'            CODE PAGE 1149   @05A
         DC    CL8'FFFFFFFF',XL2'FFFF'            DO NOT REMOVE/CHANGE
***********************************************************************
* THE FOLLOWING IS THE CHARACTER SET TABLE FOR THE 4247 PRINTER. THE  *
* FIRST EIGHT (8) BYTES OF EACH ENTRY IS THE NAME OF THE FONT CHARAC- *
* TER SET. THE NEXT TWO (2) BYTE FIELD IS THE GLOBAL ID OF CORRESPON- *
* DING RESIDENT FONT. THE NEXT TWO (2) BYTE FIELD DEFINES THE FONT    *
* ATTRIBUTES. THE FIRST BYTE IS RESERVED AND THE SECOND BYTE DEFINES  *
* WHETHER THE FONT IS ITALIC, DOUBLE STRIKE, BOLD, OR DOUBLE WIDE.@04P*
***********************************************************************
         ENTRY APSCS47V                           DO NOT REMOVE/CHANGE
APSCS47V DS 0D                                     DO NOT REMOVE/CHANGE
         DC CL8'C0L00BOA',XL2'0003',XL2'0000'     OCR-B              @04P
         DC CL8'C0S0CR10',XL2'000B',XL2'0000'     COURIER 10         @04P
         DC CL8'C0E0CR0R',XL2'000B',XL2'0000'     COURIER 10         @06A
         DC CL8'C0520060',XL2'000B',XL2'0000'     COURIER 10         @06A
         DC CL8'C0S0CB10',XL2'000B',XL2'0002'     COURIER BOLD 10    @04P
         DC CL8'C0S0CI10',XL2'000B',XL2'0008'     COURIER ITAL 10    @04P
         DC CL8'C0L00AOA',XL2'0013',XL2'0000'     OCR-A              @04P
         DC CL8'C0L0KATA',XL2'001A',XL2'0000'     KATAKANA           @04P
         DC CL8'C0S0AE10',XL2'001A',XL2'0000'     APL 10             @04P
         DC CL8'C0D0GT10',XL2'001A',XL2'0000'     GOTHIC TEXT 10     @04P
         DC CL8'C0D0GB10',XL2'001A',XL2'0002'     GOTHIC BOLD 10     @04P
         DC CL8'C0S0CR12',XL2'0055',XL2'0000'     COURIER 12         @04P
         DC CL8'C0S0CB12',XL2'0055',XL2'0002'     COURIER BOLD 12    @04P
         DC CL8'C0S0CI12',XL2'0055',XL2'0008'     COURIER ITAL 12    @04P
         DC CL8'C0D0GT12',XL2'0057',XL2'0000'     GOTHIC TEXT 12     @04P
         DC CL8'C0D0GB12',XL2'0057',XL2'0002'     GOTHIC BOLD 12     @04P
         DC CL8'C0D0GI12',XL2'0057',XL2'0008'     GOTHIC ITAL 12     @04P
         DC CL8'C0S0ESTR',XL2'00A0',XL2'0000'     ESSAY              @04P
         DC CL8'C0S0EBTR',XL2'00A0',XL2'0002'     ESSAY BOLD         @04P
         DC CL8'C0S0EITR',XL2'00A0',XL2'0008'     ESSAY ITALIC       @04P
         DC CL8'C0D0GT15',XL2'00DE',XL2'0000'     GOTHIC TEXT 15     @04P
         DC CL8'C0S0CR15',XL2'00DF',XL2'0000'     COURIER 15         @04P
         DC CL8'C0S0CB15',XL2'00DF',XL2'0002'     COURIER BOLD 15    @04P
         DC CL8'C0S0CI15',XL2'00DF',XL2'0008'     COURIER ITAL 15    @04P
         DC CL8'C0S0CD15',XL2'00DF',XL2'0001'     COUR DBL-WID 15    @04P
         DC CL8'C0S0CW15',XL2'00DF',XL2'0009'     COUR DBL-WID ITAL@04P
         DC CL8'FFFFFFFF',XL2'FFFF',XL2'FFFF'     DO NOT REMOVE/CHANGE
```

Figure 153. APSRFTBL resident font resource table (Part 7 of 10)

```
        ************************************************************************
        * THE FOLLOWING IS THE CODE PAGE TABLE FOR THE 4247 PRINTER. THE FIRST*
        *EIGHT (8) BYTES OF EACH ENTRY IS THE NAME OF THE  CODE PAGE. THE     *
        *NEXT TWO (2) BYTE FIELD IS THE GLOBAL ID OF THE CORRESPONDING        *
        *RESIDENT PAGE.                                                  @04P*
        ************************************************************************
                ENTRY APSCP47V                        DO NOT REMOVE/CHANGE
        APSCP47V DS   0D                               DO NOT REMOVE/CHANGE
                DC    CL8'T1D0BASE',XL2'03EA'          CODE PAGE 1002    @04P
                DC    CL8'T1L038BA',XL2'03EA'          CODE PAGE 1002    @04P
                DC    CL8'T1GDP256',XL2'0100'          CODE PAGE 256     @04A
                DC    CL8'T1V10037',XL2'0025'          CODE PAGE 37      @04P
                DC    CL8'T1000260',XL2'0104'          CODE PAGE 260     @06A
                DC    CL8'T1V10273',XL2'0111'          CODE PAGE 273     @04P
                DC    CL8'T1V10274',XL2'0112'          CODE PAGE 274     @04P
                DC    CL8'T1V10275',XL2'0113'          CODE PAGE 275     @04P
                DC    CL8'T1V10277',XL2'0115'          CODE PAGE 277     @04P
                DC    CL8'T1V10278',XL2'0116'          CODE PAGE 278     @04P
                DC    CL8'T1V10280',XL2'0118'          CODE PAGE 280     @04P
                DC    CL8'T1V10281',XL2'0119'          CODE PAGE 281     @04P
                DC    CL8'T1V10282',XL2'011A'          CODE PAGE 282     @04P
                DC    CL8'T1V10284',XL2'011C'          CODE PAGE 284     @04P
                DC    CL8'T1V10285',XL2'011D'          CODE PAGE 285     @04P
                DC    CL8'T1000286',XL2'011E'          CODE PAGE 286     @06A
                DC    CL8'T1000287',XL2'011F'          CODE PAGE 287     @06A
                DC    CL8'T1000288',XL2'0120'          CODE PAGE 288     @06A
                DC    CL8'T1000289',XL2'0121'          CODE PAGE 289     @06A
                DC    CL8'T1000290',XL2'0122'          CODE PAGE 290     @04P
                DC    CL8'T1L02773',XL2'0122'          CODE PAGE 290 SUB@04P
                DC    CL8'T1L02774',XL2'0122'          CODE PAGE 290 SUB@04P
                DC    CL8'T1S0AE10',XL2'0125'          CODE PAGE 293     @04P
                DC    CL8'T1V10297',XL2'0129'          CODE PAGE 297     @04P
                DC    CL8'T1000310',XL2'0136'          CODE PAGE 310     @04P
                DC    CL8'T1L0OCR1',XL2'0154'          CODE PAGE 340     @06A
                DC    CL8'T1L0OCRB',XL2'0154'          CODE PAGE 340     @06A
                DC    CL8'T1000420',XL2'01A4'          CODE PAGE 420     @04P
                DC    CL8'T1000423',XL2'01A7'          CODE PAGE 423     @04P
                DC    CL8'T1000424',XL2'10A8'          CODE PAGE 424     @04P
                DC    CL8'T1000437',XL2'01B5'          CODE PAGE 437     @06A
                DC    CL8'T1V10500',XL2'01F4'          CODE PAGE 500     @04A
                DC    CL8'T1000803',XL2'0323'          CODE PAGE 803     @04A
                DC    CL8'T1V10282',XL2'033F'          CODE PAGE 831     @06A
                DC    CL8'T1H00833',XL2'0341'     ETHERNET ONLY CP 833   @06A
                DC    CL8'T1000838',XL2'0346'          CODE PAGE 838     @04P
                DC    CL8'T1000870',XL2'0366'          CODE PAGE 870     @04P
                DC    CL8'T1V10871',XL2'0367'          CODE PAGE 871     @04P
                DC    CL8'T1000875',XL2'036B'          CODE PAGE 875     @04A
                DC    CL8'T1000880',XL2'0370'          CODE PAGE 880     @04A
                DC    CL8'T1000890',XL2'037A'          CODE PAGE 890     @06A
                DC    CL8'T1000892',XL2'037C'          CODE PAGE 892     @04P
                DC    CL8'T1L0OCR1',XL2'037C'          CODE PAGE 892     @04P
                DC    CL8'T1000893',XL2'037D'          CODE PAGE 893     @04P
                DC    CL8'T1L0OCRB',XL2'037D'          CODE PAGE 893     @04P
                DC    CL8'T1L00924',XL2'039C'          CODE PAGE 924     @05A
                DC    CL8'T1001025',XL2'0401'          CODE PAGE 1025    @04A
                DC    CL8'T1001026',XL2'0402'          CODE PAGE 1026    @04A
                DC    CL8'T1001112',XL2'0458'     ETHERNET ONLY CP 1112  @06A
                DC    CL8'T1001122',XL2'0462'     ETHERNET ONLY CP 1122  @06A
                DC    CL8'T1001140',XL2'0474'          CODE PAGE 1140    @05A
                DC    CL8'T1001141',XL2'0475'          CODE PAGE 1141    @05A
                DC    CL8'T1001142',XL2'0476'          CODE PAGE 1142    @05A
                DC    CL8'T1001143',XL2'0477'          CODE PAGE 1143    @05A
                DC    CL8'T1001144',XL2'0478'          CODE PAGE 1144    @05A
                DC    CL8'T1001145',XL2'0479'          CODE PAGE 1145    @05A
                DC    CL8'T1001146',XL2'047A'          CODE PAGE 1146    @05A
                DC    CL8'T1001147',XL2'047B'          CODE PAGE 1147    @05A
                DC    CL8'T1001148',XL2'047C'          CODE PAGE 1148    @05A
                DC    CL8'T1001149',XL2'047D'          CODE PAGE 1149    @05A
                DC    CL8'FFFFFFFF',XL2'FFFF'          DO NOT REMOVE/CHANGE
```

Figure 154. APSRFTBL resident font resource table (Part 8 of 10)

```
        *********************************************************************
        * THE FOLLOWING IS THE CHARACTER SET TABLE FOR THE 64XX PRINTER. IT   *
        * IS ALSO USED FOR THE 65XX PRINTERS WHEN RUNNING IN 6408 EMULATION   *
        * MODE. IF THE 64XX OR 65XX PRINTER IS RUNNING IN 4234 EMULATION  MODE*
        * USE THE APSCS34X TABLE. THE FIRST EIGHT(8) BYTES OF EACH ENTRY  @08A*
        * IS THE NAME OF THE FONT CHARACTER SET. THE NEXT TWO (2) BYTE FIELD  *
        * IS THE GLOBAL ID OF THE CORRESPONDING RESIDENT FONT. THE NEXT TWO   *
        * (2) BYTE FIELD DEFINES THE FONT ATTRIBUTES. THE FIRST BYTE IS       *
        * RESERVED AND THE SECOND BYTE DEFINES WHETHER THE FONT IS ITALIC,    *
        * DOUBLE STRIKE, BOLD, OR DOUBLE WIDE.                          @03C*
        *********************************************************************
                ENTRY APSCS64X                        DO NOT REMOVE/CHANGE
        APSCS64X DS   0D                               DO NOT REMOVE/CHANGE
                DC    CL8'C0L00BOA',XL2'0003',XL2'0000'  OCR-B          @03P
                DC    CL8'C0L0KBOA',XL2'020A',XL2'0000'  OCR-B/KATAKANA @FAA
                DC    CL8'C0S0CR10',XL2'000B',XL2'0000'  COURIER 10     @03P
                DC    CL8'C0S0CB10',XL2'002E',XL2'0000'  COURIER BOLD 10 @03P
                DC    CL8'C0S0CI10',XL2'0012',XL2'0000'  COURIER ITAL 10 @03P
                DC    CL8'C0L00AOA',XL2'0013',XL2'0000'  OCR-A          @03P
                DC    CL8'C0L0KATA',XL2'001A',XL2'0000'  KATAKANA       @03P
                DC    CL8'C0S0AE10',XL2'001A',XL2'0000'  APL 10         @03P
                DC    CL8'C0D0GT10',XL2'001A',XL2'0000'  GOTHIC TEXT 10 @03P
                DC    CL8'C0D0GB10',XL2'001A',XL2'0002'  GOTHIC BOLD 10 @03P
                DC    CL8'C0S0CR12',XL2'0055',XL2'0000'  COURIER 12     @03P
                DC    CL8'C0S0CB12',XL2'006C',XL2'0000'  COURIER BOLD 12 @03P
                DC    CL8'C0S0LB12',XL2'006E',XL2'0000'  LET GOTH BOLD 12 @03A
                DC    CL8'C0S0CI12',XL2'0055',XL2'0008'  COURIER ITAL 12 @03P
                DC    CL8'C0D0GT12',XL2'0057',XL2'0000'  GOTHIC TEXT 12 @03P
                DC    CL8'C0D0GB12',XL2'0057',XL2'0002'  GOTHIC BOLD 12 @03P
                DC    CL8'C0D0GI12',XL2'0057',XL2'0008'  GOTHIC ITALIC 12 @03P
                DC    CL8'C0S0ESTR',XL2'00A0',XL2'0000'  ESSAY          @03P
                DC    CL8'C0S0EBTR',XL2'00A3',XL2'0000'  ESSAY BOLD     @03P
                DC    CL8'C0S0EITR',XL2'00A2',XL2'0000'  ESSAY ITALIC   @03P
                DC    CL8'C0D0GT13',XL2'00CC',XL2'0000'  GOTHIC TEXT 13 @03A
                DC    CL8'C0S0D224',XL2'00CC',XL2'0000'  GOTHIC TEXT 13 @03P
                DC    CL8'C0S0D225',XL2'00CC',XL2'0000'  GOTHIC TEXT 13 @03P
                DC    CL8'C0S0D226',XL2'00CC',XL2'0002'  GOTHIC BOLD 13 @03P
                DC    CL8'C0S0D227',XL2'00CC',XL2'0008'  GOTHIC ITAL 13 @03P
                DC    CL8'C0D0GT15',XL2'00DE',XL2'0000'  GOTHIC TEXT 15 @03P
                DC    CL8'C0S0CR15',XL2'00DF',XL2'0000'  COURIER 15     @03P
                DC    CL8'C0S0CB15',XL2'00DF',XL2'0002'  COURIER BOLD 15 @03P
                DC    CL8'C0S0CI15',XL2'00DF',XL2'0008'  COURIER ITAL 15 @03P
                DC    CL8'C0S0CD15',XL2'00DF',XL2'0001'  COUR DBL-WID 15 @03P
                DC    CL8'C0S0CW15',XL2'00DF',XL2'0009'  COUR DBL-WID ITAL@03P
                DC    CL8'C0D0GT18',XL2'0102',XL2'0000'  GOTHIC TEXT 18 @03A
                DC    CL8'FFFFFFFF',XL2'FFFF',XL2'FFFF'  DO NOT REMOVE/CHANGE
```

Figure 155. APSRFTBL resident font resource table (Part 9 of 10)

```
        ********************************************************************
        * THE FOLLOWING IS THE CODE PAGE TABLE FOR THE 64XX PRINTER. IT IS   *
        * ALSO USED FOR THE 65XX PRINTERS WHEN RUNNING IN 6408 EMULATION     *
        * MODE. IF THE 64XX OR 65XX PRINTER IS RUNNING IN 4234 EMULATION MODE *
        * USE THE APSCS34X TABLE. THE FIRST EIGHT(8) BYTES OF EACH ENTRY  @08A*
        * IS THE NAME OF THE CODE PAGE. THE NEXT TWO (2) BYTE FIELD IS THE    *
        * GLOBAL ID OF THE CORRESPONDING RESIDENT CODE PAGE.             @03C*
        ********************************************************************
                ENTRY APSCP64X                           DO NOT REMOVE/CHANGE
APSCP64X DS     0D                                       DO NOT REMOVE/CHANGE
                DC    CL8'T1D0BASE',XL2'03EA'             CODE PAGE 1002    @03P
                DC    CL8'T1L038BA',XL2'03EA'             CODE PAGE 1002    @03P
                DC    CL8'T1GDP256',XL2'0100'             CODE PAGE 256     @03A
                DC    CL8'T1V10037',XL2'0025'             CODE PAGE 37      @03P
                DC    CL8'T1V10273',XL2'0111'             CODE PAGE 273     @03P
                DC    CL8'T1V10274',XL2'0112'             CODE PAGE 274     @03P
                DC    CL8'T1V10275',XL2'0113'             CODE PAGE 275     @03P
                DC    CL8'T1V10277',XL2'0115'             CODE PAGE 277     @03P
                DC    CL8'T1V10278',XL2'0116'             CODE PAGE 278     @03P
                DC    CL8'T1V10280',XL2'0118'             CODE PAGE 280     @03P
                DC    CL8'T1V10281',XL2'0119'             CODE PAGE 281     @03P
                DC    CL8'T1V10282',XL2'011A'             CODE PAGE 282     @03P
                DC    CL8'T1V10284',XL2'011C'             CODE PAGE 284     @03P
                DC    CL8'T1V10285',XL2'011D'             CODE PAGE 285     @03P
                DC    CL8'T1000290',XL2'0122'             CODE PAGE 290     @03P
                DC    CL8'T1L02773',XL2'0122'             CODE PAGE 290 SUB@03P
                DC    CL8'T1L02774',XL2'0122'             CODE PAGE 290 SUB@03P
                DC    CL8'T1S0AE10',XL2'0125'             CODE PAGE 293     @03P
                DC    CL8'T1V10297',XL2'0129'             CODE PAGE 297     @03P
                DC    CL8'T1000310',XL2'0136'             CODE PAGE 310     @03P
                DC    CL8'T1000420',XL2'01A4'             CODE PAGE 420     @03P
                DC    CL8'T1000423',XL2'01A7'             CODE PAGE 423     @03P
                DC    CL8'T1000424',XL2'01A8'             CODE PAGE 424     @03P
                DC    CL8'T1V10500',XL2'01F4'             CODE PAGE 500     @03P
                DC    CL8'T1000803',XL2'0323'             CODE PAGE 803     @03A
                DC    CL8'T1000838',XL2'0346'             CODE PAGE 838     @03P
                DC    CL8'T1000870',XL2'0366'             CODE PAGE 870     @03P
                DC    CL8'T1V10871',XL2'0367'             CODE PAGE 871     @03P
                DC    CL8'T1000875',XL2'036B'             CODE PAGE 875     @03A
                DC    CL8'T1000880',XL2'0370'             CODE PAGE 880     @03A
                DC    CL8'T1000892',XL2'037C'             CODE PAGE 892     @03P
                DC    CL8'T1L0OCR1',XL2'037C'             CODE PAGE 892     @03P
                DC    CL8'T1000893',XL2'037D'             CODE PAGE 893     @03A
                DC    CL8'T1L0OCRB',XL2'037D'             CODE PAGE 893     @03P
                DC    CL8'T1000905',XL2'0389'             CODE PAGE 905     @03A
                DC    CL8'T1L00924',XL2'039C'             CODE PAGE 924     @05A
                DC    CL8'T1001025',XL2'0401'             CODE PAGE 1025    @03A
                DC    CL8'T1001026',XL2'0402'             CODE PAGE 1026    @03A
                DC    CL8'T1001140',XL2'0474'             CODE PAGE 1140    @05A
                DC    CL8'T1001141',XL2'0475'             CODE PAGE 1141    @05A
                DC    CL8'T1001142',XL2'0476'             CODE PAGE 1142    @05A
                DC    CL8'T1001143',XL2'0477'             CODE PAGE 1143    @05A
                DC    CL8'T1001144',XL2'0478'             CODE PAGE 1144    @05A
                DC    CL8'T1001145',XL2'0479'             CODE PAGE 1145    @05A
                DC    CL8'T1001146',XL2'047A'             CODE PAGE 1146    @05A
                DC    CL8'T1001147',XL2'047B'             CODE PAGE 1147    @05A
                DC    CL8'T1001148',XL2'047C'             CODE PAGE 1148    @05A
                DC    CL8'T1001149',XL2'047D'             CODE PAGE 1149    @05A
                DC    CL8'FFFFFFFF',XL2'FFFF'             DO NOT REMOVE/CHANGE
                END   APSRFTBL                            DO NOT REMOVE/CHANGE
```

Figure 156. APSRFTBL resident font resource table (Part 10 of 10)

Appendix F. Field descriptions for Exit 7

This information describes Exit 7 parameter areas and fields, which contain information for intended Programming Interfaces. The information presents these:

- Descriptions of parameter areas
- A control block listing for each parameter area
- A description of each field in each parameter area
- A processing indicator cross-reference, which you can use to match fields with exit calls

XTP7 parameter area (DSECT)

XTP7, a DSECT (dummy section) in APSGEXTP, contains information that Exit 7 needs to determine what action to take. The XTPRECP pointer in APSGEXTP provides the address of XTP7. XTP7 is mapped within macro APSGEXTP. For more information about APSGEXTP, see Figure 87 on page 266–Figure 89 on page 268.

All of the fields in XTP7 can be read by the exit; however, most of those fields are not intended to be written to by the exit, and some fields do not always contain information about the exit call in progress. PSF looks for changes to specific fields only after certain calls to the exit are made.

"XTP7 parameter area format" on page 421 shows the format of XTP7.

XTP7 parameter area format

```
Offsets   Type      Length  Name         Description

0    (0) CHARACTER   233  XTP7           EXIT 7 INPUT RECORD
0    (0) CHARACTER     4  XTP7CID        CONTROL BLOCK ID = 'XTP7'
4    (4) SIGNED        4  XTP7LENG       LENGTH OF XTP7 AREA
8    (8) BITSTRING     2  XTP7ETYP       FLAGS SET BY PSF-INDICATES
                                         TO EXIT 7 THE TYPE OF EXIT
                                         CALL BEING MADE.
     1... ....            XTP7INIT       INITIALIZATION EXIT
     .1.. ....            XTP7BDS        BEGINNING OF DATA SET
     ..1. ....            XTP7ACC        RESOURCE ACCESS TIME EXIT
     ...1 ....            XTP7LDB        LOAD BEGIN EXIT
     .... 1...            XTP7LDE        LOAD END EXIT
     .... .1..            XTP7DSE        RESOURCE DELETE CALL
     .... ..1.            XTP7TERM       TERMINATION OF FSA
     .... ...1            XTP7EDS        END OF DATA SET
     1111 1111            *              RESERVED
10   (A) BITSTRING     2  XTP7RTYP       FLAGS SET BY PSF-INDICATES TO
                                         EXIT 7 THE TYPE OF RESOURCE
                                         CURRENTLY BEING PROCESSED
     1... ....            XTP7PD         PAGEDEF
     .1.. ....            XTP7FD         FORMDEF
     ..1. ....            XTP7CF         CODED FONT
     ...1 ....            XTP7MO         OVERLAY
     .... 1...            XTP7PS         PAGE SEGMENT
     .... .1..            XTP7OC         OBJECT CONTAINER
     .... ..11            *              RESERVED
     1111 1111            *              RESERVED
12   (C) BITSTRING     1  XTP7DSTY       FLAGS SET BY PSF-INDICATES
                                         TO EXIT 7 THE TYPE OF DATA
                                         SET CURRENTLY BEING PROCESSED
     1... ....            XTP7AUX        PROCESSING SYSTEM DEFAULT OR
                                         FOR AUXILIARY DATA SETS
     .1.. ....            XTP7PDS        PRINT DATA SET
     ..11 1111            *              RESERVED
```

Figure 157. XTP7 parameter area format (Part 1 of 9)

```
13   (D) BITSTRING     1  XTP7DSAT    FLAGS SET BY PSF-TYPE OF THE
                                      DATA SET BEING PROCESSED
        1... ....         XTP7PDFT    SYSTEM DEFAULT
        .1.. ....         XTP7PJHD    JOB HEADER
        ..1. ....         XTP7PJTR    JOB TRAILER
        ...1 ....         XTP7PDSH    DATA SET HEADER
        .... 1...         XTP7PMDS    MESSAGE DATA SET
        .... .1..         XTP7PLDS    LAST PRINT DATA SET IN JOB
                                      VALID IN BDS AND DSE EXITS ONLY
        .... ..11         *           RESERVED
14   (E) UNSIGNED      1  XTP7ACSE    AUTOMATIC CUT-SHEET EMULATION
                                      CHECKING. VALID FOR
                                      INITIALIZATION 'XTP7INIT', BDSC
                                      'XTP7BDS', AND EDSC 'XTP7EDS'.
15   (F) UNSIGNED      1  XTP7IPCC    NUMBER OF COPIES OF PAGES TO
                                      PRODUCE.
                                      - VALID ONLY WITH
                                      THE INTERRUPT MESSAGE XTP7IPMF.
16   (10) BITSTRING    4  XTP7NOTY    NOTIFICATION FLAGS SET BY
                                      EXIT 7, INDICATES TO PSF WHEN
                                      THE EXIT WANTS CONTROL
16   (10) BITSTRING    1  XTP7MISC    MISCELLANEOUS TYPE EXITS
        1... ....         XTP7EBDS    BEGINNING OF DATA SET
        .1.. ....         XTP7ETRM    TERMINATION OF FSA
        ..1. ....         XTP7EEDS    END OF DATA SET
        ...1 1111         *           RESERVED
17   (11) BITSTRING    1  XTP7NACC    ACCESS TIME EXITS
        1... ....         XTP7APD     PAGEDEFS
        .1.. ....         XTP7AFD     FORMDEFS
        ..1. ....         XTP7ACF     CODED FONTS
        ...1 ....         XTP7AMO     MEDIUM OVERLAYS
        .... 1...         XTP7APS     PAGE SEGMENTS
        .... .1..         XTP7AOC     OBJECT CONTAINERS
        .... ..11         *           RESERVED
18   (12) BITSTRING    1  XTP7NLD     LOAD TIME EXITS
        1... ....         XTP7LPD     PAGEDEFS
        .1.. ....         XTP7LFD     FORMDEFS
        ..1. ....         XTP7LCF     CODED FONTS
        ...1 ....         XTP7LMO     MEDIUM OVERLAYS
        .... 1...         XTP7LPS     PAGE SEGMENTS
        .... .1..         XTP7LOC     OBJECT CONTAINERS
        .... ..11         *           RESERVED
19   (13) BITSTRING    1  XTP7NDEL    DELETE TIME (DSE) EXITS
        1... ....         XTP7DPD     PAGEDEFS
        .1.. ....         XTP7DFD     FORMDEFS
        ..1. ....         XTP7DCF     CODED FONTS
        ...1 ....         XTP7DMO     MEDIUM OVERLAYS
        .... 1...         XTP7DPS     PAGE SEGMENTS
        .... .1..         XTP7DOC     OBJECT CONTAINERS
        .... ..11         *           RESERVED
```

Figure 158. XTP7 parameter area format (Part 2 of 9)

```
20   (14) BITSTRING    2  XTP7RFLG      RESPONSE FLAGS
       1... ....          XTP7ETFF      EJECT TO FRONT FACING BETWEEN
                                        DOCUMENTS:
                                        (VALID FOR INITIALIZATION
                                        'XTP7INIT', BDSC 'XTP7BDS',
                                        AND EDSC 'XTP7EDS')
                                        0 = ETFF ACTIVE
                                        1 = BYPASS ETFF
       .1.. ....          XTP7TDS       TERMINATE THE DATA SET
                                        0 = DO NOT TERMINATE
                                        1 = TERMINATE
       ..1. ....          XTP7EFFH      EJECT TO FRONT FACING BEFORE
                                        JOB HEADER:
                                        (VALID FOR INITIALIZATION
                                        'XTP7INIT', BDSC 'XTP7BDS',
                                        AND EDSC 'XTP7EDS')
                                        0 = ETFF ACTIVE
                                        1 = BYPASS ETFF
       ...1 ....          XTP7HDF       SELECT HARDWARE DEFAULT FONT
                                        (VALID FOR 3800 ONLY):
                                        0 = USE HARDWARE DEFAULT
                                        1 = USE 1ST FONT MAPPED
       .... 1...          XTP7HQUE      SEND DATA SET TO JES AND
                                        MARK IT AS UNPRINTABLE
                                        (VALID ONLY WHEN DATA SET
                                        TERMINATED BY XTP7TDS):
                                        0 = PURGE THE DATA SET
                                        1 = MARK DATA SET AS UNPRINTABLE
       .... .1..          XTP7PRNS      SINGLE BYTE FONT PRUNING FLAG
                                        (VALID ONLY ON INITIALIZATION
                                        CALL 'XTP7INIT')
                                        0 = PRUNE SINGLE BYTE FONTS
                                        1 = BYPASS PRUNING
       .... ..1.          XTP7PRND      DOUBLE BYTE FONT PRUNING FLAG
                                        (VALID ONLY ON INITIALIZATION
                                        CALL 'XTP7INIT'
                                        0 = PRUNE DOUBLE BYTE FONTS
                                        1 = BYPASS PRUNING
       .... ...1          XTP7IPMF      MARK FORM REQUEST
                                        (VALID FOR INITIALIZATION
                                        'XTP7INIT', BDSC 'XTP7BDS',
                                        AND EDSC 'XTP7EDS')
                                        SEE XTP7IPCC FOR COPIES.
                                        0 = NO MARKING OF INTERRUPT MESSAGE
                                            PAGE
                                        1 = MARK INTERRUPT MESSAGE PAGE
       1... ....          XTP7IPOS      OFFSET STACK INTERRUPT MESSAGE PAGE.
                                        (VALID FOR INITIALIZATION
                                        'XTP7INIT', BDSC 'XTP7BDS',
                                        AND EDSC 'XTP7EDS')
                                        0 = NO OFFSET STACKING
                                        1 = OFFSET STACKING
       .1.. ....          XTP7MTOF      MAP FONTS TO OUTLINE FONTS
                                        (VALID ONLY ON INITIALIZATION
                                        CALL 'XTP7INIT')
                                        0 = DO NOT MAP FONTS TO OUTLINE FONTS
                                        1 = MAP FONTS TO OUTLINE FONTS
       ..1. ....          XTP7RFNF      RECOVER FROM FONT-NOT-FOUND
                                        CONDITIONS (VALID ONLY ON
                                        INITIALIZATION CALL 'XTP7INIT').
                                        0 = DO NOT QUERY HOST FONT LIBRARIES
                                            TO ENSURE THAT THE MAPPED FONT
                                            ACTUALLY EXISTS.
                                        1 = QUERY THE HOST FONT LIBRARIES FOR
                                            THE MAPPED FONT.  IF THE MAPPED
                                            DOES NOT EXIST, GO BACK AND USE
                                            THE ORIGINAL FONT.
                                            NOTE:  DUE TO THE ADDITIONAL
                                            LIBRARY QUERIES FOR EVERY FONT
                                            RESOURCE, PERFORMANCE WILL BE
                                            DEGRADED IF THIS OPTION IS
                                            TURNED ON.
```

Figure 159. XTP7 parameter area format (Part 3 of 9)

```
      ...1  ....          XTP7IPSU      SUPPRESS INTERRUPT MESSAGE PAGES
                                        (VALID FOR INITIALIZATION
                                        'XTP7INIT')
                                        0 = ALLOW INTERRUPT MSG PAGES
                                        1 = SUPPRESS INTERRUPT MSG PAGES
      .... 1...          XTP738MO      SET THE MEDIA ORIGIN TO THE
                                        3800 MEDIA ORIGIN FOR THE USER'S
                                        DATA SET
                                        (VALID FOR INITIALIZATION,
                                        'XTP7INIT', BDSC 'XTP7BDS',
                                        AND EDSC 'XTP7EDS')
                                        0 = DO NOT SET MEDIA ORIGIN TO
                                            3800 MEDIA ORIGIN
                                        1 = SET MEDIA ORIGIN TO
                                            3800 MEDIA ORIGIN
      .... .1..          XTP7MDSD      SEND THE MESSAGE DATA SET TO A
                                        SYSOUT DATA SET (VALID FOR
                                        INITIALIZATION 'XTP7INIT', BDSC
                                        'XTP7BDS', AND EDSC 'XTP7EDS')
                                        0 = DO NOT SEND MSG DATA SET TO A
                                            SYSOUT DATA SET
                                        1 = SEND MSG DATA SET TO A SYSOUT
                                            DATA SET
      .... ..1.          XTP7HCA       RESERVED - DO NOT USE
      .... ...1          XTP7C2SI      CONVERT A MULTIPLE-CELLED IM1
                                        IMAGE TO A SINGLE IOCA IMAGE
                                        (VALID FOR INITIALIZATION
                                        'XTP7INIT', BDSC 'XTP7BDS',
                                        AND EDSC 'XTP7EDS')
                                        0 = CONVERT TO MULTIPLE IMAGES
                                        1 = CONVERT TO SINGLE IOCA IMAGE
 22  (16) CHARACTER   14 XTP7RRLV      REASONABLE RESOURCE LOADED
                                        VALUE-DETERMINES THE NUMBER OF
                                        RESOURCES RETAINED AT DATA SET END
 22  (16) SIGNED       2 XTP7RPD       RRLV FOR PAGEDEFS
 24  (18) SIGNED       2 XTP7RFD       RRLV FOR FORMDEFS
 26  (1A) SIGNED       2 XTP7RMO       RRLV FOR OVERLAYS
 28  (1C) SIGNED       2 XTP7RPS       RRLV FOR PAGE SEGMENTS
  0  (1E) SIGNED       2 XTP7RCF       RRLV FOR FONTS
 32  (20) SIGNED       2 XTP7ROC       RRLV FOR OBJECT CONTAINERS
 34  (22) CHARACTER    2 *             RESERVED
 36  (24) CHARACTER    8 XTP7RLST      APSURLST INFORMATION
 36  (24) ADDRESS      4 XTP7LSTP      POINTER TO APSURLST
 40  (28) SIGNED       2 XTP7RESC      RESOURCE COUNT-# OF DISTINCT
                                        RESOURCES IN RLST CHAIN
 42  (2A) SIGNED       2 XTP7LSTC      LIST COUNT-# OF ENTRIES IN
                                        RLST CHAIN
 44  (2C) CHARACTER    8 XTP7OVLY      NAME OF OVERLAY CURRENTLY
                                        BEING PROCESSED
 52  (34) CHARACTER   20 XTP7PINF      PRINTER INFORMATION
 52  (34) CHARACTER    8 XTP7PRT       PRINTER TYPE AND MODEL
 52  (34) CHARACTER    1 XTP7ATCH      ATTACHMENT TYPE
 53  (35) CHARACTER    4 XTP7FMLY      DEVICE FAMILY
 57  (39) CHARACTER    2 XTP7PRTT      DEVICE TYPE
 59  (3B) CHARACTER    1 XTP7PRTM      MODEL NUMBER
 60  (3C) BITSTRING    1 XTP7PFLG      PRINTER FLAGS
      1... ....          XTP7PSEC      SECURE PRINTER
      .111 1111          *             RESERVED
 61  (3D) CHARACTER    3 *             RESERVED
 64  (40) CHARACTER    8 XTP7PNAM      PRINTER NAME
```

Figure 160. XTP7 parameter area format (Part 4 of 9)

```
 72   (48) CHARACTER    6  *           MESSAGE INSERT INFORMATION
                                       POINTER & LENGTH OF TEXT FOR
                                       OPTIONAL MESSAGE INSERT FOR APS954I
 72   (48) ADDRESS      4  XTP7MSGP    POINTER TO USER SUPPLIED, 64
                                       BYTE MESSAGE INSERT
 76   (4C) UNSIGNED     1  XTP7MSGL    LENGTH OF MESSAGE INSERT
                                       (0-64)
 77   (4D) CHARACTER    1  *           RESERVED
 78   (4E) BITSTRING    1  XTP7CRLF    CRL FLAGS
                                       0 = FREE STORAGE
                                       1 = KEEP THE CRL
      1... ....            XTP7CFPD    FREE/KEEP PAGEDEF CRL
      .1.. ....            XTP7CFFD    FREE/KEEP FORMDEF CRL
      ..1. ....            XTP7CFCF    FREE/KEEP FONT DEFAULT
      ...1 ....            XTP7CFMO    FREE/KEEP OVLY DEFAULT
      .... 1...            XTP7CFPS    FREE/KEEP SEGMENT/OCA DEFAULT CRL
      .... .1..            XTP7CSSD    FREE/KEEP SECUR. DEFN DEFAULT CRL
      .... ..1.            XTP7CSCF    FREE/KEEP SECUR. FONT DEFAULT CRL
      .... ...1            XTP7CSMO    FREE/KEEP SECUR. OVLY DEFAULT CRL
 79   (4F) BITSTRING    1  XTP7CRF2    CRL FLAGS
                                       0 = FREE STORAGE
                                       1 = KEEP THE CRL
      1... ....            XTP7CSPS    FREE/KEEP SECUR. SEGMENT DEFAULT CRL
      .1.. ....            XTP7CFOC    FREE/KEEP OBJ CONTAINER CRL
      ..1. ....            XTP7C2CF    FREE/KEEP FONT 240 CRL
      ...1 ....            XTP7C3CF    FREE/KEEP FONT 300 CRL
      .... 1...            XTP7C2MO    FREE/KEEP OVERLAY 240 CRL
      .... .1..            XTP7C3MO    FREE/KEEP OVERLAY 300 CRL
      .... ..1.            XTP7C2PS    FREE/KEEP SEGMENT/OCA 240 CRL
      .... ...1            XTP7C3PS    FREE/KEEP SEGMENT/OCA 300 CRL
 80   (50) CHARACTER   40  XTP7CRLP    POINTERS TO THE CONCATENATION
                                       RETRIEVAL LISTS
 80   (50) ADDRESS      4  XTP7CLPD    POINTER TO PAGEDEF CRL
 84   (54) ADDRESS      4  XTP7CLFD    POINTER TO FORMDEF CRL
 88   (58) ADDRESS      4  XTP7CLCF    POINTER TO FONT DEFAULT CRL
 92   (5C) ADDRESS      4  XTP7CLMO    POINTER TO OVERLAY DEFAULT CRL
 96   (60) ADDRESS      4  XTP7CLPS    POINTER TO SEGMENT/OCA DEFAULT CRL
100   (64) ADDRESS      4  XTP7CLOC    POINTER TO OBJECT CONTAINER CRL
104   (68) ADDRESS      4  XTP7CLSD    PTR TO SECURITY DEFN DEFAULT CRL
108   (6C) ADDRESS      4  XTP7CLSF    PTR TO SECURITY FONT DEFAULT CRL
112   (70) ADDRESS      4  XTP7CLSO    PTR TO SECURITY OVLY DEFAULT CRL
116   (74) ADDRESS      4  XTP7CLSP    PTR TO SECURITY SGMT DEFAULT CRL
120   (78) BITSTRING    1  XTP7MFLG    MORE RESPONSE FLAGS
      1... ....            XTP7JHMO    SET THE JOB HEADER MEDIA
                                       ORIGIN TO TOP LEFT CORNER
                                       (VALID FOR INITIALIZATION
                                       'XTP7INIT', BDSC 'XTP7BDS',
                                       AND EDSC 'XTP7EDS')
                                       0 = DO NOT SET JOB HEADER
                                           MEDIA ORIGIN
                                       1 = SET JOB HEADER MEDIA
                                           ORIGIN TO TOP LEFT
                                           CORNER
      .1.. ....            XTP7JTMO    SET THE JOB TRAILER MEDIA
                                       ORIGIN TO TOP LEFT CORNER
                                       (VALID FOR INITIALIZATION
                                       'XTP7INIT', BDSC 'XTP7BDS',
                                       AND EDSC 'XTP7EDS')
                                       0 = DO NOT SET JOB
                                           TRAILER MEDIA ORIGIN
                                       1 = SET JOB TRAILER MEDIA
                                           ORIGIN TO TOP LEFT
```

Figure 161. XTP7 parameter area format (Part 5 of 9)

```
     ..1. ....          XTP7DHMO    SET THE DATA SET HEADER
                                    MEDIA ORIGIN TO TOP LEFT
                                    CORNER (VALID FOR
                                    INITIALIATION 'XTP7INIT', BDSC
                                    'XTP7BDS', AND EDSC 'XTP7EDS')
                                    0 = DO NOT SET DATASET
                                        HEADER MEDIA ORIGIN
                                    1 = SET DATASET HEADER
                                        MEDIA ORIGIN TO TOP
                                        LEFT CORNER
     ...1 ....          XTP7MDMO    SET THE MESSAGE DATASET
                                    MEDIA ORIGIN TO TOP LEFT
                                    CORNER (VALID FOR
                                    INITIALIZATION 'XTP7INIT', BDSC
                                    'XTP7BDS', AND EDSC 'XTP7EDS')
                                    0 = DO NOT SET MESSAGE
                                        DATASET MEDIA ORIGIN
                                    1 = SET MESSAGE DATASET
                                        ORIGIN TO TOP LEFT
                                        CORNER
     .... 1...          XTP7RSTR    ABEND RESTART FLAG
                                    (VALID ONLY ON TERMINATION
                                    CALL 'XTP7TERM')
                                    0 = NO ABEND OR ABEND IS
                                        NOT RESTARTABLE
                                    1 = RESTARTABLE ABEND
                                        OCCURRED
     .... .1..          XTP7WIDE    WIDE OR NARROW PAPER FLAG - (VALID
                                    FOR INITIALIZATION "XTP7INIT", BDSC
                                    "XTP7BDS", AND EDSC "XTP7EDS").
                                    THIS FLAG IS ONLY VALID WHEN ONE OR
                                    MORE OF THE 3800 MEDIA ORIGIN FLAGS
                                    IS SET AND THE PRINTER IS IN CUT-SHEET
                                    EMULATION MODE:
                                    1 = LANDSCAPE PAGES ARE GENERATED
                                        FOR PRINTING WHEN IN CUT-SHEET
                                        EMULATION MODE.
                                    0 = PORTRAIT PAGES ARE GENERATED
                                        FOR PRINTING WHEN IN CUT-SHEET
                                        EMULATION MODE.
     .... ..1.          XTP7IHIB    INHIBIT RECOVERY FLAG (VALID FOR
                                    INITIALIZATION "XTP7INIT", BDSC
                                    "XTP7BDS", AND EDSC "XTP7EDS")
                                    0 = ALLOW RECOVERY.
                                    1 = INHIBIT RECOVERY.
     .... ...1          XTP7ENDC    END THE SNA LU1 CONVERSATION (VALID
                                    ONLY ON INIT CALL "XTP7INIT"):
                                    1 = END THE SNA LU1 CONVERSATION
                                        WITHOUT ENDING THE SESSION.
                                    0 = ONLY END THE SNA CONVERSATION
                                        WHEN THE SESSION ENDS.
121  (79) CHARACTER   2  XTP7ABRC    ABEND REASON CODE (VALID ONLY ON
                                    TERMINATION CALL 'XTP7TERM')
```

Figure 162. XTP7 parameter area format (Part 6 of 9)

```
123   (7B) UNSIGNED    1    XTP7JOG      JOGGING VALUE - (VALID FOR
                                         INITIALIZATION "XTP7INIT" AND BDSC
                                         "XTP7BDS" FOR FIRST DATA SET OF A JOB
                                         0 = DEFAULT TO JES COPYMARK SETTING.
                                         1 = DO NOT JOG.
                                         2 = JOG ON A DATA SET BOUNDARY.
                                         3 = JOG ON A JOB BOUNDARY.
124   (7C) UNSIGNED    2    XTP7FMT      CURRENT DATA SET RESOLUTION VALUE SET
                                         BY EXIT 7
126   (7E) BITSTRING   1    XTP7CRF3     CRL FLAGS
                                         0 = FREE STORAGE
                                         1 = KEEP THE CRL
      1... ....             *            RESERVED
      .1.. ....             XTP7CFPF     FREE/KEEP FONT PATH CRL
      ..1. ....             XTP7S2CF     FREE/KEEP SECUR. FONT 240 CRL
      ...1 ....             XTP7S3CF     FREE/KEEP SECUR. FONT 300 CRL
      .... 1...             XTP7S2MO     FREE/KEEP SECUR. OVLY 240 CRL
      .... .1..             XTP7S3MO     FREE/KEEP SECUR. OVLY 300 CRL
      .... ..1.             XTP7S2PS     FREE/KEEP SECUR. SEG 240 CRL
      .... ...1             XTP7S3PS     FREE/KEEP SECUR. SEG 300 CRL
127   (7F) CHARACTER   1    *            RESERVED
128   (80) CHARACTER   56   XTP7CRP2     POINTERS TO THE CONCATENATION
                                         RETRIEVAL LISTS
128   (80) ADDRESS     4    XTP72LCF     POINTER TO FONT 240 CRL
132   (84) ADDRESS     4    XTP73LCF     POINTER TO FONT 300 CRL
136   (88) ADDRESS     4    XTP72LMO     POINTER TO OVLY 240 CRL
140   (8C) ADDRESS     4    XTP73LMO     POINTER TO OVLY 300 CRL
144   (90) ADDRESS     4    XTP72LPS     POINTER TO SEGM/OCA 240 CRL
148   (94) ADDRESS     4    XTP73LPS     POINTER TO SEGM/OCA 300 CRL
152   (98) ADDRESS     4    XTP7CRFP     POINTER TO FONT PATH CRL
156   (9C) ADDRESS     4    *            RESERVED
160   (A0) ADDRESS     4    XTP72LSF     POINTER TO SECUR FONT 240 CRL
164   (A4) ADDRESS     4    XTP73LSF     POINTER TO SECUR FONT 300 CRL
168   (A8) ADDRESS     4    XTP72LSO     POINTER TO SECUR OVLY 240 CRL
172   (AC) ADDRESS     4    XTO73LSO     POINTER TO SECUR OVLY 300 CRL
176   (B0) ADDRESS     4    XTP72LSP     POINTER TO SECUR SEG 240 CRL
180   (B4) ADDRESS     4    XTP73LSP     POINTER TO SECUR SEG 300 CRL
184   (B8) BITSTRING   1    XTP7MFG1      MORE RESPONSE FLAGS
      1... ....             XTP7APSF      ACKNOWLEDGEMENT PER SHEET
                                          FLAG VALID FOR CHANNEL-ATTACHED
                                          PRINTERS ONLY (VALID ONLY ON
                                          INITIALIZATION CALL "XTP7INIT")
                                          0 = ACK PER PAGE
                                          1 = ACK PER SHEET
      .1.. ....             XTP7NSF       CSE NEW SHEET FLAG
                                          (VALID ONLY ON INITIALIZATION
                                          CALL "XTP7INIT")
                                          0 = IN CSE MODE, JOG(AOS) OR
                                              THE FIRST PAGE OF A DATA
                                              SET CAUSES A SKIP TO THE
                                              NEXT FRONT-SIDE PARTITION
                                          1 = IN CSE MODE, JOG(AOS) OR
                                              THE FIRST PAGE OF A DATA
                                              SET CAUSES A SKIP TO A
                                              NEW SHEET
      ..1. ....             XTP7CLOS      CLOSE RESOURCE LIBRARIES
                                          (VALID ONLY ON INITIALIZATION
                                          CALL "XTP7INIT")
                                          0 = DON'T CLOSE LIBRARIES
                                          1 = CLOSE WHEN IDLE
      ...1 ....             XTP7SDCH      SEND DEFAULT CHARACTER FLAG -
                                          USED FOR SINGLE-BYTE OUTLINE
                                          FONT CODE PAGE (VALID FOR
                                          INITIALIZATION "XTP7INIT", BDSC
                                          "XTP7BDS", AND EDSC "XTP7EDS")
                                          0 = DON'T SEND DEFAULT CHARACTER
                                          1 = SEND DEFAULT CHARACTER
```

Figure 163. XTP7 parameter area format (Part 7 of 9)

```
        .... 1...           XTP7CINR        CAPTURE INLINE RESOURCES WHEN
                                            CONNECTED VIA DPF OR WPM (VALID FOR
                                            INITIALIZATION "XTP7INIT", BDSC
                                            "XTP7BDS", AND EDSC "XTP7EDS")
                                            0 = DO NOT CAPTURE INLINE
                                                RESOURCES WHEN CONNECTED
                                                VIA DPF OR WPM
                                            1 = CAPTURE INLINE RESOURCES
                                                WHEN CONNECTED VIA DPF OR WPM
        .... .1..           XTP7RDSR        RELEASE DATA SETS WHEN
                                            REPOSITIONING (VALID FOR
                                            INITIALIZATION "XTP7INIT")
                                            0 = DO NOT RELEASE DATA SETS
                                                WHEN REPOSITIONING
                                            1 = RELEASE DATA SETS WHEN
                                                REPOSITIONING
        .... ..1.           XTP7SUCM        SUPPRESS COPY MARKS (VALID FOR
                                            INITIALIZATION "XTP7INIT", BDSC
                                            "XTP7BDS", AND EDSC "XTP7EDS")
                                            0 = DO NOT SUPPRESS COPY MARKS
                                            1 = SUPPRESS COPY MARKS
        .... ...1           XTP7RSV1        RESERVED - DO NOT USE
 185  (B9) CHARACTER    8   XTP7PRMD        PRMODE - SPECIFIES THE TYPE
                                            OF DATA IN THE PRINT DATA SET
                                            (VALID FOR INITIALIZATION
                                            "XTP7INIT" AND BDSC "XTP7BDS")
                                            ONLY SOSI1, SOSI2, SOSI3, OR
                                            SOSI4 CAN BE SPECIFIED.
 193  (C1) CHARACTER    1   *               RESERVED

 194  (C2) UNSIGNED     2   XTP7DRES        DEVICE RESOLUTION (PELS/10 IN)
                                            FOR X DIRECTION.
                                            X'960' = 240 PEL PRINTER
                                            X'BB8' = 300 PEL PRINTER
 196  (C4) CHARACTER   12   XTP7BINS        SIDE AND EDGE SENSITIVITY BINS
 196  (C4) UNSIGNED     1   XTP7BINA        FIRST BIN
 197  (C5) UNSIGNED     1   XTP7SIMA        FIRST SIMPLEX BIN
 198  (C6) UNSIGNED     1   XTP7DUPA        FIRST DUPLEX BIN
 199  (C7) UNSIGNED     1   XTP7BINB        SECOND BIN
 200  (C8) UNSIGNED     1   XTP7SIMB        SECOND SIMPLEX BIN
 201  (C9) UNSIGNED     1   XTP7DUPB        SECOND DUPLEX BIN
 202  (CA) UNSIGNED     1   XTP7BINC        THIRD BIN
 203  (CB) UNSIGNED     1   XTP7SIMC        THIRD SIMPLEX BIN
 204  (CC) UNSIGNED     1   XTP7DUPC        THIRD DUPLEX BIN
 205  (CD) UNSIGNED     1   XTP7BIND        FOURTH BIN
 206  (CE) UNSIGNED     1   XTP7SIMD        FOURTH SIMPLEX BIN
 207  (CF) UNSIGNED     1   XTP7DUPD        FOURTH DUPLEX BIN
 208  (D0) CHARACTER    1   XTP7MFG2        MORE RESPONSE FLAGS
        1... ....           XTP7ASAP        AFP STATISTICS ABOUT THE PRINT
                                            FILE (VALID ONLY ON THE BDS CALL
                                            "XTP7BDS")
                                            0 = DO NOT GATHER AFP STATISTICS
                                                FOR THIS DATA SET (PRINT FILE)
                                            1 = GATHER AFP STATISTICS FOR THIS
                                                PRINT FILE
```

Figure 164. XTP7 parameter area format (Part 8 of 9)

```
          .1.. ....          XTP7SENS          SIDE AND EDGE SENSITIVITY FLAG
                                               (VALID ONLY ON INIT CALL "XTP7INIT")
                                               0 = DO NOT USE SIDE AND EDGE
                                               SENSITIVITY
                                               1 = ACTIVATE SIDE AND EDGE
                                               SENSITIVITY
          ..1. ....          XTP7IMSG          INTERVENTION MESSAGE FLAG
                                               (VALID ONLY ON INIT CALL
                                               "XTP7INIT")
                                               0 = DO NOT DISPLAY THE INTERVENTION
                                                   MESSAGE ON THE OPERATOR CONSOLE
                                               1 = DISPLAY THE INTERVENTION MESSAGE
                                                   ON THE OPERATOR CONSOLE
          ...1 ....          XTP7PJO           PRINT JOB FINISHING OFF
                                               (VALID FOR BDSC 'XTP7BDS'
                                               AND EDSC 'XTP7EDS')
                                               0 = DO NOT ALTER PRINT JOB
                                                   FINISHING SETTING
                                               1 = SET PRINT JOB FINISHING OFF
          .... 1...          XTP7CPPP          CSE PRESERVE PAGE PLACEMENT
                                               (VALID ONLY ON INIT CALL "XTP7INIT")
                                               0 = DO NOT PRESERVE PAGE PLACEMENT
                                                   ACROSS REPOSITIONS
                                               1 = PRESERVE PAGE PLACEMENTS ACROSS
                                                   REPOSITIONS

          .... .1..          XTP7LCNT          USE LINE COUNT (LINECT) IN LINE
                                               MODE MIGRATION TO CALCULATE THE
                                               NUMBER OF LINES ON A PAGE
                                               (VALID ONLY ON INIT CALL "XTP7INIT")
                                               0 = DO NOT USE LINECT
                                               1 = USE LINECT
          .... ..1.          XTP7HLCF          HIGHLIGHT COMMUNICATIONS FAILURE
                                               MESSAGE APS6501A
                                               (VALID ONLY ON INIT CALL "XTP7INIT")
                                               0 = DO NOT HIGHLIGHT
                                               1 = HIGHLIGHT MESSAGE

          .... ...1          XTP7LMCM          ISSUE LINE-MODE CONVERSION MESSAGE
                                               APS973I IF PAPER-LENGTH MISMATCH
                                               ERROR IN AFP DOWNLOAD PLUS
                                               (VALID ONLY ON INIT CALL "XTP7INIT")
                                               0 = DO NOT ALLOW MESSAGE
                                               1 = ALLOW MESSAGE
 209   (D1) CHARACTER   24   *                 RESERVED
```

Figure 165. XTP7 parameter area format (Part 9 of 9)

XTP7 fields

XTP7CID
> Contains the control block identifier XTP7.

XTP7LENG
> Contains the length of the XTP7 DSECT.

XTP7ETYP
> Contains information flags set by PSF to indicate which type of exit call is in progress. Only one of these flags is set to 1 for each call to the exit.

XTP7RTYP
> Information flags set by PSF to indicate the type of resource that is processed. For calls to the exit that involve resources (RAC, RLC, and RDC), only one XTP7RTYP flag is set to 1. For calls to the exit that do not involve resources (INIT, BDSC, EDSC, and TERM), all XTP7RTYP flags are set to 0.

XTP7DSTY
> Information flags set by PSF to indicate which type of data set is being processed. Only one of these flags is set to 1 except during the INIT and TERM calls, when all of these flags are set to 0.

> **XTP7AUX**
>> PSF is processing auxiliary resources. This includes all resources that are used in auxiliary data sets and the default page definition and form definition. Auxiliary data sets are the job header and trailer, data set header, and message data set.

XTP7PDS

PSF is processing a user's print data set.

XTP7DSAT

Information flags, set by PSF, that indicate the specific type of data set being processed. Only one of these flags is set to 1 except during the INIT and TERM calls, when all of these flags are set to 0. These flags give more detail than XTP7DSTY about the data set being processed.

XTP7PDFT

PSF is processing the default page definition or form definition.

XTP7PJHD

PSF is processing the job header.

XTP7PJTR

PSF is processing the job trailer.

XTP7PDSH

PSF is processing the data set header.

XTP7PMDS

PSF is processing the message data set.

XTP7PLDS

PSF is processing the last print data set in the job; a job trailer is printed.

XTP7ACSE

This option can be set by the Printer Inventory or during the INIT call, the BDSC, or the EDSC. It remains set between calls and is valid only when CSE is being used. PSF initially sets this field to X'00' (XTP7NCK), which means PSF does not check whether a page fits 2 up on the sheet. If this field is set to X'01' (XTP7K1P), PSF uses a new copy group to check whether the first page printed fits 2 up on the sheet. If this field is set to X'02' (XTP7CKAP), PSF checks all the font side pages to see whether they fit 2 up on the sheet. If XTP7ACSE is set to any other value, it defaults to X'00'. For information about specifying this option in the Printer Inventory, see Check CSE fit.

XTP7IPCC

This option can be set by the Printer Inventory or during the INIT call, the BDSC, or the EDSC; it remains set between calls. PSF initially sets this field to X'00', which causes one copy of the interrupt message page to be printed. If flag XTP7IPMF is set, and Exit 7 sets this field to a binary value 1–10 decimal inclusive, PSF produces many copies of the interrupt message page. If XTP7IPCC is set to a value outside the valid range, it defaults to 1. This field is ignored if the XTP7IPMF flag is not set or if the device type is 3800. For details about copies when forms are marked, see "Forms marking" on page 335. For information about specifying this option in the Printer Inventory, see Interrupt message page: Copies.

XTP7NOTY

Flags that indicate when the exit wants control. The default state of all XTP7NOTY flags is 0 (inactive). If an XTP7NOTY flag is set to 1, PSF calls the exit when it gets to that particular point in the processing of a data set. An XTP7NOTY flag set to 0 indicates that PSF does not call the exit.

- The state of the XTP7NOTY flags can be changed only by the exit.

- PSF supplies the current settings of the XTP7NOTY flags to the exit before it calls the exit.

- All the flags in XTP7NOTY can be set independent of one another.

XTP7MISC

XTP7EBDS

A flag that tells PSF the exit is to be called for the beginning of the data set call (BDSC) function.

XTP7ETRM

A flag that tells PSF the exit is to be called for the FSA termination call (TERM) function. This is for stopping the FSA both normally and abnormally.

XTP7EEDS

A flag that tells PSF that the exit is to be called for the end-data-set call (EDSC) function.

XTP7NACC

Flags telling PSF that the exit is to be called for the resource access call (RAC) function. Each resource type has an XTP7NACC flag.

XTP7NLD

Flags telling PSF that the exit is to be called for the resource load call (RLC) function. Each resource type has an XTP7NLD flag.

XTP7NDEL

Flags telling PSF that the exit is to be called for the resource delete call (RDC) function. Each resource type has an XTP7NDEL flag.

XTP7RFLG

Flags that the exit can modify to request certain functions.

XTP7ETFF

This option can be set by the Printer Inventory or during the INIT call, the BDSC, or the EDSC; it remains set between calls. PSF sets this flag to B'0' to indicate that an eject to front facing is to be done before the beginning of the next document. Exit 7 can set this field to B'1' to bypass the ETFF function. This flag is valid only for continuous-forms printers. For information about specifying this option in the Printer Inventory, see Eject to front facing.

XTP7TDS

PSF sets this flag to B'0'. If the exit sets this flag to B'1', PSF stops processing the current data set. This flag is valid for the BDSC, RAC, and RLC exit calls to Exit 7.

XTP7EFFH

This option can be set by the Printer Inventory or during the INIT call, the BDSC, or the EDSC; it remains set between calls. PSF sets this flag to B'0' to indicate that an eject to front facing is to be done before the job header page. Exit 7 can set this field to B'1' to bypass the ETFF function. This flag is valid only for continuous-forms printers. For information about specifying this option in the Printer Inventory, see Eject to front facing.

XTP7HDF

This option can be set by the Printer Inventory or during the INIT call. It remains set between calls and is only valid with 3800 printers. PSF sets this flag to B'0' to indicate that the printer uses the hardware default font. If Exit 7 sets this field to B'1', PSF allows the printer to substitute the first font in the current font list and replace the hardware default font. For more information about selecting the 3800 hardware default font, see "Font selection" on page 335. For information about specifying this option in the Printer Inventory, see 3800 compatibility: Override default font.

XTP7HQUE

Valid only when Exit 7 requested the end of processing for the current data set (XTP7TDS=B'1'). PSF initially sets this field to B'0'. When processing ends, the data set is purged. Exit 7 can set this field to B'1' to request that the data set is sent to JES and marked as unprintable rather than purged.

XTP7PRNS

This option can be set by the Printer Inventory or during the INIT call. PSF initially sets this field to B'0' to indicate that non-3800 single-byte fonts are to be pruned. For more information about font pruning, see "Number and type of resources" on page 188. If Exit 7 sets this field to B'1', PSF does not prune single-byte fonts. For information about specifying this option in the Printer Inventory, see Prune single-byte fonts.

XTP7PRND

This option can be set by the Printer Inventory or during the INIT call. PSF initially sets this field to B'0' to indicate that non-3800 double-byte fonts are to be pruned. For more information about font pruning, see "Number and type of resources" on page 188. If Exit 7 sets this field to B'1', PSF is not to prune double-byte fonts. For information about specifying this option in the Printer Inventory, see Prune double-byte fonts.

XTP7IPMF

This option can be set by the Printer Inventory or during the INIT call, the BDSC, or the EDSC; it remains set between calls. PSF initially sets this field to B'0' to indicate that no form marks are required for the interrupt message page. If Exit 7 sets this field to B'1', PSF marks the interrupt

message page with form marks. For information about specifying this option in the Printer Inventory, see Interrupt message page: Mark page.

XTP7IPOS

This option can be set by the Printer Inventory or during the INIT call, the BDSC, or the EDSC; it remains set between calls. PSF initially sets this field to B'0' to indicate that no offset stacking is required for the interrupt message page. If Exit 7 sets this field to B'1', the printed output is offset-stacked, starting at the interrupt message page. For information about specifying this option in the Printer Inventory, see Interrupt message page: Offset page.

XTP7MTOF

This option can be set by the Printer Inventory, on the PRINTDEV statement, or during the INIT call. PSF initially sets this field to B'0' to indicate that fonts are not mapped to outline fonts. If Exit 7 sets this flag to B'1', PSF maps fonts to an equivalent outline font if the printer supports outline fonts. See "Mapping to outline fonts" on page 236. For information about specifying this option in the Printer Inventory, see Map to outline fonts. For information about specifying this option on the PRINTDEV statement, see MAP2OLN.

XTP7RFNF

This option can be set by the Printer Inventory, on the PRINTDEV statement or during the INIT call. The Recover-from-Font-Not-Found flag applies only when XTP7MTOF is set ON to B'1'. PSF initially sets this field to B'0' to indicate that PSF is not to query the host libraries to make sure that the mapped font exists before attempting to load it into the printer. See "Recovering from font not found" on page 236. For information about specifying this option in the Printer Inventory, see Recover from font not found. For information about specifying this option on the PRINTDEV statement, see MAP2OLN.

XTP7IPSU

This option can be set by the Printer Inventory or during the INIT call. PSF initially sets this field to B'0' to indicate that interrupt message pages are wanted. If Exit 7 sets this flag to B'1', interrupt message pages are suppressed (are not printed). An interrupt message page is printed, for example, at the point at which a data set is backspaced. Intervention conditions, such as a paper jam, can also cause an interrupt message page. For information about specifying this option in the Printer Inventory, see Interrupt message page.

Suppressing interrupt message pages is suggested when PSF is attached to a printer with Finisher capability; otherwise, interrupt message pages separate finished documents into two finished groups with the inserted page in the middle.

XTP738MO

This option can be set by the Printer Inventory or during the INIT call, the BDSC, or the EDSC; it remains set between calls. PSF initially sets this field to B'0' to indicate that the user's data set media origin is not set to the 3800 media origin. If Exit 7 sets this flag to B'1', PSF sets the user's data set media origin to the 3800 media origin. For more information about setting the media origin, see "Media origin" on page 335. For information about specifying this option in the Printer Inventory, see Set media origin to 3800 origin for: Data set.

XTP7MDSD

This option can be set by the Printer Inventory or during the INIT call, the BDSC, or the EDSC. However, it does not remain set between calls; it is set to the value based on the PIMSG keyword. PSF initially sets this field to the PIMSG keyword setting on the OUTPUT statement for the message data set. If Exit 7 sets this flag to B'0', PSF does not send the message data set to a SYSOUT data set. If Exit 7 sets this flag to B'1', PSF sends the message data set to a SYSOUT data set. See "Redirecting the message data set" on page 328. For information about specifying this option in the Printer Inventory, see Send msgs to SYSOUT.

XTP7HCA

Reserved; do not use.

XTP7C2SI

This option can be set by the Printer Inventory or during the INIT call, the BDSC, or the EDSC; it remains set between calls. PSF initially sets this field to B'0' to indicate that a multiple-celled IM1 image is to be converted to multiple IOCA images. If Exit 7 sets this flag to B'1', PSF converts a

multiple-celled IM1 image to a single IOCA image. This request is accepted only if the printer does not support IOCA replicate and trim. This change improves performance only if the image is a 240-pel IM1 celled image that requests replication. For any other case, this change might degrade performance. For information about specifying this option in the Printer Inventory, see Consolidate IM1 images.

XTP7RRLV
PSF provides the current reasonable resource loaded values (RRLVs) in the following fields. These fields can be set by the Printer Inventory or during Exit 7 calls. The Printer Inventory overrides these values for the INIT call only. PSF uses the RRLVs to determine how many resources remain loaded at the end of a user's print data set. For the default values of RRLV, see Table 43 on page 455.

XTP7RPD
RRLV for page definitions. For information about specifying this option in the Printer Inventory, see Page definitions.

XTP7RFD
RRLV for form definitions. For information about specifying this option in the Printer Inventory, see Form definitions.

XTP7RMO
RRLV for medium overlays. PSF always sets this value to 0. If Exit 7 sets this field to a nonzero value, PSF ignores that setting and sets the value to 0.

XTP7RPS
RRLV for page segments. For information about specifying this option in the Printer Inventory, see Page definitions.

XTP7RCF
RRLV for fonts. A double-byte font is considered one font. For information about specifying this option in the Printer Inventory, see Fonts.

XTP7ROC
RRLV for object containers. For information about specifying this option in the Printer Inventory, see Object containers.

XTP7RLST
PSF supplies information about the resource list (APSURLST) control block in the following fields:

XTP7LSTP
Pointer to the first RLST control block in the RLST chain. This field is set to 0 if no RLST chain exists —that is, if this call to the exit does not involve resources.

XTP7RESC
Contains the number of distinct resources that are being described in the RLST chain.

XTP7LSTC
Contains the total number of RLST entries in the RLST chain. This value differs from XTP7RESC only during the RAC exit, when the RLST chain is describing a font that contains multiple entries in the Map Coded Font (MCF) structured field. In all other cases, this value is the same as XTP7RESC.

XTP7OVLY
PSF supplies the name of the overlay that is processed by PSF. If PSF is not involved in processing an overlay or any resource nested in the overlay, this field is set to null (X'FFFF000000000000').

XTP7PINF
PSF provides the following information about the attached printer:

XTP7PRT
Printer type and model.

XTP7ATCH
Attachment type: C=Channel, S=SNA, T=TCP/IP.

XTP7FMLY
Device family: 3800, 3820, or, for all other IBM printers, AFP1.

XTP7PRTT
> Device type: not intended for use in user programming.

XTP7PRTM
> Model number for the IBM 3800; for example, 3800 Model 3.

> **Note:** The IBM 3800 Model 6 is listed as a Model 3.

XTP7PFLG
> Printer flags.

> **XTP7PSEC**
>> PSF sets this flag to B'1' to indicate that the printer is a secure printer; that is, it supports the Define User Area IPDS command.

XTP7PNAM
> Printer name, as designated by the installation in JES initialization statements.

PSF message APS954I allows the exit up to 64 bytes of text at the end of the message. APS954I indicates that the resource exit stopped processing the data set (see "XTP7TDS" on page 431). The optional text can be used to explain why processing of the data set ended. To get this additional text, the exit must put the applicable information into the following two fields:

XTP7MSGP
> Pointer to the text. If this field is 0, no text is added to the message.

XTP7MSGL
> Length of the text. The domain of this field is 0–64, the maximum number of bytes PSF adds to APS954I. If XTP7MSGL is 0, no text is added to the message.

XTP7CRLF, XTP7CRF2
> Flags that tell PSF to keep or delete the 10 concatenation retrieval lists (CRLs). Before the INIT call to the exit, PSF sets all CRL flags to 0. If the exit sets a CRL flag to 1, PSF retains the CRL for the entire FSA session. Only after the INIT call to the exit does PSF delete a CRL and free its storage.

XTP7CRLP
> PSF provides pointers to the 10 concatenation retrieval lists (see "APSLCRL parameter area (concatenation retrieval list)" on page 438) at every call to the exit. If the storage for a CRL is freed (see "XTP7CRLF, XTP7CRF2" on page 434) or if no library is found for a resource type, PSF sets the pointer to 0.

XTP7MFLG
> Flags that the exit can modify to request certain functions.

> **XTP7JHMO**
>> This option can be set by the Printer Inventory or during the INIT call, the BDSC, or the EDSC; it remains set between calls. PSF initially sets this field to B'0' to indicate that the job header media origin is not set to the upper left corner. If Exit 7 sets this flag to B'1', PSF sets the job header media origin to the upper left corner. For more information about setting the media origin, see "Media origin" on page 335. For information about specifying this option in the Printer Inventory, see Set media origin to 3800 origin for: Job header.

> **XTP7JTMO**
>> This option can be set by the Printer Inventory or during the INIT call, the BDSC, or the EDSC; it remains set between calls. PSF initially sets this field to B'0' to indicate that the job trailer media origin is not set to the upper left corner. If Exit 7 sets this flag to B'1', PSF sets the job trailer media origin to the upper left corner. For more information about setting the media origin, see "Media origin" on page 335. For information about specifying this option in the Printer Inventory, see Set media origin to 3800 origin for: Job trailer.

> **XTP7DHMO**
>> This option can be set by the Printer Inventory or during the INIT call, the BDSC, or the EDSC; it remains set between calls. PSF initially sets this field to B'0' to indicate that the data set header media origin is not set to the upper left corner. If Exit 7 sets this flag to B'1', PSF sets the data set header media origin to the upper left corner. For more information about setting the media origin,

see "Media origin" on page 335. For information about specifying this option in the Printer Inventory, see Set media origin to 3800 origin for: Data set header.

XTP7MDMO

This option can be set by the Printer Inventory or during the INIT call, the BDSC, or the EDSC; it remains set between calls. PSF initially sets this field to B'0' to indicate that the message data set media origin is not set to the upper left corner. If Exit 7 sets this flag to B'1', PSF sets the message data set media origin to the upper left corner. For more information about setting the media origin, see "Media origin" on page 335. For information about specifying this option in the Printer Inventory, see Set media origin to 3800 origin for: Message data set.

XTP7RSTR

Valid only on the TERM call. PSF initially sets this field to B'1' if the abend is restartable or to B'0' if no abend occurred or if the abend is not restartable. If this field is set to B'1' (restartable abend), Exit 7 can set this field to B'0' to cause PSF not to restart.

XTP7WIDE

This option can be set by the Printer Inventory or during the INIT call, the BDSC, or the EDSC; it remains set between calls. PSF initially sets this field to B'0' to indicate that portrait pages are generated for printing in cut-sheet emulation mode. If Exit 7 sets this flag to B'1', PSF generates landscape pages for printing in cut-sheet emulation mode. For more information about setting the media origin, see "Media origin" on page 335. For information about specifying this option in the Printer Inventory, see CSE orientation.

XTP7IHIB

This option can be set by the Printer Inventory or during the INIT call, the BDSC, or the EDSC; it remains set between calls. PSF initially sets this field to B'0'. If Exit 7 sets this field to B'1', PSF inhibits recovery. For more information about inhibiting recovery, see "Inhibiting PSF recovery" on page 326. For information about specifying this option in the Printer Inventory, see Inhibit recovery.

XTP7ENDC

This option can be set by the Printer Inventory or during the INIT call. PSF initially sets this field to B'0', which indicates that PSF maintains the SNA LU1 conversation between print jobs. If Exit 7 sets this field to B'1', PSF ends the SNA LU1 conversation while it maintains the SNA session with the printer when the NPRO timer expires or after no job to print is available for 1 minute and the last page printed is stacked. For information about specifying this option in the Printer Inventory, see End SNA conversation.

XTP7ABRC

Valid only on the TERM call. This field contains the abend reason code if the FSA is stopping abnormally. It contains X'0000' if the FSA is stopping normally.

XTP7JOG

This option can be set by the Printer Inventory or during the INIT call, or during the BDSC; it remains set between calls. PSF initially sets this flag to B'0' to indicate that it is not specified and the default is the COPYMARK setting. For more information, see "Separating copy marking and offset stacking" on page 158. If Exit 7 sets this flag to B'1', B'2', or B'3', PSF uses that value for offset stacking. For information about specifying this option in the Printer Inventory, see Offset stacking.

XTP7FMT

This option can be set by the Printer Inventory or during the INIT call, the BDSC, or the EDSC; it remains set between calls. PSF initially sets this field to 0 to indicate that no format resolution is specified by the exit. For more information, see "Specifying format resolution" on page 217. If Exit 7 sets this field to either 240 or 300, PSF uses that as the specified format resolution value. For information about specifying this option in the Printer Inventory, see Resolution.

XTP7CRF3

Flags that tell PSF to keep or delete the specified concatenation retrieval lists (CRLs). Before the INIT call to the exit, PSF sets all CRL flags to 0. If the exit sets a CRL flag to 1, PSF retains the CRL for the entire FSA session. Only after the INIT call to the exit does PSF delete a CRL and free its storage.

XTP7CRP2

PSF provides pointers to the specified concatenation retrieval lists (see "APSLCRL parameter area (concatenation retrieval list)" on page 438) at every call to the exit. If the storage for a CRL is freed (see "XTP7CRLF, XTP7CRF2" on page 434 and "XTP7CRF3" on page 435), or if no library is found for a resource type, PSF sets the pointer to 0.

XTP7MFG1

Flags that the exit can modify to request certain functions.

XTP7APSF

This option can be set only by the Printer Inventory or during the INIT call. PSF initially sets this field to B'0' to indicate that an acknowledgment per page is requested by PSF. If Exit 7 sets this flag to B'1', PSF requests an acknowledgment per sheet. This field applies only to channel-attached printers. See "ACK per sheet" on page 196. For information about specifying this option in the Printer Inventory, see Acknowledgement level.

XTP7NSF

This option can be set only by the Printer Inventory or during the INIT call. PSF initially sets this field to B'0' to indicate that, in CSE mode, an AOS or new data set causes a skip to the next front-side partition. If Exit 7 sets this flag to B'1', then in CSE mode, an AOS or new data set causes a skip to a sheet. See "Things to consider when using cut-sheet emulation" on page 208. For information about specifying this option in the Printer Inventory, see CSE sheet eject.

XTP7CLOS

This option can be set only by the Printer Inventory or during the INIT call. PSF initially sets this flag to B'0' to indicate that PSF does not close the resource libraries when idle. If Exit 7 sets this flag to B'1', PSF closes the resource libraries when the printer is idle for 60 seconds. The disconnect interval timer must be set to greater than 60 seconds or PSF does not close the resource libraries. Instead, PSF ends the session with the printer before it closes the resource libraries. For information about specifying this option in the Printer Inventory, see Close libraries when idle.

XTP7SDCH

This option can be set only by the Printer Inventory or during the INIT call, the BDSC, or the EDSC; it remains set between calls. PSF initially sets this flag to B'0' to indicate that PSF does not pass the default character information to the printer by fully populating the outline single-byte code page. If Exit 7 sets this flag to B'1', PSF passes the default character information to the printer by fully populating the outline single-byte code page. See "Sending default characters in single-byte outline fonts" on page 234. For information about specifying this option in the Printer Inventory, see Send default character.

XTP7CINR

This option can be set only by the Printer Inventory or during the INIT call, the BDSC, or the EDSC. PSF initially sets this flag to B'0' to indicate that PSF tells DPF or WPM to not capture inline resources. If Exit 7 sets this flag to B'1', PSF tells DPF or WPM to capture inline resources. For information about specifying this option in the Printer Inventory, see Capture inline resources.

XTP7RDSR

This option can be set only by the Printer Inventory or during the INIT call. PSF initially sets this flag to B'0' to indicate that PSF does not release spool data sets during repositioning. If Exit 7 sets this flag to B'1', PSF will release data sets during repositioning and then reacquire them after printing restarts. See "Disposition of spool data sets during repositioning" on page 325. For information about specifying this option in the Printer Inventory, see Release data set when repositioning.

XTP7SUCM

This option can be set only by the Printer Inventory or during the INIT call, the BDSC, or the EDSC. PSF initially sets this flag to B'0' to indicate that PSF tells the printer to print copy marks as specified by the JES initialization parameters. If Exit 7 sets this flag to B'1', PSF tells the printer to suppress copy marks. See "Suppressing copy marks" on page 157. For information about specifying this option in the Printer Inventory, see Suppress copy marks.

XTP7RSV1

Reserved; do not use.

XTP7PRMD

This option can be set by the Printer Inventory, on the PRINTDEV statement, during the INIT call, or during the BDSC; it remains set between calls. This field contains the default for PRMODE. Only SOSI1, SOSI2, SOSI3, or SOSI4 can be specified. Anything else is ignored. For information about specifying this option in the Printer Inventory, see Default process mode. For information about specifying this option on the PRINTDEV statement, see PRMODE.

XTP7DRES

PSF supplies the device resolution (pels/10 inches) for the X direction.

XTP7BINS

A list of possible input bins, which are mapped from the logical bins, that might contain side or edge sensitive paper.

XTP7BINA

First bin that contains side or edge sensitive paper. When set, this field remains set between calls.

XTP7SIMA

First bin that contains simplex side or edge sensitive paper. When set, this field remains set between calls.

XTP7DUPA

First bin that contains duplex side or edge sensitive paper. When set, this field remains set between calls.

XTP7BINB

Second bin that contains side or edge sensitive paper. When set, this field remains set between calls.

XTP7SIMB

Second bin that contains simplex side or edge sensitive paper. When set, this field remains set between calls.

XTP7DUPB

Second bin that contains duplex side or edge sensitive paper. When set, this field remains set between calls.

XTP7BINC

Third bin that contains side or edge sensitive paper. When set, this field remains set between calls.

XTP7SIMC

Third bin that contains simplex side or edge sensitive paper. When set, this field remains set between calls.

XTP7DUPC

Third bin that contains duplex side or edge sensitive paper. When set, this field remains set between calls.

XTP7BIND

Fourth bin that contains side or edge sensitive paper. When set, this field remains set between calls.

XTP7SIMD

Fourth bin that contains simplex side or edge sensitive paper. When set, this field remains set between calls.

XTP7DUPD

Fourth bin that contains duplex side or edge sensitive paper. When set, this field remains set between calls.

XTP7MFG2

Flags that the exit can modify to request certain functions.

XTP7ASAP

This option can be set only during the BDSC. PSF sets this flag to B'0' when this data set (print file) does not request an AFP Statistics (AFPSTATS) report or sets this bit to B'1' when this data set requests an AFPSTATS report. Exit 7 can change this setting for XTP7ASAP, thereby overriding the request as indicated by the data set. PSF does AFPSTATS gathering based on the setting of this bit after the BDSC. See "Obtaining data about print file resources" on page 181.

XTP7SENS

This option can be set during the INIT call and it remains set between calls. PSF initially sets this flag to B'0' to indicate that PSF does not use the Side and Edge Sensitive Paper Handling function. If Exit 7 set this flag to B'1', PSF activates the Side and Edge Sensitive Paper Handling function. See Chapter 16, "Using Side and Edge Sensitive Paper Handling," on page 209.

XTP7IMSG

This option can be set by the Printer Inventory, on the PRINTDEV statement, or during the INIT call. PSF initially sets this flag to B'0' to indicate that PSF does not issue a message to the operator console when the printer is SNA- or TCP/IP-attached and requires intervention. If Exit 7 sets this flag to B'1', PSF issues a message to the operator console when the printer is SNA- or TCP/IP-attached and requires intervention.

For information about specifying this option in the Printer Inventory, see Issue intervention messages. For information about specifying this option on the PRINTDEV statement, see INTRVMSG.

XTP7PJO

This option can be set only during the BDSC or the EDSC, and it remains set between calls. PSF initially sets this flag to B'0' to indicate that the print job finishing setting is not altered. If Exit 7 sets this flag to B'1', PSF sets print job finishing off.

XTP7CPPP

This option can be set only during the INIT call. PSF initially sets this flag to B'0' to indicate that PSF does not preserve page placement after a reposition in CSE mode. If Exit 7 sets this flag to B'1', PSF preserves page placement after a reposition and, in CSE mode, an AOS or new data set causes a skip to a sheet (as if XTP7NSF was set to B'1'). See "Things to consider when using cut-sheet emulation" on page 208. For information about specifying this option in the Printer Inventory, see CSE preserve page position.

XTP7LCNT

This option can be set only during the INIT call. PSF initially sets this flag to B'0' to indicate that PSF does not use LINECT in Line Mode Migration to calculate the number of lines on a page. If Exit 7 sets this flag to B'1', PSF uses LINECT in Line Mode Migration to calculate the number of lines on a page.

XTP7HLCF

This option can be set only during the INIT call. PSF initially sets this flag to B'0' to indicate that PSF does not highlight communications failure message APS6501I. If Exit 7 sets this flag to B'1', PSF highlights communications failure message APS6501A. For information about specifying this option in the Printer Inventory, see Highlight communications failure message.

XTP7LMCM

This option can be set only during the INIT call. AFP Download Plus initially sets this flag to B'0' to indicate that AFP Download Plus does not allow message APS973I to be issued. If Exit 7 sets this flag to B'1', AFP Download Plus allows message APS973I to be issued.

APSLCRL parameter area (concatenation retrieval list)

The concatenation retrieval list (CRL) is a control block that PSF builds to pass information about resource libraries to the exit. Before the INIT call to the exit, PSF builds a CRL for each resource library specified in PSF initialization statements. PSF supplies pointers to the CRLs and to flags that tell PSF whether CRL information is to be saved. The pointers and flags are in the XTP7CRLF, XTP7CRF2, and XTP7CRLP sections of APSGEXTP (see "XTP7CRLF, XTP7CRF2" on page 434 and "XTP7CRLP" on page 434).

If a CRL does not exist (either because no library was specified or because PSF freed the storage for it), the CRL pointer is set to 0. The CRL is not intended to be written to by the exit; however, PSF does not reference the CRL other than to supply the pointers to the exit.

Figure 166 on page 439 shows the format of APSLCRL parameter area, and Figure 167 on page 440 shows the format of the APSLCRL data area.

Note: The CRLs are for the system libraries only. No CRL is passed for a user library.

```
   Offsets   Type      Length   Name            Description
   0    (0) STRUCTURE    40   APSLCRL         CONCATENATION RETRIEVAL LIST
   0    (0) CHARACTER    20   CRLPFIX         PREFIX SECTION
   0    (0) CHARACTER     4   CRLID           CONTROL BLOCK ID
   4    (4) SIGNED        4   CRLLEN          LENGTH INCLUDING DATA SET
                                              ENTRIES
   8    (8) CHARACTER     6   CRLPTF          PTF DATE
  14    (E) CHARACTER     2   CRLREL          RELEASE LEVEL
  16   (10) SIGNED        1   CRLEID          OBJECT ENCODED ID FROM APSGSFX
                                              IDENTIFIES TYPE OF OBJECTS IN
                                              THIS LIBRARY.
  17   (11) BITSTRING     1   CRLSPN          SUBPOOL NUMBER
  18   (12) BITSTRING     1   CRLFLG0         FLAGS
       1... ....              CRLSECUR        SECURITY LIBRARY
       .111  1111              *              RESERVED
  19   (13) CHARACTER     1    *              RESERVED
  20   (14) CHARACTER    20   CRLBASE         BASE SECTION
  20   (14) CHARACTER     8   CRLDDNM         DDNAME FOR DATA SET ENTRIES
  28   (1C) SIGNED        2   CRLRTRVD        NUMBER OF DATA SETS FOR WHICH
                                              JFCBS WERE RETRIEVED. FROM
                                              FIELD ARLRETRV IN IHAARL OR
                                              GENERATED BY APSLOPEN.
                                              DOMAIN = (0  256)
                                              0 = NO DATA SET NAMES
                                                  RETRIEVED
  30   (1E) SIGNED        2   CRLCONC         NUMBER OF DATA SETS
                                              CONCATENATED TOGETHER. FROM
                                              FIELD ARLCONC IN IHAARL OR
                                              GENERATED BY APSLOPEN.
                                              DOMAIN = (1  256)
                                              1 = NOT A CDS
                                              2 TO 256 = CDS
  32   (20) SIGNED        2   CRLLOWCN        LOW CONCATENATION NUMBER FOR
                                              THIS DATA SET MAPPING.
                                              DOMAIN = (0  255)
  34   (22) SIGNED        2   CRLHIGCN        HIGH CONCATENATION NUMBER FOR
                                              THIS DATA SET MAPPING.
                                              DOMAIN = (0  255)
  36   (24) CHARACTER     4    *              RESERVED
  40   (28) CHARACTER         CRLEND
```

Figure 166. APSLCRL parameter area format

```
    Offsets   Type      Length  Name        Description

    0    (0) STRUCTURE          CRLDATA     DATA SET NAME AND VOLUME ENTRY. NUMBER
                                            OF ENTIRES IS EQUAL VALUE IN FIELD
                                            CRLRTRVD.
    0    (0) CHARACTER   317   CRLFLEN      FIXED LENGTH PORTION OF CRLDATA
    0    (0) SIGNED        2   CRLDSELN     LENGTH OF DATA SET AND VOLUME SERIAL
                                            LIST PLUS LENGTH BYTES. CRLDSELN IS
                                            USED TO INCREMENT TO NEXT ENTRY.
                                            MAPPED BY CRLDATA. DOMAIN = (0 32767)
    2    (2) SIGNED        2   CRLCATN      CONCATENATION NUMBER FOR DATA SET NAME
                                            IN CRLDSNM. GENERATED BY APSLJFCB OR
                                            APSLOPEN. DOMAIN = (0 255)
                                            0 = FIRST OR ONLY DATA SET
                                            1 = 2ND DATA SET
    4    (4) SIGNED        2   CRLNVOL      NUMBER OF VOLUMES FROM JFCBNVOL.
                                            DOMAIN = (1 256)
    6    (6) CHARACTER    44   CRLDSNM      JFCB INDEX BY CRLRTRVD. X'40' IN FIRST
                                            BYTE OF ENTRY INDICATES DATA SET NAME
                                            COULD NOT BE RETRIEVED.
                                            DOMAIN = (ALPHANUMERIC)
   50   (32) CHARACTER   255   CRLPNM       UNIX FILE PATH NAME. IF THE CRLDATA
                                            ENTRY IS FOR AN HFS PATH NAME, CRLDSN
                                            IS SET TO "PATH= SPECIFIED".
  305  (131) CHARACTER     1   *            RESERVED FOR BOUNDARY.
  306  (132) UNSIGNED      2   CRLPNMLEN    LENGTH OF UNIX FILE PATH NAME.
  308  (134) BITSTRING     1   CRLRFORM     RECORD ATTRIBUTE FROM THE DSCB FORMAT 1
        11.. ....             CRLRECFM     RECORD FORMAT
                                            10 - FIXED
                                            01 - VARIABLE
                                            11 - UNDEFINED
        ..1. ....             CRLRECTO     TRACE OVERFLOW
        ...1 ....             CRLRECFB     BLOCKED
                                            1 = BLOCKED
                                            0 = UNBLOCKED
        .... 1...             CRLRECFS     FOR FIXED LENGTH RECORD FORMAT -
                                            STANDARD BLOCKS. FOR VARIABLE LENGTH
                                            RECORD FORMAT - SPANNED RECORDS.
        .... .11.             CRLRECCC     CARRIAGE CONTROL
                                            10 - ANSI CARRIAGE CONTROL
                                            01 - MACHINE CARRIAGE CONTROL
        .... ...1             *            RESERVED
  309  (135) BITSTRING     1   CRLLIBTYP    DATA SET TYPE FLAG
                                            ONLY ONE BIT WILL BE SET
        1... ....             CRLPDSTYP    PARTITIONED DATA SET
        .1.. ....             CRLHFSTYP    UNIX FILE DATA SET
        ..11 1111             *            RESERVED
  310  (136) CHARACTER     7   *            RESERVED
  317  (13D) CHARACTER     0   CRLVOLS      VOLUME LIST
  317  (13D) CHARACTER     6   CRLAVOLS     VOLUME SERIAL NUMBERS. INDEX BY CRLNVOL.
                                            X'FF' IN FIRST BYTE OF ENTRY INDICATES
                                            VOLUME SERIAL NOT DETERMINED. ONE VOLUME
                                            FOR PARTITIONED DATA SET
                                            DOMAIN: 1-6 ALPHANUMERIC CHARACTERS
```

Figure 167. Format of the CRL data area

APSLCRL fields

CRLPFIX
CRL prefix section.

CRLID
Contains the control block identifier CRL.

CRLLEN
Contains the length of the CRL, including the data set entries.

CRLPTF
Contains the date of the PTF.

CRLREL
Contains the release level.

CRLEID
Contains an encoded object ID.

CRLSPN
Contains the subpool number.

CRLFLGO
Contains flags.

CRLBASE
CRL base section.

CRLDDNM
Contains the DDNAME for the data set entries.

CRLRTRVD
Contains the number of data set entries; if no data set names are retrieved, it is set to 0.

CRLCONC
Contains the number of concatenated data sets; if this is not a concatenated data set, CRLCONC is set to 1.

CRLLOWCN
Contains the lowest concatenation number for this data set mapping.

CRLHIGCN
Contains the highest concatenation number for this data set mapping.

CRLEND
End of the CRL base section; can be used as addressability for CRLDATA.

CRL data area

CRLDATA
This area contains the data set names and volume entries. The number of entries is equal to the value in CRLRTRVD.

CRLSDELN
Contains the length of the data set and volume serial list. This field is used to increment to the next entry mapped by CRLDATA.

CRLCATN
Contains the concatenation number for the data set name in CRLDSNM. This number is supplied to the exit in the RLSTPDS, RLSTCPDS, or RLSTFNDS fields during the resource load calls.

CRLNVOL
Contains the number of volumes in CRLVOLS; can be used as a maximum index for CRLAVOLS.

CRLDSNM
Contains the name of the partitioned data set. A X'40' in the first byte indicates that the data set cannot be retrieved.

CRLPNM
Contains the UNIX file system path name.

CRLPNMLEN
Contains the length of the UNIX file system path name.

CRLRFORM
Contains the record attributes for this data set. The attributes are from the Data Set Control Block (DSCB) format 1, field DS1RECFM.

CRLLIBTYP
Contains the data set type flag.

CRLVOLS
Volume list.

CRLAVOLS
An array that contains the volume serial numbers. CRLNVOL contains the number of elements to this array. A X'FF' in the first byte indicates that PSF is unable to determine the volume serial number.

APSURLST parameter area (resource list)

APSURLST is a control block that PSF builds when it needs to provide information to the exit about a specific resource. Typically, PSF builds one RLST entry for each resource associated with the current call to the exit. If multiple RLST entries are required to describe all the resources to the exit, the RLST entries are linked to form a single-headed, double-threaded chain. The first, or only, RLST entry is pointed to by XTP7LSTP in the XTP7 control block area.

Note: Exit 7 must never modify the pointers RLSTPRVS, RLSTNEXT, and RLSTEXTP as defined in APSURLST.

All the fields in APSURLST are read by the exit; however, some fields contain information that is valid only for certain exit calls and certain resource types. Also, most fields in APSURLST are not intended to be written to by the exit. PSF looks for changes to specific fields after certain calls to the exit. For more details, see "APSURLST fields" on page 443.

```
Offsets   Type      Length  Name          Description
0    (0) CHARACTER     *  APSURLST
0    (0) CHARACTER     4  RLSTCID       CONTROL BLOCK ID = RLST
4    (4) SIGNED        4  RLSTLENG      LENGTH OF RLST CONTROL BLOCK
8    (8) CHARACTER    16  RLSTPTRS      RESOURCE LIST POINTERS
8    (8) ADDRESS       4  RLSTPRVS      POINTER TO PREVIOUS ENTRY
12   (C) ADDRESS       4  RLSTNEXT      POINTER TO NEXT ENTRY
16  (10) ADDRESS       4  RLSTEXTP      POINTER TO FONT EXTENSION AREA
20  (14) ADDRESS       4  RLSTMSGP      POINTER TO 64 BYTE MESSAGE
                                          INSERT TEXT
24  (18) CHARACTER    16  RLSTRID       RESOURCE IDENTIFICATION
24  (18) CHARACTER     8  RLSTNAME      RESOURCE NAME
24  (18) BITSTRING     2  RLSTGCID      GRAPHIC CHARACTER SET
                                        GLOBAL IDENTIFIER GCSGID
26  (1A) BITSTRING     2  RLSTCPID      CODE PAGE GLOBAL IDENTIFIER
28  (1C) BITSTRING     2  RLSTFGID      FONT GLOBAL IDENTIFIER
30  (1E) BITSTRING     2  RLSTFW        FONT WIDTH
32  (20) UNSIGNED      1  RLSTPDS       PDS NUMBER FOR RESOURCE
                                        IDENTIFIED IN RLSTNAME FIELD
33  (21) UNSIGNED      1  *             RESERVED
34  (22) BITSTRING     2  RLSTATTR      RESOURCE ATTRIBUTES
     1... ....            RLSTDFT       DEFAULT PAGEDEF/FORMDEF
     .1.. ....            *             RESERVED
     ..1. ....            RLSTINL       INLINE RESOURCE
     ...1 ....            RLSTSPS       SOFT PAGE SEGMENT
     .... 1...            RLSTMEF       MULTIPLE ENTRY FONT
     .... .1..            RLSTSEC       SECURITY RESOURCE
     .... ..1.            RLSTUSER      USERLIB RESOURCE
     .... ...1            RLSTSOFT      SOFT RESOURCE - CAN BE PSEG
                                        OR DOR (OBJECT CONTAINER OR
                                        IOCA); NOT VALID ON ACCESS
                                        CALL
     1111 1111            *             RESERVED
36  (24) BITSTRING     2  RLSTSTAT      RESOURCE STATUS FLAGS
     1... ....            RLSTAUX       AUXILIARY RESOURCE
     .1.. ....            RLSTDEP       HAS DEPENDENCIES
     ..1. ....            RLSTNEST      NESTED RESOURCE
     ...1 ....            RLSTPLR       PRIORITY LEVEL RES.
     .... 1111            *             RESERVED
     1111 1111            *             RESERVED
38  (26) CHARACTER     2  *             RESERVED
40  (28) SIGNED        2  RLSTRESC      RESOURCE COUNT-THE 5TH
                                        RESOURCE IN THE RLST CHAIN
                                        WILL BE #5
42  (2A) UNSIGNED      1  RLSTMSGL      LENGTH OF USER'S MESSAGE
                                        INSERT POINTED AT BY RLSTMSGP
43  (2B) CHARACTER     1  *             RESERVED
44  (2C) CHARACTER     4  RLSTFLGS      FLAGS
44  (2C) BITSTRING     1  RLSTMISC      MISCELLANEOUS FLAGS
     1... ....            RLSTOFNT      PRINTER SUPPORTS
                                        OUTLINE FONTS
     .1.. ....            RLSTGRID      RESOURCE NAME IS A GRID
     ..1. ....            RLSTOIDR      REFERENCED BY OID
     ...1 ....            RLSTNAMR      REFERENCED BY NAME
     .... 1111            *             RESERVED
```

Figure 168. APSURLST parameter area format (Part 1 of 2)

```
      45  (2D) BITSTRING   1  RLSTAFLG      FLAGS USED ONLY AT ACCESS TIME
          1... ....           RLSTLOAD      FORCE LOAD (RELOAD)
          .1.. ....           RLSTSMSG      ISSUE SUB MESSAGE
          ..1. ....           RLSTRRI       RELOAD REQ. IGNORED (SET BY
                                            PSF)
          ...1 ....           RLSTEMSG      ISSUE SUBSTITUTION ERROR
                                            MESSAGE
          .... 1111           *             RESERVED
      46  (2E) BITSTRING   1  RLSTLFLG      FLAGS USED ONLY AT LOAD TIME
          1... ....           RLSTLCF       LOADING A CODED FONT
          .1.. ....           RLSTLCP       LOADING A CODE PAGE
          ..1. ....           RLSTLFN       LOADING A FONT CHARACTER SET
          ...1 1111           *             RESERVED
      47  (2F) BITSTRING   1  RLSTDFLG      FLAGS USED ONLY AT DELETE TIME
          1... ....           RLSTKEEP      KEEP THIS RESOURCE
          .111 1111           *             RESERVED
      48  (30) UNSIGNED    2  RLSTVFS       VERTICAL FONT SIZE
      50  (32) UNSIGNED    2  RLSTHFS       HORIZONTAL FONT SIZE
      52  (34) UNSIGNED    2  RLSTHSCF      HORIZONTAL SCALE FACTOR
                                            SPECIFIED IN 1440THS OF AN
                                            INCH
      54  (36) UNSIGNED    1  RLSTOIDL      LENGTH OF OID
      55  (37) UNSIGNED  129  RLSTOID       THE OID DOR THIS RESOURCE
     184  (B8) CHARACTER   24  *            RESERVED
```

Figure 169. APSURLST parameter area format (Part 2 of 2)

APSURLST fields

RLSTCID
> Contains the control block identifier RLST.

RLSTLENG
> Contains the length of the RLST control block.

RLSTPTRS
> Pointers.
>
> > **RLSTPRVS**
> > The pointer to the previous RLST entry in the chain. If this is the first RLST entry, this variable is set to 0. The exit must never modify this field.
> >
> > **RLSTNEXT**
> > The pointer to the next RLST entry in the chain. If this is the last RLST entry, this variable is set to 0. The exit must never modify this field.
> >
> > **RLSTEXTP**
> > The pointer to the font extension area, RLSTFEXT. If a font is described as a code page and character set pair (sometimes abbreviated as CP/FN pair), PSF builds the font extension area. If a FEXT does not exist, this field is 0. The exit must never modify this field.
> >
> > **RLSTMSGP**
> > The pointer to the message insert text. The exit can add up to 64 bytes of text to the end of PSF message APS563I. This field is used with FLSTMSGL and RLSTSMSG. For more information about the optional text for APS563I, see the descriptions of these fields. If the exit leaves this field at 0, no text is added to the message.

RLSTRID
> Contains information about the resource represented by the RLST entry. All fields contain information that PSF supplies to the exit.
>
> > **RLSTNAME**
> > The resource member name. For resources other than fonts, this name contains all 8 characters of the member name, for example, P106060. For coded fonts, this 8-byte field contains all 8 characters of the member name or the GRID. Flag RLSTGRID is set when the field is a GRID. The GRID contains 4 fields: RLSTGCID, RLSTCPID, RLSTFGID, and RLSTFW. Sometimes fonts are described not by this eight-character field, but by their code page and character set names. If the coded font name or GRID is not available, this field is set to null, and the code page and character set names are in the font extension area (see RLSTFEXT). The exit can modify this field during the

RAC if it wants to make a substitution of the resource. Substitutions must be name for name and GRID for GRID.

RLSTGCID
> The 2-byte graphic character set global identifier (GCSGID) in the GRID for this font.

RLSTCPID
> The 2-byte code page global identifier in the GRID for this font.

RLSTFGID
> The font global identifier in the GRID for this font.

RLSTFW
> The font width in the GRID for this font.

RLSTPDS
> The partitioned data set concatenation number. PSF supplies this information during both the resource-load call and the resource-delete call. This field is not valid for inline resources.
>
> For more information, see "CRLCATN" on page 441.
>
> If the resource is from the system library, RLSTPDS is the PDS concatenation number in the system library. If the resource is from the user library, RLSTPDS is the PDS concatenation number in the user library.

RLSTATTR
> Attributes of this resource.

RLSTDFT
> If set, indicates this is a default page definition or form definition.
>
> **Note:** This flag is set only for the page definition or form definition specified in the Printer Inventory or the PRINTDEV statement.

RLSTINL
> If set, indicates that this is an inline resource. This flag is only valid during the resource-access call (RAC).

RLSTSPS
> If set, indicates that this is a soft page segment. This flag is only valid during the resource-load call (RLC).

RLSTMEF
> If set, indicates that this is a multiple-entry font.

RLSTSEC
> If set, indicates that the resource is from a security library.

RLSTUSER
> If set, indicates that the resource is from a user library.

RLSTSOFT
> If set, indicates that the resource is a soft page segment, IOCA, or object container resource.

RLSTSTAT
> Contains information that affects the deletion status of the resource.

RLSTAUX
> If set, indicates that this is an auxiliary resource. PSF never deletes auxiliary resources during data set end processing. The exit cannot change the auxiliary status of a resource.

RLSTDEP
> If set to 1, indicates that this resource has dependencies; that is, it mapped (nested) another resource. Overlays are the only resource type that can nest another resource. When an overlay that has dependencies is loaded in the printer, a nested resource cannot be deleted. The exit cannot change the dependency status of a resource.

RLSTNEST
> If set to 1, indicates that this resource is mapped (nested) in another resource. When a resource that has dependencies is loaded in the printer, a nested resource cannot be deleted.

Fonts and hard page segments are the only resources that can be nested. The exit cannot change the nested status of a resource.

RLSTPLR

Indicates that this resource is a priority-level resource. All resources default to a non-PLR (RLSTPLR=0). By setting this field to 1, the exit can make the resource a PLR; this improves its chance of remaining in the printer at the end of the data set.

RLSTRESC

Field set by PSF to indicate the position of this resource in the RLST chain. If a resource requires more than one RLST entry (multiple-entry fonts), all RLST entries for that resource have the same RLSTRESC value. The value of RLSTRESC is never more than XTP7RESC.

RLSTMSGL

Indicates the length of the message insert text for APS563I. PSF always sets this field to 0. If the exit supplies a pointer to some text in RLSTMSGP and sets this field to the length of the text (64 maximum), PSF adds that text to the end of APS563I.

RLSTFLGS

Flags.

RLSTMISC

General flags

RLSTOFONT

Set by PSF to B'1' on entry to the exit when the attached printer supports outline fonts. For printers that do not support outline fonts, set to B'0'.

RLSTGRID

Set by PSF to B'1' when the RLSTNAME field contains a GRID for a coded font. Otherwise, set to B'0'.

RLSTOIDR

Set by PSF to B'1' to indicate that a data object resource is referenced by OID. This flag is only valid for data object resources (object containers or IOCA resources). When this flag is on, the RLSTOID field contains an OID.

RLSTNAMR

Set by PSF to B'1' to indicate that a data object resource is referenced by name. This flag is only valid for data object resources (object containers or IOCA resources). When this flag is on, the RLSTNAME field contains a valid name.

Note: RLSTOIDR and RLSTNAMR are mutually exclusive flags.

RLSTAFLG

Flags valid only during the resource-access call (RAC).

RLSTLOAD

Forces a load of this resource. PSF always sets this flag to 0. The exit can set it to 1 to force PSF to load this resource. If the resource is already loaded, PSF deletes the resource and then reloads it.

Note: If the resource concerned is a page definition or a form definition, the load that is involved is from DASD to PSF virtual storage. For other resources, setting this flag causes the resource to be reloaded into printer storage.

RLSTSMSG

Issues substitution message APS3503I for data object resources (object containers and IOCA resources) or substitution message APS563I for all other resource types. PSF always sets this flag to 0. If the exit requested a substitution of this resource and it sets RLSTSMSG to 1, PSF issues message APS3503I or message APS563I.

RLSTRRI

Set by PSF to B'1' to indicate that PSF is to ignore a load request (RLSTLOAD) for this resource.

RLSTEMSG

Issues substitution error message APS3504I if a resource substitution error occurs. PSF always sets this flag to 0. If the exit requested a substitution of this resource, it sets RLSTEMSG to 1, and if a problem exists in the substitution, PSF issues message APS3504I. For information about valid and not valid resource substitution, see "Considerations related to inline resources" on page 299.

RLSTLFLG

Flags valid only during the resource-load call (RLC).

RLSTLCF

Informational flag set by PSF to indicate that PSF is loading a coded font.

RLSTLCP

Informational flag set by PSF to indicate that PSF is loading a code page.

RLSTLFN

Informational flag set by PSF to indicate that PSF is loading a font character set.

RLSTDFLG

Flags valid only during the resource-delete call (RDC).

RLSTKEEP

Keep or delete this resource. Before PSF calls the exit for the RDC, it sets this flag in accordance with its current pruning algorithm. By modifying this field, the exit tells PSF which resources must remain loaded or be deleted. If the field is set to 0, PSF deletes this resource. If it is set to 1, PSF does not delete this resource.

Note: PSF is not always able to follow the RLSTKEEP flag when it is deleting resources.

RLSTVFS

The vertical size for this font, if a size is specified, can be set by the exit to scale an outline font. This is the recommended field for scaling an outline font.

RLSTHFS

The horizontal size for this font, if a size is specified. When the vertical font size is not specified, RLSTHFS can be changed to scale an outline font.

RLSTHSCF

The horizontal scale factor for this font, if one is specified, can be set by the exit to contract or expand an outline font horizontally. This is known as *anamorphic scaling*. This field is not needed when the exit changes the point of a font but the normal character shape is maintained.

RLSTOIDL

The length of the OID for this resource. If this resource is referenced by OID, RLSTOIDL and RLSTOID are set, and RLSTOIDR is set to B'1'.

RLSTOID

The value of the OID for this resource. If this resource is referenced by OID, RLSTOID and RLSTOIDL are set, and RLSTOIDR is set to B'1'.

RLSTFEXT parameter area (font extension area)

The font extension area is used when the code page and character set names are available for the font. The RLSTFEXT area consists of a header and an area, RLSTCPFM, that describes the font sections. RLSTCPFM is an array with a number of elements equal to the number of sections in the font. For information about defining fonts and the Map Coded Font (MCF) structured field, see *Advanced Function Presentation: Host Font Data Stream Reference*.

Figure 170 on page 447 shows the format of RLSTFEXT.

```
Offsets   Type     Length  Name          Description
0    (0) CHARACTER    *   RLSTFEXT      FONT RLST EXTENSION
0    (0) CHARACTER    4   RLSTHDR       HEADER
0    (0) SIGNED       2   RLSTEXTC      NUMBER OF FONT EXTENSION
                                        REPEATING GROUPS
2    (2) CHARACTER    2   *             RESERVED
4    (4) CHARACTER   28   RLSTCPFN      CP/FN PAIR REPEATING GROUPS
4    (4) SIGNED       2   RLSTSID       SECTION ID NUMBER
6    (6) CHARACTER    2   RLSTPDSP      PDS PAIR
6    (6) UNSIGNED     1   RLSTCPDS      PDS NUMBER FOR CODE PAGE
7    (7) UNSIGNED     1   RLSTFNDS      PDS NUMBER FOR CHARACTER SET
8    (8) CHARACTER   16   RLSTPAIR      FONT PAIR
8    (8) CHARACTER    8   RLSTCPNM      CODE PAGE NAME
16  (10) CHARACTER    8   RLSTFNNM      FONT CHARACTER SET NAME
24  (18) BITSTRING    1   RLSTPFLG      PAIR FLAGS
     1... ....            RLSTCPUS      USERLIB CODE PAGE
     .1.. ....            RLSTFNUS      USERLIB FONT CHAR SET
     ..1. ....            RLSTCPIL      INLINE CODE PAGE
     ...1 ....            RLSTFNIL      INLINE FONT CHAR SET
     .... 1111            *             RESERVED
25  (19) CHARACTER    7   *             RESERVED
```

Figure 170. Format of the RLSTFEXT parameter area

RLSTFEXT fields

RLSTHDR
Header of the font extension area.

> **RLSTEXTC**
> Extension count field. PSF sets this field equal to the number of elements in the RLSTCPFN array. For a single-byte font, there is never more more than one element in the array. The exit must never modify this field.

RLSTCPFN
An array in which each element describes a section of the font as a code page/character set pair.

> **RLSTSID**
> Section ID. PSF sets this field equal to the coded font section identifier from the MCF structured field.

> **RLSTPDSP**
> Contains the partitioned data set concatenation numbers for the code page and character set in this section. These numbers are available only during the resource-load call to the exit.

> > **RLSTCPDS**
> > Contains the PDS concatenation number for the code page. If the code page resides in the system library, RLSTCPDS is the PDS concatenation number in the system library. If the code page resides in the user library, RLSTCPDS is the PDS concatenation number in the user library.

> > **RLSTFNDS**
> > Contains the PDS concatenation number for the character set. If the character set resides in the system library, RLSTFNDS is the PDS concatenation number in the system library. If the character set resides in the user library, RLSTFNDS is the PDS concatenation number in the user library.

> **RLSTPAIR**
> Contains the code page and character set names for this section. These fields are similar to the RLSTNAME field.

> > **RLSTCPNM**
> > Contains the code page name for this section. During the resource-access call to the exit, a new code page name is substituted if the exit changes the name in this field.

> > **RLSTFNNM**
> > Contains the character set name for this section. During the resource-access call to the exit, a new character set is substituted if the exit changes the name in this field.

RLSTPFLG

    Contains flags about the code page and character set for this field.

    RLSTCPUS

        If set, indicates that the code page is from the user library.

    RLSTFNUS

        If set, indicates that the character set is from the user library.

    RLSTCPIL

        If set, indicates that the code page is inline.

    RLSTFNIL

        If set, indicates that the character set is inline.

Processing indicator cross-reference for Exit 7

For definitions of the fields in , see "XTP7 fields" on page 429 and "APSURLST fields" on page 443.

Table 42. Input and output summary for Exit 7 processing indicators. Input is marked I, and output is marked O. If the indicator is both, it is marked IO.

Exit 7 fields	INIT	BDSC	EDSC	RAC	RLC	RDC	TERM
XTP7ETYP	I	I	I	I	I	I	I
XTP7RTYP				I	I	I	
XTP7DSTY							
XTP7AUX				I	I		
XTP7PDS		I	I	I	I	I	
XTP7DSAT							
XTP7PDFT				I	I		
XTP7PJHD				I	I		
XTP7PJTR				I	I		
XTP7PDSH				I	I		
XTP7PMDS				I	I		
XTP7PLDS		I	I			I	
XTP7ACSE	IO	IO	IO				
XTP7IPCC	IO	IO	IO				
XTP7NOTY	IO	IO	IO	IO	IO	IO	
XTP7RFLG							
XTP7ETFF	IO	IO	IO				
XTP7TDS		O		O	O		
XTP7EFFH	IO	IO	IO				
XTP7HDF	IO	IO	IO	IO	IO	IO	IO
XTP7HQUE		O		O	O		
XTP7PRNS	IO						
XTP7PRND	IO						

Exit 7 fields	INIT	BDSC	EDSC	RAC	RLC	RDC	TERM
XTP7IPMF	IO	IO	IO				
XTP7IPOS	IO	IO	IO				
XTP7MTOF	IO						
XTP7RFNF	IO						
XTP7IPSU	IO						
XTP738MO	IO	IO	IO				
XTP7MDSD	IO	IO	IO				
XTP7C2SI	IO	IO	IO				
XTP7RRLV	IO	IO	IO	IO	IO	IO	
XTP7RLST				I	I	I	
XTP7OVLY				I	I		
XTP7PINF	I	I	I	I	I	I	I
XTP7MSGP and MSGL		O		O	O		
XTP7CRLF	IO						
XTP7CRF2	IO						
XTP7CRLP	I						
XTP7MFLG							
XTP7JHMO	IO	IO	IO				
XTP7JTMO	IO	IO	IO				
XTP7DHMO	IO	IO	IO				
XTP7MDMO	IO	IO	IO				
XTP7RSTR							IO
XTP7WIDE	IO	IO	IO				
XTP7IHIB	IO	IO	IO				
XTP7ENDC	IO						
XTP7ABRC							I
XTP7JOG	IO	IO	I				
XTP7FMT	IO	IO	IO				
XTP7CRF3	IO						
XTP7CRP2	I						
XTP7MFG1							
XTP7APSF	IO						
XTP7NSF	IO						
XTP7CLOS	IO						

Table 42. Input and output summary for Exit 7 processing indicators. Input is marked I, and output is marked O. If the indicator is both, it is marked IO. *(continued)*

Exit 7 fields	INIT	BDSC	EDSC	RAC	RLC	RDC	TERM
XTP7SDCH	IO	IO	IO				
XTP7CINR	IO	IO	IO				
XTP7RDSR	IO						
XTP7SUCM	IO	IO	IO				
XTP7PRMD	IO	IO					
XTP7DRES	I	I	I	I	I	I	I
XTP7BINS							
XTP7BINA	IO						
XTP7SIMA	IO						
XTP7DUPA	IO						
XTP7BINB	IO						
XTP7SIMB	IO						
XTP7DUPB	IO						
XTP7BINC	IO						
XTP7SIMC	IO						
XTP7DUPC	IO						
XTP7BIND	IO						
XTP7SIMD	IO						
XTP7DUPD	IO						
XTP7MFG2							
XTP7ASAP		IO					
XTP7SENS	IO						
XTP7IMSG	IO						
XTP7PJO		IO	IO				
XTP7CPPP	IO						
XTP7LCNT	IO						
XTP7HLCF	IO						
XTP7LMCM	IO						
RLSTPTRS							
RLSTPRVS				I	I	I	
RLSTNEXT				I	I	I	
RLSTEXTP				I	I	I	
RLSTMSGP				O			
RLSTRID				I	I	I	

Table 42. Input and output summary for Exit 7 processing indicators. Input is marked I, and output is marked O. If the indicator is both, it is marked IO. *(continued)*

Exit 7 fields	INIT	BDSC	EDSC	RAC	RLC	RDC	TERM
RLSTNAME				IO	I	I	
RLSTGCID				IO	I	I	
RLSTCPID				IO	I	I	
RLSTFGID				IO	I	I	
RLSTFW				IO	I	I	
RLSTPDS				I	I	I	
RLSTATTR				I	I	I	
RLSTSTAT							
RLSTAUX				I	I	I	
RLSTDEP				I	I	I	
RLSTNEST				I	I	I	
RLSTPLR				IO	IO	IO	
RLSTRESC				I	I	I	
RLSTMSGL				O			
RLSTFLGS							
RLSTMISC				I	I	I	
RLSTAFLG				IO			
RLSTLFLG					I		
RLSTDFLG						IO	
RLSTVFS				IO	IO	IO	
RLSTHFS				IO	IO	IO	
RLSTHSCF				IO	IO	IO	
RLSTOIDL				IO	IO	IO	
RLSTOID				IO	IO	IO	

Table 42. Input and output summary for Exit 7 processing indicators. Input is marked I, and output is marked O. If the indicator is both, it is marked IO. *(continued)*

Appendix G. APTRCONV resource conversion utility for VSE

APTRCONV is a utility program that resides in PSF. Its purpose is to convert resources so that VSE can use them.

You can run APTRCONV with JCL to convert an AFP resource in a partitioned data set to a VSE job stream. APTRCONV assembles and link-edits the resource into a VSE *library.sublibrary*.

Converting a resource

Figure 171 on page 453 shows an example of the APTRCONV control statements by which a PAGEDEF resource can be converted for use by VSE.

```
//USER1T  JOB (SYS10000),CLASS=A,MSGCLASS=T,
//    USER=USER1T,REGION=1M
//S1      EXEC PGM=APTRCONV,PARM=(P1075A0,PAGEDEF,TEST)
//RESC      DD DSN=SYS1.PAGEDEF(P1075A0),DISP=SHR
//SYSPCH    DD DSN=USER1T.CONVOUT,
//    DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120),
//    SPACE=(TRK,(5,5)),DISP=(,CATLG),
//    UNIT=3380,VOL=SER=TEMP00
```

Figure 171. Sample JCL for converting the PAGEDEF resource for use by VSE

The sample JCL in Figure 171 on page 453 uses the following parameters:

P1075A0
   Specifies the name of the resource converted.

PAGEDEF
   Specifies the VSE library in which the converted resource is to be cataloged.

TEST
   Specifies the VSE sublibrary in which the converted resource is to be cataloged.

   Note: The converted resource is entered in the VSE *library.sublibrary* under the same name as the z/OS resource.

When a job is being networked from JES to POWER®, a special POWER utility name must be used to route the job to the reader rather than to the punch. For example:

```
//OUTPUT1   OUTPUT DEST=xxx.PWR$JOB
//SYSPCH    DD SYSOUT=x,OUTPUT=*.OUTPUT1
```

Figure 172. Sample JCL to route job to reader

The sample JCL in Figure 172 on page 453 uses the following parameters:

xxx
   Specifies the POWER NJE name.

PWR$JOB
   Specifies a special utility to route the job to the VSE reader rather than to the punch.

x
   Specifies the z/OS installation-defined punch class; for example, *b*.

Note: With this utility, the converted resource job stream runs without operator intervention.

453

Appendix H. Reasonable resource loaded values (RRLV)

This information lists the default reasonable resource loaded values. (See also "Partial control of resource deletion" on page 311.)

Note: PSF deletes page definitions and form definitions from virtual storage, not from printer storage.

Table 43. Reasonable resource loaded values (RRLV)

Resource	Device	Default value
Page definitions	3800 3820 All other printers	6 6 6
Form definitions	3800 3820 All other printers	6 6 6
Overlays	3800 3820 All other printers	0^{26} 0^{26} 0^{26}
Page segments	3800 3820 All other printers	0 0 0
Coded fonts	3800 3800 printer with storage expansion feature 3820 All other printers	10 64 24 PRT[27]
Object containers	3800 3800 with storage expansion feature 3820 All other printers	0 0 0 200

---

[26] If an overlay is specified in the Printer Inventory or the PRINTDEV, this overlay is kept.

[27] PRT is derived from printer information (storage size). Six fonts are left in the printer storage for each 256 KB of pattern storage available—for example, 48 fonts for 2 MB.

Appendix I. Font conversion program, 240-pel to 300-pel (APSRCF30)

Some JCL jobs that convert fonts are provided with PSF. They are intended as samples; as provided, they might not be suitable for your requirements.

APSRCF30 is a font conversion utility program provided with PSF. The program uses your currently installed 240-pel fonts as input, converts (resizes) them to 300-pel fonts, and places the converted output into a partitioned data set. APSRCF30 uses 24-bit addressing.

Note: APSRCF30 converts 240-pel single-byte raster fonts to 300-pel raster fonts. It cannot convert 240-pel double-byte raster fonts. Higher resolution double-byte fonts are available with double-byte outline fonts in the AFP Font Collection or the z/OS Font Collection (see *IBM AFP Fonts: Font Summary for AFP Font Collection* or *z/OS Font Collection*). APSRCF30 converts a 240-pel single-byte raster font to a close approximation of the font at 300-pel resolution. However, IBM does not warrant the quality of the resulting character pattern when you are using this utility.

Understanding APSRCF30

If you install a 300-pel printer, you must have a 300-pel font library to use with the printer. Use the APSRCF30 font conversion utility program only if you do not have a 300-pel font library containing the fonts needed for your printer. For information about 300-pel fonts a printer supports, see the documentation provided with the printer.

Using APSRCF30

APSRCF30 is included as part of PSF. After PSF is installed, if you do not have a 300-pel font library, you can run APSRCF30 to convert 240-pel fonts into 300-pel fonts.

Note: The IBM Core interchange raster fonts are included in both 240-pel and 300-pel resolutions; therefore, you do not have to convert them. Outline fonts are resolution-independent and do not need to be converted.

Before you convert fonts with APSRCF30, you need to:

1. Define a new font library for 300-pel fonts, if one does not exist. SYS1.FONT300 is the recommended name for your 300-pel system font library.

   You can use the existing 240-pel font library as a model, and allocate the new 300-pel font library similarly. The 300-pel font library requires more space than the 240-pel font library.

2. Create a list of 240-pel fonts that are to be converted.

   Only the character set members whose file names start with C0 must be converted with APSRCF30. See . The sample JCL converts all members whose file names start with C0:

   - Coded font members whose file names start with X0 must not be used as input to APSRCF30.
   - Code page members whose file names start with T1 must not be used as input to APSRCF30.

3. Make any changes in the JCL for APSRCF30 that are needed for your installation. See .

   The JCL supplied with APSRCF30 must be changed to indicate the font member input required. The JCL uses the font member names for a few sample fonts as input to NAME on the PROC statement. You must specify the font member names you want by using NAME on the PROC statement. For example, in the JCL for APSRCF30, specify:

   ```
   C0L0AD10  EXEC  PROC=CON,NAME=C0L0AD10
   C0L0AD12  EXEC  PROC=CON,NAME=C0L0AD12
   ```

```
       ⋮
C0S0AP13   EXEC   PROC=CON,NAME=C0S0AP13
```

4. Submit the JCL.

After you convert fonts with APSRCF30, you need to:

1. Run APSRMARK to mark the fonts for resident use and capture.

2. Change the PSF startup procedure to use the new 300-pel font library.

3. Arrange for PSF to have access to the code pages and coded fonts that are not converted. Either copy the code pages and the coded fonts from the 240-pel library to the 300-pel library, or specify font library concatenation so that the 300-pel library is accessed before the 240-pel library. For example:

```
//FONT01    DD DSN=SYS1.FONT300,DISP=SHR
//          DD DSN=SYS1.FONTLIBB,DISP=SHR
```

4. If you specified both a 240-pel printer and a 300-pel printer in the same PSF startup procedure, create DD statements for each font library. The FONT01 DD statement contains only the 240 concatenation list, and the FONT02 DD statement contains the 300-pel library (concatenated ahead of the 240-pel libraries). The PRINTDEV statement for each printer must contain a separate reference to the correct FONT DD statement.

Error handling

APSRCF30 detects each of the errors in the following list, and places the return code (RC) for it in register 15. The return code prints on the screen when you are using the APSRCF30 EXEC. When an error occurs, the font-conversion program might create an incomplete output file, which you discard.

RC

  Meaning

4

One or more of the conversion character data set members that are required by the conversion program cannot be found. The DD of the required conversion character data set is FONTB300.

20

An I/O error occurred while APSRCF30 was writing to the output data set.

28

The input font data set either cannot be found or cannot be opened. This error can occur if SAPSDAT0 (the conversion character data set) was allocated with record format FBM rather than VBM.

32

The font that is named in the input data set is not valid for any of the following reasons:

- The member contains structured fields that are not valid.
- The font is not bounded-box.
- The font is not 240-pel.

44

An unexpected end-of-file occurred on the input data set.

88

Insufficient virtual storage was available for program processing.

99

Not valid font data. This return code is presented if any of the following structured fields in a font have a length less than 1:

  FNI

    Font index

  FNM

    Font patterns map

FNG
    Font patterns

APSRCF30 considerations

Do not convert the Bar Code|Optical Character Recognition font (Program Number 5688-021). The conversion process typically distorts the font, making it unreadable by an optical scanning device. Check all the JCL for this font before you run APSRCF30.

APSRCF30 uses a comparatively large amount of processor resources, especially if started on a smaller IBM Z processor complex. To run APSRCF30, you need a minimum region size of 5 MB.

Before you run APSRCF30, you allocate the target font library for the conversion.

For an example of JCL that can be changed, see "APSRCF30 sample jobs" on page 459.

APSRCF30 has no PARM statement; it uses the following ddnames:

- SYSUT1: Input font library (including member name); uses DSORG=PS.
- SYSUT2: Output font library (including member name); uses DSORG=PS.
- FONTB300: Conversion character data set, installed with PSF (used for conversion of special characters); uses DSORG=PO. The default name is SYS1.SAPSDAT0.

APSRCF30 sample jobs

Sample font conversion jobs are included with PSF in SYS1.SAMPLIB. You can change them to convert bounded-box fonts from 240-pel fixed metrics to 300-pel relative metrics by using a PROC invocation of APSRCF30. The PROC statements that are supplied are provided as samples to assist you in tailoring JCL to reflect the bounded-box fonts that require converting.

Allocate the font library used by the PROC before you run APSRCF30.

You can tailor APSWAL30, which is in SYS1.SAMPLIB, to convert all the 240-pel fonts in your library to 300-pel fonts. The sample JCL for APSWAL30 is given in "Converting all 240-Pel fonts in source library (APSRAL30)" on page 459.

If you select some of your 240-pel fonts to be converted, rather than converting all the fonts in your source library, see "Example for selected font conversion" on page 461 for sample JCL (APSWCF30) that you can modify and use when you tailor the JCL from your source library or from the individual sample jobs in SYS1.SAMPLIB. The JCL you use might differ from the example.

Converting all 240-Pel fonts in source library (APSRAL30)

You can change the following sample JCL and use it to convert all the font character sets (all members with prefix C0) in your source library from 240-pel fixed metrics to 300-pel relative metrics.

```
//APSWAL30 JOB 'account #','name,MSGLEVEL=(1.1)
//***************************************************************
//* APSWAL30 SAMPLE JOB TO CREATE JCL FOR ALL PDS MEMBERS
//***************************************************************
//*
//NAME     EXEC PGM=APSRAL30
//*
//* SYSUT1 IS 240-PEL FONTS TO BE CONVERTED TO 300-PEL
//SYSUT1    DD DSN=SYS1.FONTLIBB,DISP=SHR
//*
//* SYSUT2 IS USER'S JCL IN PDS OR FILE SEQ OR DUMMY
//SYSUT2    DD DSN=SYS1.SAMPLIB(APSWALHD),DISP=SHR
//*
//* SYSUT3 IS JCL AND EXEC PROCEDURE CARDS, FILE SEQ OR PDS WITH MEMBER NAME
//SYSUT3    DD DSN=your.runjob.dataout(out),DISP=SHR
//*
```

The following inline procedure (CON) is referenced by SYSUT2, and is included in SYS1.SAMPLIB under the name APSWALHD. This JCL is not submitted; it is copied into the output data set specified in SYSUT3 by APSRAL30.

```
//APSWALHD JOB 'account #','name,MSGLEVEL=(1.1)
//*
//CON      PROC NAME=
//S1       EXEC PGM=APSRCF30,REGION=5M
//*                   INSERT YOUR INPUT 240-PEL FONT LIBRARY
//SYSUT1   DD DISP=SHR,DSN=SYS1.FONTLIBB(&NAME)
//*                   INSERT YOUR OUTPUT 300-PEL FONT LIBRARY
//SYSUT2   DD DISP=SHR,DSN=SYS1.FONT300(&NAME)
//*                   CONVERSION CHARACTER DATA SET DDNAME
//FONTB300 DD DISP=SHR,DSN=SYS1.APSDAT00
//         PEND
//*
//*
```

In the JCL to run APSRCF30, you must list every character set member that is to be converted. However, since the APSRAL30 utility is provided, you do not have to list the name of every character set to be converted. APSRAL30 creates the JCL for APSRCF30 (including the list of character set members) for a 240-pel font data set. To convert the font data set, submit the JCL created by APSRAL30.

Input to APSRAL30

SYSUT1
Specifies the 240-pel font data set to be converted.

SYSUT2
Specifies the data set containing the JCL that is to be copied into the APSRCF30 JCL. SYS1.SAMPLIB (APSWALHD) provides an example of the JCL required.

Output from APSRAL30

SYSUT3
Specifies the output data set. It contains the APSRCF30 JCL to convert every character set member in a font data set.

APSRAL30 processing

APSRAL30 copies the JCL in the data set specified by SYSUT2 into the SYSUT3 output data set. This is the JCL to run APSRCF30. APSRAL30 then reads the directory of the 240-pel font data set. It creates an entry in the SYSUT3 output data set for every character set found.

Note: A character set is identified by the prefix C0.

After APSRAL30 is run, the output data set contains the JCL to convert all the font character set members of a font data set. To convert the character sets, you must submit the output data set JCL. You can modify the output data set JCL before you submit it. For example, if you want to leave some of the font character sets unconverted, delete their member names from the output data set.

To run APSRAL30:

1. Allocate the output data set specified in SYSUT3.

   The data set must be a fixed-block data set with a record length of 80. It can be sequential or partitioned with a member name; the member name does not need to be predetermined.

2. Tailor the JCL in the data set specified by SYSUT2.

   A sample JCL file, APSWALHD, is provided in SYS.SAMPLIB.

3. Tailor the sample APSRAL30 JCL to your installation.

4. Submit APSRAL30.

5. Optionally, edit the APSRCF30 JCL created by APSRAL30.

6. Submit the APSRCF30 JCL created by APSRAL30.

7. Run APSRMARK to mark the fonts for resident use and capture.

8. Change the PSF startup procedure to use the new 300-pel font library.

9. Arrange for PSF to have access to the code pages and coded fonts that are not converted. Either copy the code pages and the coded fonts from the 240-pel library to the 300-pel library, or specify font library concatenation so that the 300-pel library is accessed before the 240-pel library. For example:

```
//FONT01    DD DSN=SYS1.FONT300,DISP=SHR
//          DD DSN=SYS1.FONTLIBB,DISP=SHR
```

10. If you specified both a 240-pel printer and a 300-pel printer in the same PSF startup procedure, create DD statements for each font library. The FONT01 DD statement contains only the 240 concatenation list, and the FONT02 DD statement contains the 300-pel library (concatenated ahead of the 240-pel libraries). The PRINTDEV statement for each printer must contain a separate reference to the correct FONT DD statement.

APSRAL30 considerations

Some considerations include:

- SYSUT1 cannot contain a concatenation list, because only the PDS directory is being processed.
- The JCL contained in the input data set SYSUT2 is output to your SYSUT3 data set. SYSUT2 can be a PDS with a member name or a sequential data set. It can also be a JCL DUMMY statement, in which case only the EXEC PROC statements are output to your SYSUT3 data set.
- The output on SYSUT3 contains the JCL from SYSUT2, followed by up to 255 EXEC PROC statements that represent the character sets in SYSUT1.

  If the input PDS directory on SYSUT1 has more than 255 members whose names begin with C0, SYSUT3 contains the SYSUT2 JCL repeated with up to another 255 EXEC PROC statements. This process continues until all the SYSUT1 PDS members whose names begin with C0 are processed. z/OS has a limitation of 255 EXEC statements in a job.

- You can edit the SYSUT3 output before you submit the job.
- The output of this program always refers to the inline procedure CON.

Example for selected font conversion

Using the format in , and updating the JCL as needed for the font you selected, you can convert font character sets from 240-pel fixed metrics to 300-pel relative metrics.

```
//APSWCF30 JOB 'account #','name',MSGLEVEL=(1,1)
//**********************************************************************
//* THIS IS A SAMPLE JOB USED TO CONVERT A GIVEN BOUNDED-BOX           *
//* CHARACTER SET FROM 240-PEL FIXED METRICS TO 300-PEL RELATIVE       *
//* METRICS, USING A PROC INVOCATION OF THE APSRCF30 FONT SCALING      *
//* PROGRAM.  THE SUPPLIED PROC STATEMENTS AND CHARACTER SETS ARE      *
//* PROVIDED AS SAMPLE ONLY, AND SHOULD BE TAILORED TO REFLECT THE     *
//* BOUNDED-BOX CHARACTER SETS THAT REQUIRE SCALING.                   *
//*                                                                    *
//* THE FONT LIBRARY USED BY THIS PROC SHOULD BE ALLOCATED BEFORE      *
//* THIS PROCEDURE IS INVOKED.                                         *
//*                                                                    *
//* NOTE: IF CONCATENATION OF FONT LIBRARIES WILL NOT BE PERFORMED     *
//*       WITH PSF, THE CODE PAGES AND CODED FONTS SHOULD BE COPIED    *
//*       TO THE 300-PEL FONT LIBRARY.                                 *
//*                                                                    *
//* DDNAME USAGE:                                                      *
//* ============                                                       *
//* SYSUT1   - INPUT FONT LIBRARY (MEMBER NAME REQUIRED BY PROC)       *
//* SYSUT2   - OUTPUT FONT LIBRARY (MEMBER NAME REQUIRED BY PROC)      *
//* FONTB300 - CONVERSION CHARACTER DATA SET                          *
//*                                                                    *
//**********************************************************************
//*
//CON       PROC NAME=
//S1        EXEC PGM=APSRCF30,REGION=5M
//*             INSERT YOUR INPUT 240-PEL FONT LIBRARY
//SYSUT1    DD DISP=SHR,DSN=SYS1.FONTLIBB(&NAME)
//*             INSERT YOUR OUTPUT 300-PEL FONT LIBRARY
//SYSUT2    DD DISP=SHR,DSN=SYS1.FONT300(&NAME)
//*             CONVERSION CHARACTER DATA SET DDNAME
//FONTB300  DD DISP=SHR,DSN=SYS1.SAPSDAT0
//          PEND
//*
//**********************************************************************
//* SAMPLE - CONVERT APL2 BOUNDED-BOX CHARACTER SET FROM 240-PEL       *
//*          FIXED METRICS TO 300-PEL RELATIVE METRICS                 *
//* PRODUCT NAME  : APL2                                               *
//* PRODUCT NUMBER: 5771-ADB                                           *
//* PARTLIST      : CBIPOADB PARTLIST                                  *
//**********************************************************************
//C0L0AD10  EXEC PROC=CON,NAME=C0L0AD10
//C0L0AD12  EXEC PROC=CON,NAME=C0L0AD12
//C0L0AG10  EXEC PROC=CON,NAME=C0L0AG10
//C0L0AG12  EXEC PROC=CON,NAME=C0L0AG12
//C0L0AG15  EXEC PROC=CON,NAME=C0L0AG15
//C0L0AI10  EXEC PROC=CON,NAME=C0L0AI10
//C0L0AI12  EXEC PROC=CON,NAME=C0L0AI12
//C0L0AT10  EXEC PROC=CON,NAME=C0L0AT10
//C0L0AT12  EXEC PROC=CON,NAME=C0L0AT12
//C0L00APL  EXEC PROC=CON,NAME=C0L00APL
//C0S0AP10  EXEC PROC=CON,NAME=C0S0AP10
//C0S0AP13  EXEC PROC=CON,NAME=C0S0AP13
//C0S0AP20  EXEC PROC=CON,NAME=C0S0AP20
```

Figure 173. JCL sample for selected font conversion

Appendix J. Installation verification program example

Figure 174 on page 463 is a sample of output obtained by running the installation verification program (IVP) on a 240-pel resolution printer.

Figure 174. IVP output printed with 240-pel resolution

Appendix K. PSF programming interfaces

Customers can use the interfaces that are identified in this information to write programs that use the services of PSF. Only the interfaces that are listed here are used to request or receive the services of PSF.

- PSF-related JES initialization statements
- PSF startup procedure
- Infoprint Server Printer Inventory
- Intended Programming Interfaces

  APSGEXTP/APSGEXTC
  APSLCRL/APSUCRLC
  APSXUCOM
  APSUBTBL
  APSUECA/APSUECA
  APSURLST/APSURLSC
  APSUXP16/APSUP16C

- Interfaces for Tailoring PSF

  APSRFTBL
  APSRGTNM
  APSRNTGM

- Installation-Wide Exits

  APSUX01/APSUC01
  APSUX02/APSUC02
  APSUX03/APSUC03
  APSUX04/APSUC04
  APSUX05/APSUC05
  APSUX06/APSUC06
  APSUX07/APSUC07
  APSUX08/APSUC08
  APSUX14/APSUC14
  APSUX15/APSUC15
  APSUX16/APSUC16

Appendix L. Mapping tables for outline fonts

This information contains two PSF tables:

- APSRGTNM, which maps global resource identifiers (GRIDs) to font names (for either raster fonts or outline fonts)
- APSRNTGM, which maps font names to GRIDs

For information about using these tables, see "Using the font mapping tables" on page 237.

The following fields are in the font mapping tables:

FGID
Font global identifier

FW
Font width (1440ths of an inch)

GCSGID
Graphic character set global identifier

FLAGS
Reserved

TEKID
The technology identifier for the font character set is:

X'05'
Raster

X'1F'
Single-byte outline

VFS
Vertical font size for the font character set (1440ths of an inch). To find the VFS, multiply the point size of the raster font, 1/72 of an inch, by 20.

FCS NAME
The member name of the code page or font character set

Table 44 on page 467 and Table 45 on page 467 give samples of the information in the tables.

Table 44. GRID-to-font-name table, APSRGTNM						
FGID	**GCSGID**	**FW**	**FLAGS**	**TEKID**	**VFS**	**FCS NAME**
416	1261	120	0	X'05'	200	C0420200
416	2039	120	0	X'05'	200	C0420000
416	2039	0	0	X'1F'	0	CZ4200
X'FFFF'	X'FFFF'	X'FFFF'	X'FF'	X'FF'	0	FFFFFFFF

Table 45. Font-name-to-GRID table, APSRNTGM						
FCS NAME	**FLAGS**	**TEKID**	**FGID**[28]	**GCSGID**	**FW**	**VFS**
C0D0GT10	0	X'05'	0040	0000	000	000
CZN200	0	X'1F'	2308	2039	0	0
C0N200B0	0	X'05'	2308	2039	80	240

---

[28] Specifying zeros (0) for FGID means do not map fonts.

Table 45. Font-name-to-GRID table, APSRNTGM (continued)						
FCS NAME	**FLAGS**	**TEKID**	**FGID**[28]	**GCSGID**	**FW**	**VFS**
FFFFFFFF	X'FF'	X'FF'	X'FFFF'	X'FFFF'	X'FFFF'	X'FFFF'

The sample in Figure 175 on page 468, Figure 176 on page 469, and Figure 177 on page 470 maps GRIDs to font names (for either raster fonts or outline fonts). Use this module to add mapping entries to PSF's default GRID-to-member-names tables, change entries, or disable entries.

```
APSRGTNM - GRID PARTS TO FONT MEMBER NAMES TABLES
*/****START OF SPECIFICATIONS*********************EXTPROC**************/
*/*                                                                  */
*/*       MODULE NAME: APSRGTNM                                      */
*/*                                                                  */
*/*       DESCRIPTIVE NAME: GRID-TO-NAMES USER MAPPING TABLE         */
*/*                                                                  */
*/*       FUNCTION: THIS MODULE PROVIDES THE ABILITY FOR THE USER    */
*/*                 TO ADD ENTRIES FOR CODE PAGE AND CHARACTER SET    */
*/*                 GRID-TO-MEMBER-NAME MAPPING.              @01C*/
*/*                                                                  */
*/*       NOTES:                                                     */
*/*          DEPENDENCIES: NONE                                      */
*/*          RESTRICTIONS: NONE                                      */
*/*          REGISTER CONVENTIONS: NONE                              */
*/*          PATCH LABEL: NONE                                       */
*/*                                                                  */
*/*       MODULE TYPE:  PROCEDURE                                    */
*/*          PROCESSOR: OS ASSEMBLER H                               */
*/*          ATTRIBUTES: NO EXECUTABLE CODE                          */
*/*                                                                  */
*/*       ENTRY POINT: APSRGTFN                                      */
*/*          PURPOSE:  THIS TABLE MAPS GRID PARTS (FGID,             */
*/*                    GCSGID, FW) AND VERTICAL FONT SIZE TO THEIR    */
*/*                    CORRESPONDING MEMBER NAMES.  EACH ENTRY ALSO   */
*/*                    HAS A TECHNOLOGY ID ASSOCIATED WITH IT. THE    */
*/*                    TABLE IS TERMINATED BY A STRING OF XL2'FFFF'.  */
*/*                                                           @01C*/
*/*                    THE FORMAT OF EACH ENTRY IS AS FOLLOWS.        */
*/*                    BYTES  0-1  FONT GLOBAL ID (FGID)             */
*/*                    BYTES  2-3  GRAPHIC CHARACTER SET GLOBAL ID    */
*/*                                (GCSGID)                   @01C*/
*/*                    BYTES  4-5  FONT WIDTH                        */
*/*                    BYTES   6   FLAGS - RESERVED, MUST BE ZERO @01A*/
*/*                    BYTES   7   TECHNOLOGY ID                     */
*/*                                VALID TECHNOLOGY IDS ARE:         */
*/*                                X'05' RASTER (SINGLE-BYTE)        */
*/*                                X'1F' TYPE 1 OUTLINE (SINGLE-BYTE) */
*/*                    BYTES  8-9  VERTICAL FONT SIZE                */
*/*                    BYTES 10-17 FONT CHARACTER SET NAME           */
*/*          LINKAGE:  NONE                                          */
*/*                                                                  */
*/*       ENTRY POINT: APSRGTCP                                      */
*/*          PURPOSE:  THIS TABLE MAPS GRID PARTS (CPGID        @01C*/
*/*                    AND GCSGID) TO THEIR CORRESPONDING CODE PAGE   */
*/*                    MEMBER NAMES.  THE TABLE IS TERMINATED BY A    */
*/*                    STRING OF XL2'FFFF'.  THE FORMAT OF EACH       */
*/*                    ENTRY IS AS FOLLOWS.                          */
*/*                    BYTES  0-1  CODE PAGE GLOBAL ID (CPGID)        */
*/*                    BYTES  2-3  CODE PAGE GRAPHIC CHARACTER SET    */
*/*                                GLOBAL ID (GCSGID)                */
*/*                    BYTES  4-11 CODE PAGE NAME                    */
*/*          LINKAGE:  NONE                                          */
*/*                                                                  */
*/*       MESSAGES: NONE                                             */
*/*                                                                  */
```

Figure 175. GRIDs mapped to font names (Part 1 of 3)

```
*/*       CHANGE ACTIVITY                                           */
*/*                                                                 */
*/* $00=LASP0006,HPRF220, 930903, BJ13ATF: RELEASE 2.2.0      @00A*/
*/* $01=OW08340, HPRF220, 941020, BJ13DDA: Outline Font D-APAR @01A*/
*/* $DU=LAPS0007,HPRF310,980604,BDKURLB: Version 3.1.0        @DUA*/
*/* $DX=LAPS0008,HPRF320,991117,BUQ4RLB: Version 3.2.0        @DXA*/
*/* $EV=LAPS0009,HPRF330,010530,BUQ4RLB: Version 3.3.0        @EVA*/
*/* $FC=LAPS0010,HPRF340,020612,BDKULD: Version 3.4.0         @FCA*/
*/* $FM=LAPS0011,HPRF410,040220,BDKURDD: Version 4.1.0        @FMA*/
*/* $FU=LAPS0012,HPRF420,060630,BDKURDD: Version 4.2.0        @FUA*/
*/* $GD=LAPS0013,HPRF430,080514,BDKURDD: Version 4.3.0        @GDA*/
*/*                                                                 */
*/*********************END OF SPECIFICATIONS*************************/
*/*****************************************************************/
*/*                                                                 */
*/*                    BAL CODE FOLLOWS                             */
*/*                                                                 */
*/*****************************************************************/
APSRGTNM CSECT              ENTRY POINT OF THIS MODULE
APSRGTNM AMODE 31           DO NOT REMOVE/CHANGE
APSRGTNM RMODE ANY          DO NOT REMOVE/CHANGE
GTFN@    DC    V(APSRGTFN)  DO NOT REMOVE/CHANGE
GTCP@    DC    V(APSRGTCP)  DO NOT REMOVE/CHANGE
*****************************************************************
* THE FOLLOWING IS THE GRID-PARTS-TO-CHARACTER-SET TABLE.      *
*****************************************************************
         ENTRY APSRGTFN     DO NOT REMOVE/CHANGE
APSRGTFN DS    0D           DO NOT REMOVE/CHANGE
*****************************************************************
**THE FOLLOWING 6 LINES ARE FOR FGID 2308 (X'0904').          *
**THIS ENTRY PROVIDED TO ILLUSTRATE THE FORMAT OF A TABLE ENTRY. *
*****************************************************************
**       DC   H'2308'       FGID AS DECIMAL NUMBER
**                          (OR USE " DC XL2'NNNN' " FOR HEX NUM)
**       DC   H'2039'       GCSGID AS DECIMAL NUMBER
**                          (OR USE " DC XL2'NNNN' " FOR HEX NUM)
**       DC   H'0080'       FONT WIDTH AS DECIMAL NUMBER
**                          (OR USE " DC XL2'NNNN' " FOR HEX NUM)
**       DC XL1'00'         RESERVED, MUST BE ZERO.
**       DC XL1'05'         TECHNOLOGY ID
**       DC   H'0240'       VERTICAL FONT SIZE AS DECIMAL NUMBER
**                          (OR USE " DC XL2'NNNN' " FOR HEX NUM)
**       DC CL8'C0N200B0'   FONT CHARACTER SET NAME
*****************************************************************
**THE FOLLOWING 6 LINES MARK THE END OF APSRFNTG.             *
*****************************************************************
         DC XL2'FFFF'       DO NOT REMOVE/CHANGE
         DC XL2'FFFF'       DO NOT REMOVE/CHANGE
         DC XL2'FFFF'       DO NOT REMOVE/CHANGE
         DC XL1'FF'         DO NOT REMOVE/CHANGE
         DC XL1'FF'         DO NOT REMOVE/CHANGE
         DC XL2'FFFF'       DO NOT REMOVE/CHANGE
         DC CL8'FFFFFFFF'   DO NOT REMOVE/CHANGE
*****************************************************************
* THE FOLLOWING IS THE CODE-PAGE-TO-GRID PARTS TABLE.         *
*****************************************************************
         ENTRY APSRGTCP     DO NOT REMOVE/CHANGE
APSRGTCP DS    0D           DO NOT REMOVE/CHANGE
*****************************************************************
**THE FOLLOWING 3 LINES ARE FOR CPGID 500 (X'01F4').         *
**THIS ENTRY PROVIDED TO ILLUSTRATE THE FORMAT OF A TABLE ENTRY. *
*****************************************************************
**       DC   H'0500'       CPGID AS DECIMAL NUMBER
**                          (OR USE " DC XL2'NNNN' " FOR HEX NUM)
```

Figure 176. GRIDs mapped to font names (Part 2 of 3)

```
**       DC   H'0697'         CODE PAGE GCSGID AS DECIAML NUMBER
**                            (OR USE " DC XL2'NNNN' " FOR HEX NUM)
**       DC CL8'T1V10500'    CODE PAGE NAME
******************************************************************
**THE FOLLOWING 3 LINES MARK THE END OF APSRCPTG.              *
******************************************************************
         DC XL2'FFFF'         DO NOT REMOVE/CHANGE
         DC XL2'FFFF'         DO NOT REMOVE/CHANGE
         DC CL8'FFFFFFFF'    DO NOT REMOVE/CHANGE
******************************************************************
**THE FOLLOWING LINE MARKS THE END OF APSRGTNM.               *
******************************************************************
         END   APSRGTNM       DO NOT REMOVE/CHANGE
```

Figure 177. GRIDs mapped to font names (Part 3 of 3)

The sample in Figure 178 on page 470, Figure 179 on page 471, and Figure 180 on page 472 maps font names to GRIDs. Use this module to add mapping entries to PSF's default member-names-to-GRID tables, change entries, or disable entries.

```
APSRNTGM - FONT MEMBER NAMES TO GRID PARTS TABLES
*/****START OF SPECIFICATIONS********************EXTPROC**************/
*/*                                                                 */
*/*      MODULE NAME: APSRNTGM                                      */
*/*                                                                 */
*/*      DESCRIPTIVE NAME: NAMES-TO-GRID USER MAPPING TABLE         */
*/*                                                                 */
*/*      FUNCTION: THIS TABLES PROVIDES THE ABILITY FOR THE USER    */
*/*                TO ADD ENTRIES FOR CODE PAGE AND CHARACTER SET    */
*/*                MEMBER NAME-TO-GRID MAPPING.             @01C*/
*/*                                                                 */
*/*      NOTES:                                                     */
*/*         DEPENDENCIES: NONE                                      */
*/*         RESTRICTIONS: NONE                                      */
*/*         REGISTER CONVENTIONS: NONE                              */
*/*         PATCH LABEL: NONE                                       */
*/*                                                                 */
*/*      MODULE TYPE:  PROCEDURE                                    */
*/*         PROCESSOR: OS ASSEMBLER H                               */
*/*         MODULE SIZE: 900                                        */
*/*         ATTRIBUTES: READ ONLY, NO EXECUTABLE CODE               */
*/*                                                                 */
*/*      ENTRY POINT: APSRFNTG                                      */
*/*         PURPOSE:  THIS TABLE MAPS FONT CHARACTER SET            */
*/*                   MEMBER NAMES TO THE CORRESPONDING GRID PARTS  */
*/*                   (FGID, GCSGID, FW) AND A VERTICAL FONT SIZE.  */
*/*                   A TECHNOLOGY ID IS ALSO ASSOCIATED WITH EACH  */
*/*                   ENTRY.                                 @01C*/
*/*                   THE FORMAT OF EACH ENTRY IS AS FOLLOWS:       */
*/*                   BYTES  0-7  FONT CHARACTER SET MEMBER NAME    */
*/*                   BYTE    8   FLAGS -RESERVED, MUST BE ZERO. @01A*/
*/*                   BYTE    9   TECHNOLOGY ID                     */
*/*                               VALID TECHNOLOGY IDS ARE:         */
*/*                               X'05'  RASTER (SINGLE-BYTE)       */
*/*                               X'1F'  TYPE 1 OUTLINE (SINGLE-BYTE)*/
*/*                   BYTES 10-11 FONT GLOBAL ID (FGID)             */
*/*                   BYTES 12-13 GRAPHIC CHARACTER SET GLOBAL ID   */
*/*                               (GCSGID)                   @01C*/
*/*                   BYTES 14-15 FONT WIDTH                        */
```

Figure 178. Font names mapped to GRIDs (Part 1 of 3)

```
*/*                    BYTES 16-17 VERTICAL FONT SIZE                */
*/*                    THE TABLE IS TERMINATED BY A STRING OF         */
*/*                    CL8'FFFFFFFF'.                                 */
*/*         LINKAGE:  NONE                                            */
*/*      ENTRY POINT: APSRCPTG                                        */
*/*         PURPOSE:  THIS TABLE MAPS CODE PAGES TO                   */
*/*                   THE CORRESPONDING GRID PARTS(CPGID AND GCSGID).*/
*/*                   THE FORMAT OF EACH ENTRY IS AS FOLLOWS          */
*/*                   BYTES  0-7  CODE PAGE NAME                      */
*/*                   BYTES  8-9  CODE PAGE GLOBAL ID (CPGID)         */
*/*                   BYTES 10-11 CODE PAGE GRAPHIC CHARACTER SET     */
*/*                        GLOBAL ID (GCSGID)                         */
*/*                   THE TABLE IS TERMINATED BY A STRING OF          */
*/*                   CL8'FFFFFFFF'.                                  */
*/*         LINKAGE:  NONE                                            */
*/*                                                                   */
*/*      MESSAGES: NONE                                               */
*/*                                                                   */
*/*      CHANGE ACTIVITY                                              */
*/*                                                                   */
*/* $00=LAPS0006,HPRF220, 090393, BJ13ATF: RELEASE 2.2.0      @00A*/
*/* $01=OW08340, HPRF220, 102094, BJ13DDA: Outline Font D-APAR   @01A*/
*/* $DU=LAPS0007,HPRF310,980604,BDKURLB: Version 3.1.0          @DUA*/
*/* $DX=LAPS0008,HPRF320,991117,BUQ4RLB: Version 3.2.0          @DXA*/
*/* $EV=LAPS0009,HPRF330,010530,BUQ4RLB: Version 3.3.0          @EVA*/
*/* $FC=LAPS0010,HPRF340,020612,BDKULD : Version 3.4.0          @FCA*/
*/* $FM=LAPS0011,HPRF410,040220,BDKURDD: Version 4.1.0          @FMA*/
*/* $FU=LAPS0012,HPRF420,060630,BDKURDD: Version 4.2.0          @FUA*/
*/* $GD=LAPS0013,HPRF430,080514,BDKURDD: Version 4.3.0          @GDA*/
*/*                                                                   */
*/***********************END OF SPECIFICATIONS************************/
*/*******************************************************************/
*/*                                                                   */
*/*                      BAL CODE FOLLOWS                             */
*/*                                                                   */
*/*******************************************************************/
APSRNTGM CSECT             ENTRY POINT OF THIS MODULE
APSRNTGM AMODE 31          DO NOT REMOVE/CHANGE
APSRNTGM RMODE ANY         DO NOT REMOVE/CHANGE
FNTG@    DC    V(APSRFNTG)  DO NOT REMOVE/CHANGE
CPTG@    DC    V(APSRCPTG)  DO NOT REMOVE/CHANGE
*****************************************************************
* THE FOLLOWING IS THE CHARACTER-SET-TO-GRID-PARTS TABLE.       *
*****************************************************************
         ENTRY APSRFNTG    DO NOT REMOVE/CHANGE
APSRFNTG DS    0D          DO NOT REMOVE/CHANGE
*****************************************************************
**CORE TIMES NEW ROMAN - MEDIUM - RASTER 12 POINT              *
**THE FOLLOWING 6 LINES ARE FOR LATIN.  THIS ENTRY IS PROVIDED *
**TO ILLUSTRATE THE FORMAT OF EACH ENTRY.                      *
*****************************************************************
**       DC CL8'C0N200B0'   FONT CHARACTER SET NAME
**       DC XL1'00'         RESERVED, MUST BE ZERO
**       DC XL1'05'         TECHNOLOGY ID
**       DC   H'2308'       FGID AS DECIMAL NUMBER
**                          (OR USE " DC XL2'NNNN' " FOR HEX NUM)
**       DC   H'2039'       GCSGID AS DECIMAL NUMBER
**                          (OR USE " DC XL2'NNNN' " FOR HEX NUM)
**       DC   H'0080'       FONT WIDTH AS DECIMAL NUMBER
**                          (OR USE " DC XL2'NNNN' " FOR HEX NUM)
**       DC   H'0240'       VERTICAL FONT SIZE AS DECIMAL NUMBER
**                          (OR USE " DC XL2'NNNN' " FOR HEX NUM)
```

Figure 179. Font names mapped to GRIDs (Part 2 of 3)

```
        ******************************************************************
        **CORE TIMES NEW ROMAN - MEDIUM - OUTLINE FONT              *
        **THE FOLLOWING 6 LINES ARE FOR LATIN.  THIS ENTRY IS PROVIDED   *
        **TO ILLUSTRATE THE FORMAT OF EACH ENTRY.                    *
        ******************************************************************
        **      DC CL8'CZN200  '    FONT CHARACTER SET NAME
        **      DC XL1'00'          RESERVED, MUST BE ZERO
        **      DC XL1'1F'          TECHNOLOGY ID
        **      DC  H'2308'         FGID AS DECIMAL NUMBER
        **                          (OR USE " DC XL2'NNNN' " FOR HEX NUM)
        **      DC  H'1269'         GCSGID AS DECIMAL NUMBER
        **                          (OR USE " DC XL2'NNNN' " FOR HEX NUM)
        **      DC  H'0000'         FONT WIDTH AS DECIMAL NUMBER
        **                          (OR USE " DC XL2'NNNN' " FOR HEX NUM)
        **      DC  H'0000'         VERTICAL FONT SIZE AS DECIMAL NUMBER
        **                          (OR USE " DC XL2'NNNN' " FOR HEX NUM)
        ******************************************************************
        **THE FOLLOWING 6 LINES MARK THE END OF APSRFNTG.           *
        ******************************************************************
                DC CL8'FFFFFFFF'   DO NOT REMOVE/CHANGE
                DC XL1'FF'         DO NOT REMOVE/CHANGE
                DC XL1'FF'         DO NOT REMOVE/CHANGE
                DC XL2'FFFF'       DO NOT REMOVE/CHANGE
                DC XL2'FFFF'       DO NOT REMOVE/CHANGE
                DC XL2'FFFF'       DO NOT REMOVE/CHANGE
                DC XL2'FFFF'       DO NOT REMOVE/CHANGE
        *******************************************************************
        * THE FOLLOWING IS THE CODE-PAGE-TO-GRID PARTS TABLE.            *
        *******************************************************************
                ENTRY APSRCPTG     DO NOT REMOVE/CHANGE
        APSRCPTG DS    0D          DO NOT REMOVE/CHANGE
        ******************************************************************
        **THE FOLLOWING 3 LINES ARE FOR EBCDIC INTERNATIONAL CODE PAGE   *
        **THIS ENTRY IS PROVIDED TO ILLUSTRATE THE FORMAT OF ONE ENTRY.  *
        ******************************************************************
        **      DC CL8'T1V10500'    CODE PAGE NAME
        **      DC  H'0500'         CPGID AS DECIMAL NUMBER
        **                          (OR USE " DC XL2'NNNN' " FOR HEX NUM)
        **      DC  H'0697'         CODE PAGE GCSGID AS DECIMAL NUMBER
        **                          (OR USE " DC XL2'NNNN' " FOR HEX NUM)
        ******************************************************************
        **THE FOLLOWING 3 LINES MARK THE END OF APSRCPTG.           *
        ******************************************************************
                DC CL8'FFFFFFFF'   DO NOT REMOVE/CHANGE
                DC XL2'FFFF'       DO NOT REMOVE/CHANGE
                DC XL2'FFFF'       DO NOT REMOVE/CHANGE
        ******************************************************************
        **THE FOLLOWING LINE MARKS THE END OF APSRNTGM.            *
        ******************************************************************
                END   APSRNTGM     DO NOT REMOVE/CHANGE
```

Figure 180. Font names mapped to GRIDs (Part 3 of 3)

Appendix M. Installing the i-data 7913 IPDS Printer LAN Attachment

This information describes how to install and configure an i-data 7913 IPDS Printer LAN Attachment.

Note: You must configure an i-data 7913 IPDS Printer LAN Attachment on an AIX system on which the **bootp** daemon program is running.

For information about the i-data 7913 IPDS Printer LAN Attachment support for your printer, see the documentation provided with the printer.

One i-data 7913 IPDS Printer LAN Attachment is required for each printer you install, and each 7913 is installed and configured separately. The tasks of installation and configuration are:

- Understand how the i-data 7913 IPDS Printer LAN Attachment relates to PSF.
- Check prerequisites for the i-data 7913 IPDS Printer LAN Attachment.
- Configure the **bootp** daemon.
- Set up the **bootp** daemon on AIX.
- Configure a twinaxial printer.
- Set the controls for the i-data 7913 IPDS Printer LAN Attachment.
- Attach and start the i-data 7913 IPDS Printer LAN Attachment.
- Test the i-data 7913 IPDS Printer LAN Attachment.
- Troubleshoot the i-data 7913 IPDS Printer LAN Attachment.

For more information, and for instructions for downloading corrective service 7913 microcode, see *i-data 7913 IPDS Printer LAN Attachment for Token Ring Installation Guide* or *i-data 7913 IPDS Printer LAN Attachment for Ethernet Installation Guide*.

Understanding how the i-data 7913 IPDS Printer LAN Attachment works

The i-data 7913 IPDS Printer LAN Attachment is a hardware protocol converter that you can use to communicate between PSF and a coaxial or twinaxial IPDS printer.

The i-data 7913 IPDS Printer LAN Attachment requires TCP/IP communications protocols, and runs on either an IBM token-ring network or an Ethernet LAN. The i-data 7913 IPDS Printer LAN Attachment must be configured on an AIX system that is running the TCP/IP **bootp** daemon program. The **bootp** daemon controls the 7913 initialization process.
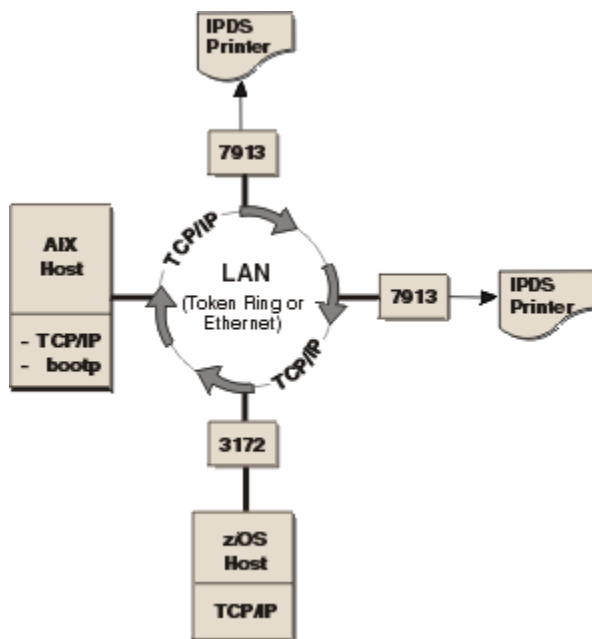
Figure 181. i-data 7913 IPDS Printer LAN Attachment in a TCP/IP LAN environment

shows a sample TCP/IP LAN configuration, with an AIX host that is running TCP/IP and the TCP/IP **bootp** daemon. The z/OS host and the AIX host can be attached directly to the same LAN as the i-data 7913 IPDS Printer LAN Attachment, as shown in , or they can be attached to a different LAN connected through bridges, routers, and gateways.

Checking the i-data 7913 IPDS Printer LAN Attachment

Before you install the i-data 7913 IPDS Printer LAN Attachment, you must satisfy the following conditions:

- You must have one i-data 7913 IPDS Printer LAN Attachment for each printer you plan to install. You can order the i-data 7913 IPDS Printer LAN Attachment as a printer feature.
- If you are installing the i-data 7913 IPDS Printer LAN Attachment on an AIX system, you must first install the following AIX hardware:
  - IBM AIX 7.1.0 or later (Program Number 5765-G98)
  - The TCP/IP (**bosnet.tcpip.obj**) option of the AIX operating system
- You must have either an IBM token-ring network or an Ethernet LAN installed and operating correctly.
- Your AIX LAN must coexist with TCP/IP. If not, see *TCP/IP Tutorial and Technical Overview*, GG24-3376.
- You must obtain values for the following parameters from your network administrator for each i-data 7913 IPDS Printer LAN Attachment you plan to install:
  - Internet Protocol address
  - Subnet mask
  - Gateway address

Configuring the bootp daemon

To configure the system, you need to configure the **bootp** daemon on the Internet Protocol network. This information describes these tasks:

- Understanding how the **bootp** daemon works
- Completing the **bootp** configuration worksheet
- Editing the **bootptab** file

Understanding how the bootp daemon works

The TCP/IP **bootp** daemon on the server supports one or more i-data 7913 IPDS Printer LAN Attachments. When you start the 7913, it broadcasts an initialization request to all LAN systems. When the **bootp** daemon responds to this request, it accesses the `bootptab` file to send back configuration information and activate the 7913 as shown in . The data flow is:

1. On startup, the 7913 sends its universal MAC address, a unique address for each 7913, to the **bootp** daemon. The universal MAC address is printed on the side panel of each 7913 in the network.
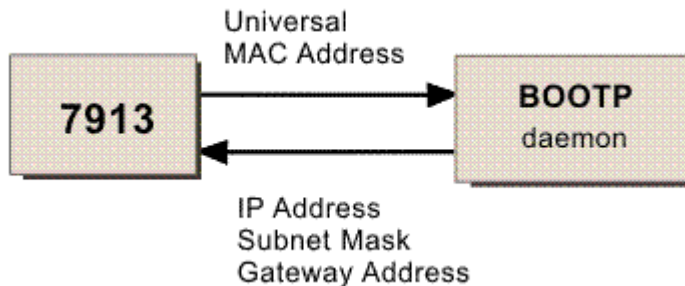
Figure 182. Data flow between the i-data 7913 IPDS Printer LAN Attachment and the LAN

IP address
  The Internet Protocol (IP) address—the full address of the 7913 as defined to the network. This address is used for configuration of the **bootp** daemon. You obtained this address earlier from your network administrator.

  Note: The IP address must be unique for each 7913 in your network.

Subnet mask
  The part of the IP address that represents the network identifier. You obtained this address earlier from your network administrator.

Gateway address
  The address of the gateway system used for routing to IP hosts on other networks. You obtained this address earlier from your network administrator.

For more information about **bootp** and related topics, see the following TCP/IP Request for Comments (RFCs):

- RFC 951: The Bootstrap Protocol (**bootp**)
- RFC 1060: Assigned Numbers

Completing the bootp configuration worksheet

Before you modify the **bootptab**, complete a **bootp** configuration worksheet for each 7913 you are installing. Enter the values for each parameter name.

Note: To obtain the Universal MAC address, look at the side panel of the i-data 7913 IPDS Printer LAN Attachment. Obtain the remaining values from your Internet Protocol network administrator.

Table 46. Configuration worksheet for bootp. Complete a separate worksheet for each 7913 you attach.	
Parameter name	**Value**
Ring speed (for IBM token-ring network model only)	
Universal MAC address	
IP address	
Subnet mask	
Gateway address	

Editing the bootptab file

The `bootptab` file contains a separate entry with configuration information for each 7913. You can use the same `bootptab` file in UNIX and AIX. The `bootptab` file is accessed under:

```
/etc
```

To configure the `bootptab` file for the 7913, you need to add the values from the worksheet (see Table 46 on page 475).

Notes:

1. You must have a separate entry with a unique IP address for each 7913.
2. On most AIX systems, you must have root user authority to update the `/etc/bootptab` file.

This information contains two examples of `bootptab` files:

- Example 1: Basic **bootp** configuration
- Example 2: More complex **bootp** configuration

Example 1: Basic bootp configuration

This example shows how to configure two i-data 7913 IPDS Printer LAN Attachments with the same subnet mask and gateway address. The configuration is:

```
P4028: ht=6: ha=00036EFFFFFF; ip=192.0.2.33:\
   sm=255.255-255.0: gw=192.0.2.255:

P3812: ht=6: ha=00036E00049F: ip=9.99.12.33:\
   sm=255.255.255.0: gw=192.0.2.255
```

The values that are used in Example 1 are:

P4028:=host name
Identifies the entry for the 7913 in the `bootptab` file. For example, the name P4028 identifies an entry for a 4028 printer attached to the LAN by use of a 7913. The host name can be anything that you want; it has no meaning to the system.

ht=host hardware type
Identifies whether the LAN is token ring or Ethernet. This entry must precede the host hardware address.

> For an IBM token-ring network, enter 6.
> For an Ethernet LAN, enter `ethernet`.

ha=host hardware address
Identifies the universal MAC address for the 7913. Each 7913 contains a unique MAC address that is specified in its hardware. The MAC address is printed on the side panel of the 7913.

ip=IP address
For a description of the IP address, see "IP address" on page 475.

> **Note:** The IP address must be unique for each 7913 in your network.

sm=subnet mask
For a description of the subnet mask, see "Subnet mask" on page 475.

gw=gateway address
For a description of the gateway address, see "Gateway address" on page 475.

Example 2: More complex bootp configuration

This example shows how to configure two 7913s in accordance with a more complex specification scheme. In this example, pointers are used to indicate an entry for the gateway and the subnet mask.

```
# global entry with information that every host uses
global.lan:\
    sm=255.255.255.-:
# the subnet entry with the gateway address
subnetida.trlan:\
        tc=global.lan: gw=192.0.2.255:
# entries for 7913 units
 P4028: tc=subnetida.trlan: ht=6: ha=00036EFFFFFF:\
        ip=subnetida.trlan: ht=6: ha=00036E00049F:\
 P3812: tc=subnetida.trlan: ht=6: ha=00036E00049F:\
        ip=192.0.2.33:
```

The values that are used in Example 2 are:

P4028:=host name
Identifies the entry for the 7913 in the `bootptab` file. For example, the name P4028 identifies an entry for a 4028 printer attached to the LAN by use of a 7913. The host name can be anything that you want; it has no meaning to the system.

tc=template host
Points to another entry that specifies addressing information. For example, `tc=subnetida.trlan` points to the `subnetida.trlan` entry that contains the gateway address.

ht=host hardware type
Identifies whether the LAN is token-ring or Ethernet. This entry must precede the host hardware address.

> For an IBM token-ring network, enter 6.
> For an Ethernet LAN, enter `ethernet`.

ha=host hardware address
Identifies the universal MAC address for the 7913. Each 7913 contains a unique MAC address that is specified in its hardware. The MAC address is printed on the side panel of the i-data 7913 IPDS Printer LAN Attachment.

ip=IP address
For a description of the IP address, see "IP address" on page 475.

Note: The IP address must be unique for each 7913 in your network.

Setting up the bootp daemon on AIX

After you configure the **bootp** daemon, you need to set it up in the AIX environment. To do this, you must have root user authority.

Typically, the 7913 and the IP host that is running the **bootp** daemon belong to the same IP subnetwork and also are on the same LAN segment; that is, they are not separated by a bridge. If you have token-ring LANs and bridges, and if your **bootp** daemon is running AIX 7.1.0 or later, the **bootp** daemon and the 7913 can be on different LAN segments.

The procedure for setting up the **bootp** daemon on AIX is:

1. Add theses two lines to your `/etc/services` file, if they are not already there:

   ```
   bootps    67/udp    # bootp server port
   bootpc    68/udp    # bootp client port
   ```

   The # character indicates that the text following it is a comment and does not affect the **inetd** super-daemon.

2. Start the **bootp** daemon with one of these methods:

   • You can use the **inetd** super-daemon; this is the typical way to start and manage the **bootp** daemon.

The `/etc/inetd.conf` file indicates which daemons are controlled by the **inetd** super-daemon. Use the AIX **more** command to view the contents of the file. If you see the following line and it is not preceded by a # character, the **bootp** daemon is controlled by the **inetd** super-daemon:

```
bootps  dgram  udp  wait  root  /etc/bootpd bootpd
```

If you do not see the previous line and you want the **bootpd** daemon to be managed by the **inetd** super-daemon, add the line to your `/etc/inetd.conf` file.

If you updated the `/etc/inetd.conf` file, enter these commands at the AIX command line prompt so that the **inetd** super-daemon recognizes the updates:

```
inetimp
refresh -s inetd
```

- You can start the **bootp** daemon manually, if it is not already being managed by the **inetd** super-daemon. Otherwise, do not start the **bootp** daemon manually.

  To start the **bootp** daemon manually, enter one of these commands at the AIX command line prompt:

  – `bootpd -s`

  – `bootpd -s -d -d -d -d -d`

  -s signifies that the **bootp** daemon is not controlled by the **inetd** super-daemon, and **-d** displays debugging information, such as the hardware address. Each additional **-d** increases the amount of debugging information that AIX displays. You can specify **-d** up to five times.

Configuring a twinaxial printer

For twinaxial printers, ensure that:

```
Station address = 0
```

Setting the controls for the i-data 7913 IPDS Printer LAN Attachment

Two models of the i-data 7913 IPDS Printer LAN Attachment are available:

- Model 03 for the IBM token-ring network
- Model 04 for the Ethernet LAN

Before you install the i-data 7913 IPDS Printer LAN Attachment, set the controls on the side panel of each 7913 by following the appropriate procedure:

- Setting the controls for IBM token-ring network models
- Setting the controls for Ethernet LAN models

Note: Before you change the controls, make sure that the i-data 7913 IPDS Printer LAN Attachment is disconnected from the electrical outlet.

IBM token-ring network model

For the IBM token-ring LAN model of the i-data 7913 IPDS Printer LAN Attachment, set the RINGSPEED switch and the PRINTERTYPE switch as shown in .
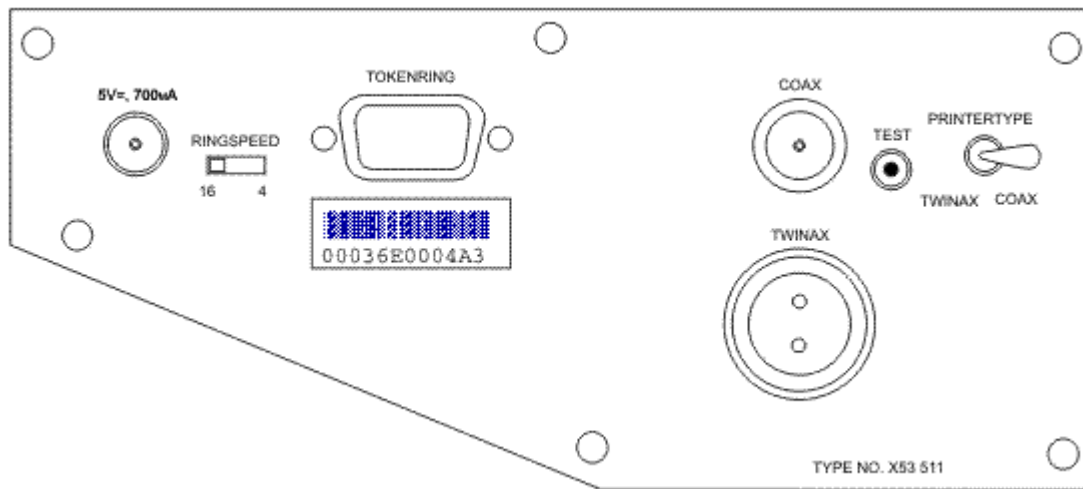
Figure 183. IBM token-ring network model of the i-data 7913 IPDS Printer LAN Attachment

To set the controls for the IBM token-ring network model of the i-data 7913 IPDS Printer LAN Attachment, follow these steps:

1. Before you change the controls, make sure that the i-data 7913 IPDS Printer LAN Attachment is disconnected from the electrical outlet.
2. Find out whether the LAN is 4 Mbps or 16 Mbps. You can get this information from your TCP/IP LAN network administrator.

   Note: If the RINGSPEED switch is set incorrectly, the LAN might malfunction.
3. Set the RINGSPEED switch to one of these:

     4 for 4-Mbps LANs
     16 for 16-Mbps LANs
4. Set the PRINTERTYPE switch to:

     COAX (for an IBM coaxial printer)
     TWINAX (for an IBM twinaxial printer)

Ethernet LAN model

For the Ethernet LAN model of the i-data 7913 IPDS Printer LAN Attachment, set the TWINAX/COAX rotary switch as shown in .
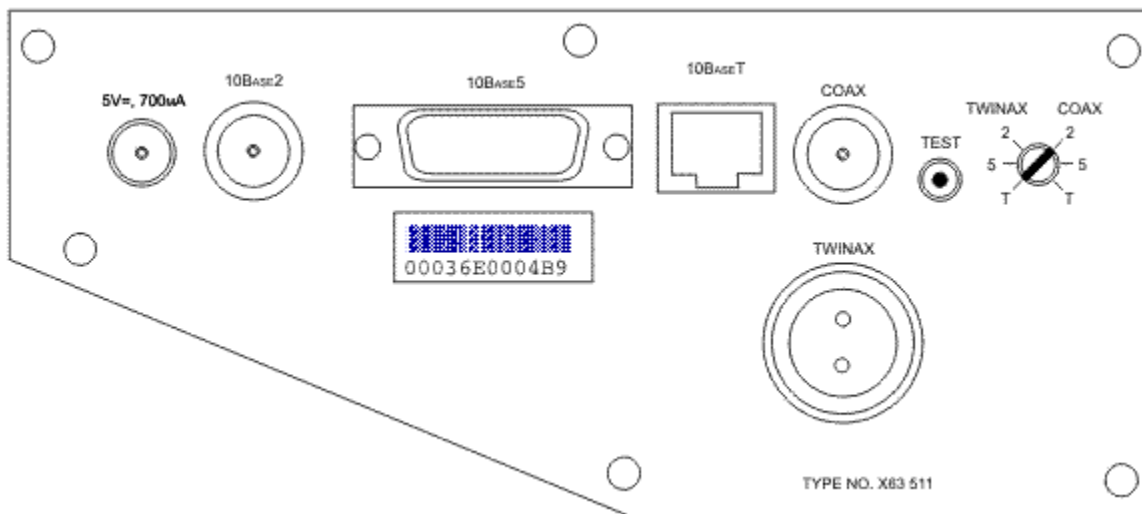
Figure 184. Ethernet LAN model of the i-data 7913 IPDS Printer LAN Attachment

To set the control for the Ethernet model of the i-data 7913 IPDS Printer LAN Attachment, follow these steps:

1. Before you change the controls, make sure that the i-data 7913 IPDS Printer LAN Attachment is disconnected from the electrical outlet.
2. Set the COAX/TWINAX rotary switch to one of the following positions:

   Coax 2
   > 10base2 to coax

   Coax 5
   > 10base5 to coax

   Coax T
   > 10baseT to coax

   Twinax 2
   > 10base2 to twinax

   Twinax 5
   > 10base5 to twinax

   Twinax T
   > 10baseT to twinax

Operating the i-data 7913 IPDS Printer LAN Attachment

Before you start the i-data 7913 IPDS Printer LAN Attachment, you must attach the 7913 to the LAN and to the printer. This information describes these tasks:

- Attaching the i-data 7913 IPDS Printer LAN Attachment
- Starting the i-data 7913 IPDS Printer LAN Attachment
- Shutting down the i-data 7913 IPDS Printer LAN Attachment

Attaching the i-data 7913 IPDS Printer LAN Attachment

To attach the i-data 7913 IPDS Printer LAN Attachment, you must install a cable to the LAN and to the printer. You must also install a ferrite cable block, which is included with the 7913. The ferrite cable block reduces the possibility of electrical interference.

Notes:

1. Cables to attach the 7913 to the LAN and to the printer are not provided with the i-data 7913 IPDS Printer LAN Attachment.
2. If you move a coaxial printer, move the ferrite cable block too.

To attach the i-data 7913 IPDS Printer LAN Attachment, follow these steps:

1. Attach the i-data 7913 IPDS Printer LAN Attachment to the LAN with a LAN connection cable.
2. Attach the i-data 7913 IPDS Printer LAN Attachment to the printer with a coaxial or twinaxial connection cable.
3. For coaxial printers, clamp the ferrite cable block to the coaxial cable, about 10 centimeters (4 inches) from the 7913.

Starting the i-data 7913 IPDS Printer LAN Attachment

To start the i-data 7913 IPDS Printer LAN Attachment, follow these steps:

1. Turn on the printer.
2. Plug the 7913 power cable into an electrical outlet. Because the 7913 is sensitive to power fluctuations, IBM recommends that you use an uninterruptible power supply (UPS).

   The POWER light comes on and several seconds later the LAN light blinks. This means that the 7913 is active on the IBM token-ring or Ethernet LAN, as in Figure 185 on page 481.
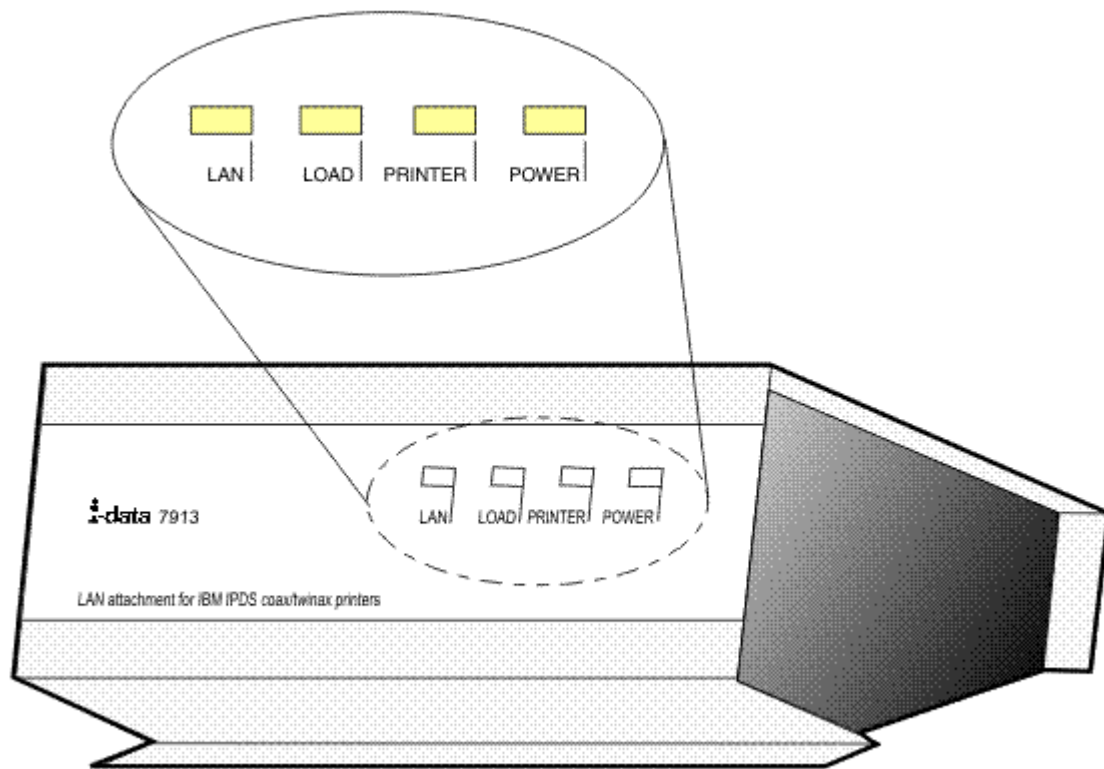
Figure 185. Status indicators on the i-data 7913 IPDS Printer LAN Attachment

Note: If the LAN light is not on, see "Troubleshooting the i-data 7913 IPDS Printer LAN Attachment" on page 484 for possible problems.

When the 7913 is powered on, the unit sends a **bootp** broadcast across the LAN with the MAC address. The **bootp** responds with:

- IP address

- Subnet mask
- Gateway address

If this information arrives successfully, the LAN light progresses from blinking status to a continuously lighted status.

For a more detailed explanation of the data flow that occurs when the i-data 7913 IPDS Printer LAN Attachment is set up, see "Understanding how the bootp daemon works" on page 475.

Shutting down the i-data 7913 IPDS Printer LAN Attachment

You can shut down the i-data 7913 IPDS Printer LAN Attachment in several ways:

- The preferred method is to stop the PSF FSA for the printer from the z/OS console by entering one of these commands, where *fsa_name* is the name of the PSF FSA for the printer:
  - For JES2:

    ```
    $P fsa_name
    ```

  - For JES3:

    ```
    *VARY fsa_name,OFFLINE
    *CANCEL fsa_name
    ```

  Then, turn off power to the i-data 7913 IPDS Printer LAN Attachment and to the printer.

- Turn off power to the printer. The i-data 7913 IPDS Printer LAN Attachment notifies PSF that the printer is powered off and the PSF FSA eventually ends normally.

- Turn off power to the i-data 7913 IPDS Printer LAN Attachment without turning off power to the printer.

  This procedure is not recommended because it might take PSF a long time to recognize that the attachment is powered off. The length of time depends on the value specified for the keep-alive packet transmissions. See "TCPCONFIG" on page 44.

  To end the PSF FSA for the printer, use the JES commands shown in the first bullet. If you are unable to stop or cancel the printer with the JES commands, enter this command:

  ```
  MODIFY fss_name,FORCE,fsa_name
  ```

  Note: Before you power the i-data 7913 IPDS Printer LAN Attachment back on, be sure to power off the printer and then power back on.

Testing the i-data 7913 IPDS Printer LAN Attachment

Before you test the installation of the i-data 7913 IPDS Printer LAN Attachment, make sure that the 7913 is plugged into an electrical outlet and attached to the LAN. These procedures are described in "Operating the i-data 7913 IPDS Printer LAN Attachment" on page 480.

To test the i-data 7913 IPDS Printer LAN Attachment, follow these steps:

1. Make sure that the POWER light is on and that the LAN light is constantly on (not blinking). This means that the 7913 is connected to the LAN.

2. Press the TEST button on the side panel of the 7913.

   A test page prints on the printer. Figure 186 on page 483 shows a sample test page for the IBM token-ring network model; Figure 187 on page 483 shows a sample test page for an Ethernet LAN mode. If no text page prints, see "Troubleshooting the i-data 7913 IPDS Printer LAN Attachment" on page 484 for possible problems.

   Note: Pressing the TEST button breaks the connection between PSF and the 7913.

```
i-data 7913-03 IPDS Printer LAN Attachment.
Version S80 085.000*05 / 00307.004
PROM Version S80 086.075 / 00380.043
FLASH Version S80 086.075 / 00308.043 (Active)
(c) 1993 i-data international a-s DK, phonr: +45 444 007
Ring Speed: 16 Mbps
LAN Type: Token Ring 16 Mbps
Early Token Ring Release: Yes
Boot Status: Booted
Printer Attachment: Coax
MAC Address: 00:03:6E:00:04:A1 Universal
IP Address: 192.0.2.33
Netmask: FFFFFF00
Default Gateway: 192.0.2.254
Broadcast Address: 192.0.2.255
Port: 5001
Response Timeout: 20 seconds
IP Defaults: None
Remote IP: Port:
Remote IP: Port:
```

Figure 186. Sample test page for IBM 7913 token-ring network model

```
i-data 7913-03 IPDS Printer LAN Attachment.
Version S80 086.000*05 / 00308.004
PROM Version S80 086.075 / 00380.043
FLASH Version S80 086.075 / 00308.043 (Active)
(C) 1993 i=data international a-s DK, phone: +45 444 007
LAN Type: Ethernet 10Base2
Boot Status: Booted
Printer Attachment: Coax
MAC Address: 00:03:6E:00:04:BC Universal
IP Address: 192.0.2.33
Netmask: FFFFFF00
Default Gateway: 192.0.2.254
Broadcast Address: 192.0.2.255
Port: 5001
IP Defaults: 192.99.10.132, 255.255.255.0, 192.99.10.220
Remote IP: Port:
Remote IP: Port:
```

Figure 187. Sample test page for Ethernet model

3. Ping the 7913 from the PSF host to verify the connectivity of both the physical LAN and the IP:

At the TSO command line, enter the ping command and the IP address or host name of the 7913. For example:

```
tso ping 192.0.2.33
```

Or from the SDSF command area, enter:

```
ping 192.0.2.33
```

If the ping is successful, you receive a system response that looks like this:

```
EZA0458I Ping V3R1: Pinging host 192.35.12.128.
(Use ATTN to interrupt.)
EZA0463I PING: Ping #1 response took 0.084 seconds.
Successes so far=1.
```

If the ping is not successful, or if problems occur with another LAN device, see "Troubleshooting the i-data 7913 IPDS Printer LAN Attachment" on page 484.

A successful ping typically indicates that the z/OS system can communicate with the i-data 7913 IPDS Printer LAN Attachment; however, you might receive a successful ping even though the IP address of the i-data 7913 IPDS Printer LAN Attachment is a duplicate of another IP address. If PSF is unable to establish a network connection with the i-data 7913 IPDS Printer LAN Attachment, or if PSF output for the printer prints elsewhere, follow these steps to verify that the IP address of the i-data 7913 IPDS Printer LAN Attachment is unique:

1. Turn off the i-data 7913 IPDS Printer LAN Attachment.

2. Wait at least 5 minutes for TCP/IP to clear the Address Resolution Protocol (ARP) tables.[29]

3. Enter the `ping` command again from the z/OS system to determine whether you receive a response.

If you receive a successful response to the `ping` command, indicating that duplicate IP addresses exist, consult your Internet Protocol network administrator to resolve the IP address problem.

Troubleshooting the i-data 7913 IPDS Printer LAN Attachment

This information contains symptoms and solutions for common problems you might encounter when you install or operate an i-data 7913 IPDS Printer LAN Attachment. For more problem-solving information you might need during installation, see the i-data publication that comes with the 7913 unit. For any further problem-solving information you need during operation, see the description of the PSF error message that is issued.

Table 47 on page 484 describes problems that might occur during the installation of the i-data 7913 IPDS Printer LAN Attachment.

Table 48 on page 485 describes problems that might occur during the operation of the i-data 7913 IPDS Printer LAN Attachment.

Table 47. Problems in installing the 7913 attachment

Symptoms	Possible problems	Solutions
LAN light goes on briefly, and then goes off. You attached the 7913 to a LAN and turned it on.	Token-ring speed set incorrectly.	Set the ring speed to: 4 for 4-Mbps LANs 16 for 16-Mbps LANs ⚠️ **Attention:** To determine whether your LAN is a 4-Mbps LAN or a 16-Mbps LAN, see your network administrator.
The LAN light blinks continuously. You attached the 7913 to a LAN and turned it on.	The **bootp** daemon is not started.	Start the **bootp** daemon on an AIX system.
	The 7913 and the **bootp** daemon are separated by a LAN bridge.	Make sure that you installed AIX 7.1.0 or later.
	The 7913 and the **bootp** daemon are separated by an IP router.	Configure the IP router to pass certain broadcast frames.
	The hardware type in the **bootp** entry is set incorrectly.	Set the hardware type to: `ht=6:` or `ht=tr:` for Token-Ring `ht=ethernet:` for Ethernet
	Cable attached incorrectly on the Ethernet LAN model.	Attach the 10Base2 cable to the 10Base2 connection, not to the COAX or TWINAX connection.

---

[29] You might need to wait longer if your installation specified a longer interval on the ARPAGE configuration statement in the TCP/IP profile. For information about the ARPAGE statement, see *z/OS Communications Server: IP Configuration Reference*.

Table 47. Problems in installing the 7913 attachment (continued)		
Symptoms	**Possible problems**	**Solutions**
No test page prints. You pressed the TEST button on the side of the 7913. The LAN and POWER lights are on. A printer is powered on, cabled to the 7913, and ready.	COAX/TWINAX switch set incorrectly.	For the Ethernet LAN model, set the COAX/TWINAX to: Coax 2: 10base2 to coax Coax 5: 10base5 to coax Coax T: 10baseT to coax Twinax 2: 10base2 to twinax Twinax 5: 10base5 to twinax Twinax T: 10baseT to twinax For the token-ring LAN model, set the PRINTERTYPE switch correctly, either to COAX or TWINAX. ⚠️ **Attention:** Turn off the 7913 before you change the switch.
No response on ping to the 7913.	The IP address of the z/OS system is not unique.	Consult your network administrator to resolve the IP address problem.
The printer immediately displays an error when powered on. A twinaxial printer is cabled to the 7913, and the 7913's switches indicate TWINAX.	The twinaxial printer's station address is incorrect.	Set the twinaxial printer's station address to 0.

Table 48. Problems in operating the 7913 attachment		
Symptoms	**Possible problems**	**Solutions**
The 7913 indicator lights indicate repeated power losses during operation. PSF repeatedly loses the connection to the 7913.	Power supply insufficient or overloaded.	Use an uninterruptible power supply (UPS).
PSF fails to establish a connection to the 7913, or loses the connection to the 7913.	The 7913 is powered on, but the printer is powered off.	Turn on the printer and restart the PSF FSA for the printer.
	Duplicate address for the 7913.	Follow Step "3" on page 483 in "Testing the i-data 7913 IPDS Printer LAN Attachment" on page 482 to verify that the address of the 7913 is unique.
	Duplicate address for the z/OS system.	Consult your network administrator.
	The MTU size for the z/OS system is too large.	For the recommended values of MTU, see "GATEWAY" on page 45.

Appendix N. APSGPPCT point counting tool

The price that you pay IBM for a PSF software license depends on the number of printers that PSF is driving and the speed of those printers. Printers are categorized and points are assigned to them based on the printer's device type and model number. The point counting tool, which is supplied with PSF, analyzes the printer points assigned to each printer in your installation, adds the points to get a total number of printer points, and determines your price tier based on the printer points. The price tier indicates the software license fee that you pay IBM.

Overview

The point counting tool, APSGPPCT, is a batch program that uses the PSF display printer information function to determine what type of printers PSF is communicating with. You activate the display printer information function by defining the PRTINFO DD statement in the PSF startup procedure. For every FSA that is started, the display printer information function automatically writes the printer information for that printer to the PRTINFO data set with a member name the same as the FSA name. APSGPPCT extracts this information from the printer information that is written to the data set:

- Printer (FSA) name
- Device type
- Model number
- Manufacturer
- Serial number
- Attachment identifier
- System name
- CPU ID

APSGPPCT uses the device type and model number to determine the correct number of printer points for each printer. After point values are determined for the printers that are driven by PSF, the program calculates the total point value by multiplying the number of printers of each type by their respective point values and adding the printer points. If no match is found on the device type and model number, the program uses a default point value of 65 to assign the number of points for that printer.

APSGPPCT uses the total point value, which represents the usage level for a particular PSF license, to select a PSF pricing tier. APSGPPCT then produces the PSF Tier Level Report, which lists your current printer point total and the PSF pricing tier that IBM recommends. You can use the IBM-supplied form definition and page definition to generate a printed report.

Note: The PSF Tier Level Report might contain inaccuracies because many printers report themselves as the type of printer they emulate, rather than the real printer machine type and model. In addition, the IBM 4000-4100 class machines report the same machine type for all models, even though the models have different rated speeds. If the point counting tool assigns an incorrect point value to a printer, verify that the printers that reported by the tool are the actual printers that you have installed, with the correct model and speed for each. If not, edit the APSGPSPD table to correct or add a printer speed and then rerun the tool.

Using the point counting tool

To use the point counting tool, you must:

- Install PSF for z/OS.
- Define the PRTINFO DD statement in the PSF startup procedure. See "Displaying printer information" on page 201 for information about how to set up the PRTINFO DD statement.
- Start each printer that is driven by PSF.

- Run the JCL EXEC, APSGPPCT. See "JCL for the point counting tool" on page 488.
- Print a hardcopy PSF Tier Level Report. See "PSF Tier Level Report" on page 488.
- Adjust the printer speed table, APSGPSPD, if your printer points are incorrect, and then rerun APSGPPCT. See "APSGPSPD printer speed table" on page 489.

JCL for the point counting tool

APSGPPCT is the program that runs the point counting tool. The JCL for running APSGPPCT requires these statements:

INDD
> Specifies the name of the data set referred to by the PRTINFO DD statement in the PSF startup procedure, such as:

```
//PRTINFO DD DSNAME=PSFMVS.PRTINFO,DISP=SHR
```

OUTDD
> Specifies the name of the output data set where the PSF Tier Level Report is placed.

Figure 188 on page 488 shows the sample JCL, APSGPPCJ, that you can use to run the point counting tool. APSGPPCJ is found in SYS1.SAMPLIB.

```
//APSGPPCJ
//*
//STEP01   EXEC PGM=APSGPPCT
//*
//SYSPRINT DD SYSOUT=A
//*
//PRTRPTIN DD DSN=PSFMVS.PRTINFO,DISP=SHR
//*
//PPCTRPT  DD DSN=PSFMVS.PPCT.REPORT,
//            DISP=(,CATLG,DELETE),
//            UNIT=SYSDA,VOL=SER=VVVVVV,
//            SPACE=(TRK,(1,1),RLSE),
//            DCB=(LRECL=512,RECFM=FB,BLKSIZE=0)
//*
//SYSIN    DD *
      INDD=PRTRPTIN,OUTDD=PPCTRPT
/*
```

Figure 188. Sample JCL for the point counting tool

PSF Tier Level Report

The PSF Tier Level Report that the point counting tool produces is in record format. You can view the report in softcopy, or you can use the IBM-supplied page definition, P1PPCT01, and form definition, F1PPCT01, to generate a printed report.

Note: The fonts in the printed version of your PSF Tier Level Report might be different than the fonts shown in Figure 189 on page 489.

```
            ******** PSF TIER LEVEL REPORT - 09/28/2005 11:33:27.65 ********

Printer Name  Device Type   Model Number   Manufacturer   Serial Number   Identifier     System Name   CPU
ID      Points
PRT610        2785          000            IBM            000000110001    192.0.156.80   DEV5
FF1200354381     6
PRT611        2775          000            IBM            000000140035    192.0.156.83   DEV5
FF1200354381     6
PRT612        4028          002                                           192.0.27.142   DEV5
FF1200354381     1
PRT613        2710          001            IBM            000000000000    192.0.64.179   DEV5
FF1200354381    12
PRT616        3300          0Û0            IBM            000000023190    192.0.162.84   DEV5
FF1200354381    20
PRT617        4100          HS2            IBM            000000000000    192.0.162.95   DEV5
FF1200354381    30
PRT618        4000          0cc            IBM            000000000000    192.0.227.58   DEV5
FF1200354381    30
PRT619        1111          002                                           192.0.227.41   DEV5
FF1200354381   **50

Based on your input, your current point total is 0155.
IBM recommends that PSF tier 10 be purchased.


Note: ** denotes the printer was not found in any table and was given the maximum point value.
```

Figure 189. PSF Tier Level Report

APSGPSPD printer speed table

Figure 190 on page 490 shows a sample of the APSGPSPD printer speed table that is found in SYS1.SAMPLIB. This table maps the FSA name, device type, or both to the printer speed. If the point counting tool assigns an incorrect point value to a printer, you can edit APSGPSPD to correct or add a printer speed. When the point counting tool is rerun, it searches this table first to get the printer speed so it can assign the correct point value. For the range of printer speeds and the points that are assigned to each printer category, see *PSF for z/OS: Licensed Program Specifications*.

To edit APSGPSPD:

1. Use these guidelines for editing the table:

   - Do not change the format of the table. For example, for each DC a CL8, a CL4, and an F must exist.
   - The dashes are used to prevent the point counting tool from incorrectly matching on a blank, but are not required.
   - These examples show the different types of valid entries:

     ```
     *               FSA Name      Type      Speed (Impressions per Minute) *
     *        DC    CL8'PRT619  ',CL4'----',F'381'                          *
     *        DC    CL8'PRT614  ',CL4'4100',F'1220'                         *
     *        DC    CL8'--------',CL4'4000',F'1002'                         *
     ```

   - If you add an impact or label printer, set the speed to 0 so no points are counted for that printer.

2. After you edit the table, compile and link-edit it. Figure 82 on page 259 contains sample JCL statements that you can use for compiling and link-editing an assembler module on z/OS.

3. Rerun the APSGPPCT exec for the point counting tool.

```
*/*******************************************************************/
*/*                                                                 */
*/*                    BAL CODE FOLLOWS                             */
*/*                                                                 */
*/*******************************************************************/
APSGPSPD CSECT                           ENTRY POINT OF THIS MODULE
APSGPSPD AMODE 24                        DO NOT REMOVE/CHANGE
APSGPSPD RMODE 24                        DO NOT REMOVE/CHANGE
SPEEDS@  DC    V(APSSPEED)               DO NOT REMOVE/CHANGE
*************************************************************************
* The following table maps the FSA name (printer name), device type   *
* or both to the printers speed (Impressions Per Minute).  The         *
* enrties in this table are of some known non-IBM printer types. These*
* entries may be changed to match a printer or new entries may be      *
* added.  The format of the table must not be changed i.e.. for each   *
* DC there must be a CL8, CL4 and a F.  The dashes are used to prevent*
* the point counting tool from incorrectly matching on a blank but are*
* not required.  The following are examples of the different types of  *
* valid entries:                                                       *
*             FSA Name    Type      Speed (Impressions per Minute) *
*        DC    CL8'PRT619  ',CL4'----',F'381'                          *
*        DC    CL8'PRT614  ',CL4'4100',F'1220'                         *
*        DC    CL8'--------',CL4'4000',F'1002'                         *
* If an impact or label printer is added the speed should be set to    *
* 0 so there are no points counted for that printer.                   *
*                                                                      *
*************************************************************************
         ENTRY APSSPEED                  DO NOT REMOVE/CHANGE
APSSPEED DS    0D                        DO NOT REMOVE/CHANGE
*             FSA Name    Type      Speed (Impressions per Minute) *
         DC    CL8'--------',CL4'106 ',F'6'
         DC    CL8'--------',CL4'218 ',F'18'
         DC    CL8'--------',CL4'1012',F'10'
         DC    CL8'--------',CL4'1124',F'3'
         DC    CL8'--------',CL4'1324',F'30'
         DC    CL8'--------',CL4'2012',F'10'
         DC    CL8'--------',CL4'2030',F'24'
         DC    CL8'--------',CL4'2050',F'50'
         DC    CL8'--------',CL4'2090',F'90'
         DC    CL8'--------',CL4'2140',F'140'
         DC    CL8'--------',CL4'2200',F'103'
         DC    CL8'--------',CL4'2300',F'206'
         DC    CL8'--------',CL4'3012',F'18'
         DC    CL8'--------',CL4'3610',F'3'
         DC    CL8'--------',CL4'3630',F'4'
         DC    CL8'--------',CL4'3920',F'20'
         DC    CL8'--------',CL4'5050',F'200'
         DC    CL8'--------',CL4'5424',F'30'
         DC    CL8'--------',CL4'6024',F'24'
         DC    CL8'--------',CL4'6060',F'60'
         DC    CL8'--------',CL4'6090',F'100'
         DC    CL8'--------',CL4'6100',F'103'
         DC    CL8'--------',CL4'6481',F'100'
         DC    CL8'--------',CL4'6482',F'200'
         DC    CL8'--------',CL4'6890',F'88'
         DC    CL8'--------',CL4'6900',F'235'
         DC    CL8'--------',CL4'7224',F'30'
         DC    CL8'--------',CL4'9324',F'3'
*************************************************************************
* The next line is the end of table indicator do not remove or change.*
*************************************************************************
         DC    CL8'FFFFFFFF',CL4'FFFF',F'9999'
         END   APSGPSPD                  DO NOT REMOVE/CHANGE
```

Figure 190. APSGPSPD printer speed table

Appendix O. Accessibility

Accessible publications for this product are offered through IBM Documentation (www.ibm.com/docs/en/zos).

If you experience difficulty with the accessibility of any z/OS information, send a detailed message to the Contact the z/OS team web page (www.ibm.com/systems/campaignmail/z/zos/contact_z) or use the following mailing address.

IBM Corporation
Attention: MHVRCFS Reader Comments
Department H6MA, Building 707
2455 South Road
Poughkeepsie, NY 12601-5400
United States

Accessibility features

Accessibility features help users who have physical disabilities such as restricted mobility or limited vision use software products successfully. The accessibility features in z/OS can help users do the following tasks:

- Run assistive technology such as screen readers and screen magnifier software.
- Operate specific or equivalent features by using the keyboard.
- Customize display attributes such as color, contrast, and font size.

Consult assistive technologies

Assistive technology products such as screen readers function with the user interfaces found in z/OS. Consult the product information for the specific assistive technology product that is used to access z/OS interfaces.

Keyboard navigation of the user interface

You can access z/OS user interfaces with TSO/E or ISPF. The following information describes how to use TSO/E and ISPF, including the use of keyboard shortcuts and function keys (PF keys). Each guide includes the default settings for the PF keys.

- *z/OS TSO/E Primer*
- *z/OS TSO/E User's Guide*
- *z/OS ISPF User's Guide Vol I*

Dotted decimal syntax diagrams

Syntax diagrams are provided in dotted decimal format for users who access IBM Documentation with a screen reader. In dotted decimal format, each syntax element is written on a separate line. If two or more syntax elements are always present together (or always absent together), they can appear on the same line because they are considered a single compound syntax element.

Each line starts with a dotted decimal number; for example, 3 or 3.1 or 3.1.1. To hear these numbers correctly, make sure that the screen reader is set to read out punctuation. All the syntax elements that have the same dotted decimal number (for example, all the syntax elements that have the number 3.1)

are mutually exclusive alternatives. If you hear the lines `3.1 USERID` and `3.1 SYSTEMID`, your syntax can include either USERID or SYSTEMID, but not both.

The dotted decimal numbering level denotes the level of nesting. For example, if a syntax element with dotted decimal number 3 is followed by a series of syntax elements with dotted decimal number 3.1, all the syntax elements numbered 3.1 are subordinate to the syntax element numbered 3.

Certain words and symbols are used next to the dotted decimal numbers to add information about the syntax elements. Occasionally, these words and symbols might occur at the beginning of the element itself. For ease of identification, if the word or symbol is a part of the syntax element, it is preceded by the backslash (\) character. The * symbol is placed next to a dotted decimal number to indicate that the syntax element repeats. For example, syntax element *FILE with dotted decimal number 3 is given the format `3 \* FILE`. Format `3* FILE` indicates that syntax element FILE repeats. Format `3* \* FILE` indicates that syntax element `* FILE` repeats.

Characters such as commas, which are used to separate a string of syntax elements, are shown in the syntax just before the items they separate. These characters can appear on the same line as each item, or on a separate line with the same dotted decimal number as the relevant items. The line can also show another symbol to provide information about the syntax elements. For example, the lines `5.1*, 5.1 LASTRUN, and 5.1 DELETE` mean that if you use more than one of the LASTRUN and DELETE syntax elements, the elements must be separated by a comma. If no separator is given, assume that you use a blank to separate each syntax element.

If a syntax element is preceded by the % symbol, it indicates a reference that is defined elsewhere. The string that follows the % symbol is the name of a syntax fragment rather than a literal. For example, the line `2.1 %OP1` means that you must refer to separate syntax fragment OP1.

The following symbols are used next to the dotted decimal numbers.

? indicates an optional syntax element
The question mark (?) symbol indicates an optional syntax element. A dotted decimal number followed by the question mark symbol (?) indicates that all the syntax elements with a corresponding dotted decimal number, and any subordinate syntax elements, are optional. If there is only one syntax element with a dotted decimal number, the ? symbol is displayed on the same line as the syntax element, (for example `5? NOTIFY`). If there is more than one syntax element with a dotted decimal number, the ? symbol is displayed on a line by itself, followed by the syntax elements that are optional. For example, if you hear the lines `5 ?`, `5 NOTIFY`, and `5 UPDATE`, you know that the syntax elements NOTIFY and UPDATE are optional. That is, you can choose one or none of them. The ? symbol is equivalent to a bypass line in a railroad diagram.

! indicates a default syntax element
The exclamation mark (!) symbol indicates a default syntax element. A dotted decimal number followed by the ! symbol and a syntax element indicate that the syntax element is the default option for all syntax elements that share the same dotted decimal number. Only one of the syntax elements that share the dotted decimal number can specify the ! symbol. For example, if you hear the lines `2? FILE`, `2.1! (KEEP)`, and `2.1 (DELETE)`, you know that (KEEP) is the default option for the FILE keyword. In the example, if you include the FILE keyword, but do not specify an option, the default option KEEP is applied. A default option also applies to the next higher dotted decimal number. In this example, if the FILE keyword is omitted, the default `FILE(KEEP)` is used. However, if you hear the lines `2? FILE`, `2.1`, `2.1.1! (KEEP)`, and `2.1.1 (DELETE)`, the default option KEEP applies only to the next higher dotted decimal number, `2.1` (which does not have an associated keyword), and does not apply to `2? FILE`. Nothing is used if the keyword FILE is omitted.

* indicates an optional syntax element that is repeatable
The asterisk or glyph (*) symbol indicates a syntax element that can be repeated zero or more times. A dotted decimal number followed by the * symbol indicates that this syntax element can be used zero or more times; that is, it is optional and can be repeated. For example, if you hear the line `5.1* data area`, you know that you can include one data area, more than one data area, or no data area. If you hear the lines `3*`, `3 HOST`, `3 STATE`, you know that you can include `HOST`, `STATE`, both together, or nothing.

Notes:

1. If a dotted decimal number has an asterisk (*) next to it and there is only one item with that dotted decimal number, you can repeat that same item more than once.

2. If a dotted decimal number has an asterisk next to it and several items have that dotted decimal number, you can use more than one item from the list, but you cannot use the items more than once each. In the previous example, you can write HOST STATE, but you cannot write HOST HOST.

3. The * symbol is equivalent to a loopback line in a railroad syntax diagram.

+ indicates a syntax element that must be included

The plus (+) symbol indicates a syntax element that must be included at least once. A dotted decimal number followed by the + symbol indicates that the syntax element must be included one or more times. That is, it must be included at least once and can be repeated. For example, if you hear the line 6.1+ data area, you must include at least one data area. If you hear the lines 2+, 2 HOST, and 2 STATE, you know that you must include HOST, STATE, or both. Similar to the * symbol, the + symbol can repeat a particular item if it is the only item with that dotted decimal number. The + symbol, like the * symbol, is equivalent to a loopback line in a railroad syntax diagram.

Notices

This information was developed for products and services that are offered in the USA or elsewhere.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
United States of America

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

This information could include missing, incorrect, or broken hyperlinks. Hyperlinks are maintained in only the HTML plug-in output for the Documentation. Use of hyperlinks in other output formats of this information is at your own risk.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Site Counsel
2455 South Road

Poughkeepsie, NY 12601-5400
USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or

reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

IBM Online Privacy Statement

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user, or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

Depending upon the configurations deployed, this Software Offering may use session cookies that collect each user's name, email address, phone number, or other personally identifiable information for purposes of enhanced user usability and single sign-on configuration. These cookies can be disabled, but disabling them will also eliminate the functionality they enable.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at ibm.com/privacy and IBM's Online Privacy Statement at ibm.com/privacy/details in the section entitled "Cookies, Web Beacons and Other Technologies," and the "IBM Software Products and Software-as-a-Service Privacy Statement" at ibm.com/software/info/product-privacy.

Policy for unsupported hardware

Various z/OS elements, such as DFSMSdfp, JES2, JES3, and MVS, contain code that supports specific hardware servers or devices. In some cases, this device-related element support remains in the product even after the hardware devices pass their announced End of Service date. z/OS may continue to service element code; however, it will not provide service related to unsupported hardware devices. Software problems related to these devices will not be accepted for service, and current service activity will cease if a problem is determined to be associated with out-of-support devices. In such cases, fixes will not be issued.

Minimum supported hardware

The minimum supported hardware for z/OS releases identified in z/OS announcements can subsequently change when service for particular servers or devices is withdrawn. Likewise, the levels of other software products supported on a particular release of z/OS are subject to the service support lifecycle of those

products. Therefore, z/OS and its product publications (for example, panels, samples, messages, and product documentation) can include references to hardware and software that is no longer supported.

- For information about software support lifecycle, see: IBM Lifecycle Support for z/OS (www.ibm.com/software/support/systemsz/lifecycle)
- For information about currently-supported IBM hardware, contact your IBM representative.

Programming interfaces

This publication includes documentation of intended programming interfaces that the customer can use to write programs to obtain the services of PSF for z/OS.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available at Copyright and Trademark information (www.ibm.com/legal/copytrade.shtml).

Adobe and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Linux® is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Glossary

This glossary defines technical terms and abbreviations used in PSF for z/OS documentation.

These cross-references are used in this glossary:

See
> Refers to preferred synonyms or to defined terms for acronyms and abbreviations.

See also
> Refers to related terms that have similar, but not synonymous, meanings, or to contrasted terms that have opposite or substantively different meanings.

Numerics

16 mm
> Microfilm with frames that are arranged in a 16 mm-high strip, typically 100 feet long, to fit in a viewer cartridge. Microfilm uses sequential retrieval marks for view positioning.

A

abend
> See abnormal end of task.

abnormal end of task (abend)
> The termination of a task, job, or subsystem because of an error condition that recovery facilities cannot resolve during processing

above the bar storage
> Storage above the 2 GB bar. The system programmer can specify that all inline resources are stored in this area during job processing. This is useful when there might not be enough virtual storage in the region area (above the 16 MB line and below the 2 GB bar).

access method
> A technique for moving data between main storage and input/output devices.

accumulator
> A printer hardware feature that supplies a separate storage area to hold data in raster form. It can be used either for composing a sheet of data that combines a large amount of variable and constant data, or for storing an electronic overlay in raster form so that the overlay is merged with variable data as the page is printed.

ACIF
> See AFP Conversion and Indexing Facility.

ACK
> See positive acknowledgment reply.

acknowledgment character (ACK character)
> A transmission control character that is sent as an affirmative response to a data transmission.

action code
> A software-generated or hardware-generated code that indicates a recovery action. In printers, the hardware action code is byte 2 of the sense data.

active environment group
> A collection of mapping structured fields, positioning controls, and data descriptors that define the environment for a page. These structured fields form an internal object in a composed text page, page definition, or overlay.

addressable point
> For page printers, any defined position or picture element in a presentation surface or physical medium that can be referenced. See also picture element and print position.

Address Resolution Protocol (ARP)
A protocol that dynamically maps an IP address to a network adapter address in a local area network.

advanced function common control unit (AFCCU)
A controller that converts Intelligent Printer Data Stream (IPDS) into a presentation format that is usable by COM and that transfers the setup data, document pages, and text-related information to COM.

Advanced Function Presentation (AFP)
A set of licensed programs, together with user applications, that use the all-points-addressable concept to print data on a wide variety of printers or to display data on a variety of display devices. AFP includes creating, formatting, archiving, retrieving, viewing, distributing, and printing information.

Advanced Function Presentation data stream (AFPDS)
A presentation data stream that is processed in the AFP environment. MO:DCA is the AFP interchange data stream. IPDS is the AFP printer data stream.

Advanced Interactive Executive (AIX)
A UNIX operating system developed by IBM that is designed and optimized to run on POWER microprocessor-based hardware, such as servers, workstations, and blades.

AFCCU
See advanced function common control unit.

AFP
See Advanced Function Presentation.

AFP Conversion and Indexing Facility (ACIF)
An optional feature of PSF for z/OS that converts a print file into a MO:DCA document, creates an index file for later retrieval and viewing, and retrieves resources used by an AFP document into a separate file.

AFPDS
See Advanced Function Presentation data stream.

AFP resource installer
An application that runs on a Windows workstation and installs and manages fonts, data objects, and color management resources (CMRs) in resource libraries. It also creates CMRs and associates CMRs with data objects.

AFP Statistics (AFPSTATS) report
Contains summary data about the resources used to print a document. The AFPSTATS report is used to indicate in which libraries PSF found a resource, diagnose some resource selection problems, obtain statistical data about how a print file is printed, and diagnose some print file printing performance problems.

AFPSTATS report
See AFP Statistics report.

AFPSTATS repository
A data set where AFP Statistics (AFPSTATS) reports are written.

AIX
See Advanced Interactive Executive.

all-points addressability (APA)
The capability to address, reference, and position text, overlays, and images at any defined position or picture element on the printable area of the paper. This capability depends on the ability of the hardware to address and to display each picture element.

all-points addressable (APA)
Pertaining to addressing, referencing, and positioning text, overlays, and images at any defined position or picture element on the printable area of the paper.

alphanumeric
Pertaining to a character set that contains letters, digits, and other characters, such as punctuation marks.

American National Standards Institute (ANSI)
A private, nonprofit organization whose membership includes private companies, US government agencies, and professional, technical, trade, labor, and consumer organizations. ANSI coordinates the development of voluntary consensus standards in the US.

American Standard Code for Information Interchange (ASCII)
A standard code used for information exchange among data processing systems, data communication systems, and associated equipment. ASCII uses a coded character set consisting of 7-bit coded characters. See also Extended Binary Coded Decimal Interchange Code.

ANSI
See American National Standards Institute.

AOS
Alternative offset stacker.

APA
See all-points addressability or all-points addressable.

APAR
See authorized program analysis report.

application program
A program used to communicate with stations in a network, enabling users to perform application-oriented activities.

ARP
See Address Resolution Protocol.

ASCII
See American Standard Code for Information Interchange.

ASYNC
See asynchronous.

asynchronous (ASYNC)
Pertaining to events that are not synchronized in time or do not occur in regular or predictable time intervals. See also synchronous.

authorized program analysis report (APAR)
A request for correction of a defect in a supported release of an IBM-supplied program.

auxiliary data set
In AFP printing, a data set that contains job header, data set header, job trailer, or message data. See also print data set.

auxiliary resource
Fonts, page segments, overlays, page definitions, or form definitions associated with auxiliary data sets.

B

bar code
An array of elements, such as bars, spaces, and two-dimensional modules, that encode data in a particular symbology. The elements are arranged in a predetermined pattern following unambiguous rules defined by the symbology.

Bar Code Object Content Architecture (BCOCA)
An architected collection of constructs used to interchange and present bar code data.

BCOCA
See Bar Code Object Content Architecture.

bin
An enclosure on a printer that contains source or destination media, including paper, foils, labels, card stock, or microfilm. See also cassette and stacker.

bitmap
A coded representation in which each bit, or group of bits, represents or corresponds to an item; for example, a configuration of bits in main storage in which each bit indicates whether a peripheral device or a storage block is available or in which each group of bits corresponds to one pixel of a display image.

bounded-box font
A font in bounded-box format. See also unbounded-box font.

BTS
See burster-trimmer-stacker.

buffer pool
An area of storage in which all buffers of a program are kept.

burst
To separate continuous-forms paper into separate sheets.

burster-trimmer-stacker (BTS)
An optional printer feature that separates continuous forms into separate sheets, trims the carrier strip from both edges of the paper, and stacks the sheets. The BTS also identifies jobs by offsetting the stacking.

C

capture
The process by which some printers can save downloaded raster and outline fonts as temporary printer-resident fonts.

carriage control character
A character that is used to specify a write, space, or skip operation. See also control character.

cassette
In cut-sheet printers, a removable container for a supply of paper. See also bin.

CCW
See channel command word.

CFS
See continuous-forms stacker.

channel-attached
Pertaining to the attachment of devices directly by input/output channels to a host processor. See also SNA-attached and TCP/IP-attached.

channel code
A number from 1 to 12 that identifies a position in the forms control buffer or a page definition.

channel command word (CCW)
An 8-byte command issued to the channel subsystem by a central processor and operating asynchronously with the issuing processor.

channel counter
A counting device that identifies how many pages have been successfully received.

character
1. Any symbol that can be entered on a keyboard, printed, or displayed. For example, letters, numbers, and punctuation marks are all characters.

2. In a computer system, a member of a set of elements that is used for the representation, organization, or control of data. See also control character, glyph, and graphic character.

3. In bar codes, a single group of bars and spaces that represent an individual number, letter, punctuation mark, or other symbol.

character rotation
The alignment of a character with respect to its character baseline, measured in degrees in a clockwise direction. See also rotation and orientation.

character set
A defined set of characters that can be recognized by a configured hardware or software system. A character set can be defined by alphabet, language, script, or any combination of these items. See also font character set.

checkpoint
A place in a program at which a check is made, or at which a recording of data is made to allow the program to be restarted in case of interruption.

client
A software program or computer that requests access to data, services, programs, and resources from a server. See also server and host.

CMR
See color management resource.

coded font
A font file that associates a code page and a font character set. For double-byte fonts, a coded font associates multiple pairs of code pages and font character sets.

coded font section
A font character set and code page pair. A single-byte coded font consists of only one coded font section; a double-byte coded font can consist of more than one.

code page
A particular assignment of code points to graphic characters. Within a given code page, a code point can only represent one character. A code page also identifies how undefined code points are handled. See also coded font and extended code page.

code page global identifier (CPGID)
A 5-digit decimal or 2-byte binary identifier that is assigned to a code page. The range of values is 00001 to 65534 (X'0001' to X'FFFE').

code point
A unique bit pattern that represents a character in a code page.

colon hexadecimal notation
The syntactical representation for a 128-bit integer that consists of eight groups of four hexadecimal numbers, separated by colons. IP addresses can be represented in colon hexadecimal notation. See also dotted decimal notation and host name.

color management resource (CMR)
An object that provides color management in presentation environments.

color mapping table
A MO:DCA object that is used to map color values specified in a source color space to color values specified in a target color space. This object is loaded into printers that support the color mapping table.

COM
See computer output microfilm.

COM device
See microfilm device.

command
A request from a terminal or automated operator for the performance of an operation or service, or a request in a batch-processing job or print file for the operation or execution of a particular program.

communication
See data communication.

communication-attached
Pertaining to a device that is SNA-attached and that uses a communication controller. See also local-attached.

compatibility font
An AFP raster font designed to emulate the uniformly spaced and fixed-pitch fonts used with line printers.

compatibility mode
A mode of operation in which a device can simulate the function of another device or model. The device functions like a different device of the same type, ignoring some or all of the additional features that the device might possess. Compatibility mode permits a migration between devices with minimal impact on programs that have device dependencies.

composed text
Text that has been formatted and that contains text-control information to direct the presentation of the text.

composed-text data
Data that has been composed into pages. Text formatting programs such as DCF can produce composed text data, which consists entirely of AFP structured fields. ACIF and AFP Download Plus can transform line data and XML data to composed-text data. See also MO:DCA data.

computer output microfilm (COM)
The hardware controller that manages the microfilm print engine and processes the functions unique to COM.

computing system RPQ
A customer request for a price quotation on alterations or additions to the functional capabilities of a computing system, hardware product, or device. The RPQ can be used in conjunction with programming RPQs to solve unique data processing problems. See also programming request for price quotation.

COM setup data
Data that the PSF user uses to designate unique microfilm printing functions for AFP print jobs. The parameter values, structure, syntax, and semantics are defined by the COM manufacturer, not by IBM-controlled architectures.

concatenate

1. To link together.

2. To join two character strings.

concatenated data set
A group of logically connected data sets that are treated as one data set for the duration of a job step. See also data set, partitioned data set, and library.

conditional processing
A page definition function that allows input data records to partially control their own formatting.

console
A display station from which an operator can control and observe the system operation.

continuous forms
A series of connected forms that feed continuously through a printing device. The connection between the forms is perforated so that the user can tear them apart. Before printing, the forms are folded in a stack, with the folds along the perforations. See also cut-sheet paper.

continuous-forms stacker (CFS)
In continuous-forms printers, an output assembly that refolds and stacks continuous forms after printing.

control character

1. A character that represents a command that is sent to an output device, such as a printer or monitor. Examples are line-feed, shift-in, shift-out, carriage return, font change, and end of transmission. See also carriage control character.

2. A character whose occurrence in a particular context initiates, modifies, or stops a control function.

copy group
An internal object in a form definition or a print data set that controls such items as modifications to a form, page placement, and overlays. See also internal copy group.

copy modification

The process of adding, deleting, or replacing data on selected copies of either a presentation space or certain pages of a print job.

copy separation

The method or printer mechanism for distinguishing consecutive copies of a single data set or print file. In the continuous-forms stacker, the method consists of the alternation between one, two, or three vertical bars placed on the left carrier strip between forms. In the burster-trimmer-stacker or for cut-sheet paper, the mechanism consists of offset stacking.

CPGID

See code page global identifier.

CSE

See cut-sheet emulation.

cut-sheet emulation (CSE)

The ability of a continuous-forms printer to provide output similar to output from a cut-sheet printer.

cut-sheet paper

Paper that is cut into uniform-size sheets before it is loaded into the printer. See also continuous forms.

D

daemon

A program that runs unattended to perform continuous or periodic functions, such as network control.

DASD

See direct access storage device.

data check

A synchronous or asynchronous indication of a condition caused by erroneous data or incorrect positioning of data. Some data checks can be suppressed.

data communication

Transfer of data among functional units by means of data transmission protocols.

data control block (DCB)

A control block used by access method routines in storing and retrieving data.

Data Facility Storage Management Subsystem (DFSMS)

An operating environment that helps automate and centralize the management of storage. To manage storage, the storage management subsystem (SMS) provides the storage administrator with control over data class, storage class, management class, storage group, and automatic class selection (ACS) routine definitions.

data link

The physical connection (communications lines, modems, controller, work stations, other communications equipment), and the rules (protocols) for sending and receiving data between two or more locations in a data network. See also telecommunication line.

data object resource

An object container resource or IOCA image resource that is either printer resident or downloaded. Data object resources can be:

- Used to prepare for the presentation of a data object, such as with a resident color profile resource object
- Included in a page or overlay through the Include Object (IOB) structured field; for example, PDF single-page and multiple-page objects, Encapsulated PostScript (EPS) objects, and IOCA images
- Called from within a data object; for example, PDF resource objects

data set

The major unit of data storage and retrieval, consisting of a collection of data in one of several prescribed arrangements and described by control information to which the system has access. See also file, concatenated data set, partitioned data set, and sequential data set.

data set header
A page in printed output that separates multiple data sets or multiple copies of a data set within a print job. See also job header.

DCB
See data control block.

DCF
See Document Composition Facility.

default
Pertaining to an attribute, value, or option that is assumed when none is explicitly specified.

deferred-printing mode
A printing mode that spools output through JES to a data set instead of printing it immediately. Output is controlled by using JCL statements. See also direct-printing mode.

device manager
The subcomponent of PSF that manages the interface to the printer.

DFSMS
See Data Facility Storage Management Subsystem

direct access storage device (DASD)
A device that allows storage to be directly accessed, such as a disk drive.

Direct Printer Services Subsystem (DPSS)
The PSF subcomponent that acts as the interface between PSF and an application program when the JES is not spooling jobs for a printer. DPSS attaches PSF as a subtask for use in the direct-printing mode.

direct-printing mode
A printing mode that gives PSF exclusive use of a channel-attached printer. Output is printed immediately and is not spooled through JES. See also deferred-printing mode.

disabled mechanism
A function of a printer that is temporarily out of operation or is not supported. In such a case, the device manager, such as PSF, might allow jobs to print with alternative options. See also enabled.

distributed print function (DPF)
A component of a Windows print server that can be installed and used to print jobs from PSF.

document

1. A machine-readable collection of one or more objects that represent a composition, a work, or a collection of data.

2. Data that has already been composed into pages and that contains a Begin Document and an End Document structured field.

Document Composition Facility (DCF)
An IBM licensed program used to format input to a printer.

dotted decimal notation
The syntactical representation for a 32-bit integer that consists of four 8-bit numbers written in base 10 and separated by periods (dots). IP addresses can be represented in dotted decimal notation. See also colon hexadecimal notation and host name.

double-byte coded font
A font in which the characters are defined by 2 bytes. The first byte defines the coded font section; the second byte defines the code point in the code page specified for that section. See also single-byte coded font.

download
To transfer data from a computer to a connected device, such as a workstation or a printer. Typically, users download from a large computer to a diskette or fixed disk on a smaller computer or from a system unit to an adapter.

DPF
See distributed print function.

DPSS
    See Direct Printer Services Subsystem.

drain
    An operator action to halt the flow of jobs to a printer, typically to stop the printer or to change print options.

duplex
    Pertaining to printing on both sides of a sheet of paper. See also normal duplex, simplex, and tumble duplex.

E

EBCDIC
    See Extended Binary Coded Decimal Interchange Code.

eject to front facing (ETFF)
    A flag used in a resource exit to tell PSF to bypass the eject-to-front-facing function.

enabled

1. Pertaining to a state of the processing unit that allows the occurrence of certain types of interruptions.

2. A condition of the printer (physically selected) in which the printer is available to the host processor for typical work. The printer is online when in an enabled condition. See also disabled mechanism.

Enterprise Systems Architecture (ESA)
    A hardware architecture that reduces the effort required for managing data sets and extends addressability for system, subsystem, and application functions.

Enterprise Systems Connection (ESCON)
    A peripheral interface for an Enterprise Systems Architecture/390 and IBM Z computer. The I/O interface uses ESA/390 logical protocols over a serial interface that configures attached units to a communication fabric.

ERP
    See error-recovery procedure.

error-recovery procedure (ERP)
    A procedure designed to help isolate and, where possible, to recover from errors in equipment. The procedure is often used in conjunction with programs that record information about machine malfunctions.

ESA
    See Enterprise Systems Architecture.

ESCON
    See Enterprise Systems Connection.

ESTAE
    See extended specify task abnormal exit.

ETFF
    See eject to front facing.

Ethernet
    A packet-based networking technology for local area networks (LANs) that supports multiple access and handles contention by using Carrier Sense Multiple Access with Collision Detection (CSMA/CD) as the access method. Ethernet is standardized in the IEEE 802.3 specification.

exception
    A condition or event that cannot be handled by a normal process.

exception highlighting
    The markings placed on the printed page to indicate the location of a data-stream error.

execution
>The process of carrying out an instruction or instructions of a computer program by a computer.

exit
>An instruction in an application, routine, or subroutine that causes control to pass to another application, routine, or subroutine. See also installation exit.

Extended Binary Coded Decimal Interchange Code (EBCDIC)
>A coded character set of 256 eight-bit characters developed for the representation of textual data. EBCDIC is not compatible with ASCII character coding. See also American Standard Code for Information Interchange.

extended code page
>A code page that is stored in a partitioned data set (PDS or PDSE) in a font resource library or in a UNIX file in a font path library. Extended code pages might contain Unicode values that a printer uses to print EBCDIC or ASCII encoded text strings with TrueType and OpenType fonts.

extended specify task abnormal exit (ESTAE)
>A z/OS macro that provides recovery capability and gives control to the user-specified exit routine for processing, diagnosing an abend, or specifying a retry address.

F

FCB
>See forms control buffer.

FGID
>See font typeface global identifier.

fibre-channel connection (FICON)
>A fibre-channel communication protocol designed for mainframe computers and peripheral devices.

fiche
>See microfiche.

FICON
>See fibre-channel connection.

file

>1. A collection of related data that is stored and retrieved by an assigned name. A file can include information that starts a program (program-file object), contains text or graphics (data-file object), or processes a series of commands (batch file).

>2. See also data set, partitioned data set, sequential data set, and library.

fixed metrics
>Measurement information in specific units such as pels, inches, or centimeters for individual or collections of graphic characters. See also font metrics and relative metrics.

flash
>See forms flash.

FLU
>See Font Library Update.

FOCA
>See Font Object Content Architecture.

font

>1. A family or assortment of characters of a given size and style, for example, 9-point Bodoni modern. A font has a unique name and might have a registry number.

>2. A particular type style (for example, Bodoni or Times Roman) that contains definitions of character sets, marker sets, and pattern sets. See also coded font and double-byte coded font.

font character set

1. Part of an AFP font that contains the raster patterns, identifiers, and descriptions of characters. See also character set.
2. A Font Object Content Architecture (FOCA) resource containing descriptive information, font metrics, and the digital representation of character shapes for a specified graphic character set.

Font Library Update (FLU)
An MVS-based utility processed against the AFP font library to ensure that the appropriate GRID information is available for processing COM text.

font mapping
Internal tables that compare and match core raster fonts to core outline fonts.

font metrics
Measurement information that defines individual character values, such as height, width, and space, as well as overall font values, such as averages and maximums. Font metrics can be expressed in specified fixed units, such as pels, or in relative units that are independent of both the resolution and size of the font. See also fixed metrics and relative metrics.

Font Object Content Architecture (FOCA)
An architecture that defines the content of digital font resources by means of a set of parameter definitions.

font section
A subdivision of a double-byte font character set. The section consists of a maximum of 256 characters. See also coded font section.

font typeface global identifier (FGID)
A unique font identifier that can be expressed as either a 2-byte binary value or a 5-digit decimal value. The FGID is used to identify a type style and the following characteristics: posture, weight, and width.

font width (FW)
A characteristic value of a font, parallel to the character baseline, that represents the size of all graphic characters in a font.

form

1. A physical piece of paper or other medium on which data is printed. See also medium, page, and sheet.
2. A display screen, printed document, or file with defined spaces for information to be inserted.

format
The shape, size, printing requirements, and general makeup of a printed document or presentation display.

format resolution
A value in pels per inch that indicates the resolution in which the data stream is formatted. PSF uses this information to determine which resolution system library to use.

formatted print records
Traditional line data made up of records that are formatted for printing on line printers. PSF uses a page definition to print formatted records on page printers.

form definition
An AFP resource object used by PSF that defines the characteristics of the form or printed media, including: overlays to be used, duplex printing, text suppression, the position of composed-text data on the form, and the number and modifications of a page.

forms control buffer (FCB)
A buffer for controlling the vertical format of printed output. The FCB is a line-printer control that is similar to the punched-paper, carriage-control tape used on IBM 1403 printers. For AFP page printers, the forms control buffer is replaced by the page definition. See also page definition.

forms flash
   In AFP support on the 3800 Printing Subsystem, a means of printing an overlay by using a negative plate projected on a form.

frame
   A rectangular area on microfilm, bounded by imaginary intersecting, orthogonal lines that form a grid pattern, within which data can be recorded. The grid pattern does not actually appear on the microfilm.

FSA
   See functional subsystem application.

FSI
   See functional subsystem interface.

FSS
   See functional subsystem.

functional subsystem (FSS)
   An extension of JES that runs in an address space separate from the JES address space. An FSS provides support for an auxiliary function to JES processing, such as a peripheral device or other component.

functional subsystem application (FSA)

   1. An area within the functional subsystem (FSS) that drives and manages a single printer. FSAs are identified with JES printer definitions.

   2. An application that uses the support facilities of the functional subsystem (FSS) to communicate with JES.

functional subsystem interface (FSI)
   A set of services that allows communication between the JES address space or DPSS and the PSF functional subsystem.

FW
   See font width.

G

gateway
   A device or program used to connect networks or systems with different network architectures. The systems can have different characteristics, such as different communication protocols, different network architecture, or different security policies, in which case the gateway performs a translation role as well as a connection role.

GCSGID
   See graphic character set global identifier.

global resource identifier (GRID)
   An 8-byte identifier that identifies a coded font resource. A GRID contains these fields, in the order listed:

   1. GCSGID of a minimum set of graphic characters required for presentation. It can be a character set that is associated with the code page, with the font character set, or with both.

   2. CPGID of the associated code page.

   3. FGID of the associated font character set.

   4. Font width (FW), in 1440ths of an inch.

glyph

   1. A graphic symbol whose appearance conveys information, for example, the vertical and horizontal arrows on cursor keys that indicate the directions in which they control cursor movement.

   2. An image, typically of a character, in a font. See also character and graphic character.

GOCA
    See Graphics Object Content Architecture.

graphical user interface (GUI)
    A type of computer interface that presents a visual metaphor of a real-world scene, often of a desktop, by combining high-resolution graphics, pointing devices, menu bars and other menus, overlapping windows, icons and the object-action relationship. See also programming interface for customers.

graphic character

1. A visual representation of a character, other than a control character, that is typically produced by writing, printing, or displaying. See also glyph.

2. A member of a set of symbols that represent data. Graphic characters can be letters, digits, punctuation marks, or other symbols.

graphic character set global identifier (GCSGID)
    A unique graphic character set identifier that can be expressed as either a 2-byte binary value or a 5-digit decimal value.

Graphics Object Content Architecture (GOCA)
    An architecture that provides a collection of graphics values and control structures used to interchange and present graphics data.

GRID
    See global resource identifier.

guaranteed print labeling
    A method of print labeling that ensures the integrity of the identification label by preventing the user from changing the label. See also nonguaranteed print labeling and identification label.

GUI
    See graphical user interface.

H

hardcopy
    A printed copy of machine output in a visually readable form, such as printed reports, documents, and summaries. See also softcopy.

hard page segment
    A page segment that is declared in the Map Page Segment structured field and loaded in the printer before printing begins. This resource can be reused during the job without being reloaded in the printer. Hard page segments can be controlled by a page segment list in a page definition. See also soft page segment.

hard resource
    A resource declared in the appropriate Map structured field and loaded in the printer the first time it is referenced. It can be reused during the job without being reloaded to the printer. See also soft resource.

hardware configuration definition (HCD)
    An interactive interface in the z/OS operating system that is used to define hardware configurations to the operating system and the channel subsystem.

hardware default font
    The font used by the printer if no other font is specified.

HCD
    See hardware configuration definition.

hexadecimal
    Pertaining to a numbering system that has a base of 16.

HFS
    See hierarchical file system.

hierarchical file system (HFS)
    A system for organizing files in a hierarchy, as in a UNIX system.

host

1. A computer that is connected to a network and provides an access point to that network. The host can be a client, a server, or both a client and server simultaneously. See also client and server.

2. In TCP/IP, any system that has at least one Internet Protocol address associated with it.

host address
See Internet Protocol address.

host font
See host resource.

host name
The network name given to a computer. Sometimes, host name is used to mean the fully qualified domain name; other times, it is used to mean the most specific subname of a fully qualified domain name. For example, if `mycomputer.city.company.com` is the fully qualified domain name, either of these host names can be used: `mycomputer.city.company.com` or `mycomputer`. See also colon hexadecimal notation and dotted decimal notation.

host processor
A processor that controls a user application network.

host resource
A resource found either in a system library, in a user library, or inline in the print data set.

host system
See host.

I

ID
See identifier.

identification label
In PSF, a protected set of resources (fonts, overlays, and page segments) that are used to label PSF output for security purposes. See also security label.

identifier (ID)
A sequence of bits or characters that identifies a user, program, device, or system to another user, program, device, or system.

image

1. A pattern of toned and untoned pels that form a picture. See also impression.

2. An electronic representation of an original document or picture produced by a scanning device or created from software.

image data

1. A pattern of bits with 0 and 1 values that define the pels in an image. A 1-bit is a toned pel.

2. Digital data derived from electrical signals that represent a visual image.

3. Rectangular arrays of raster information that define an image.

Image Object Content Architecture (IOCA)
An architecture that provides a collection of constructs used to interchange and present images, such as printing image data on a page, page segment, or overlay.

IML
See initial machine load or initial microprogram load.

IMM
See invoke medium map.

impact printer
A printer in which printing is the result of mechanically striking the printing medium. See also nonimpact printer.

impression
The transfer of an image to a sheet of paper. Multiple impressions can be printed on each side of a sheet. Printer speed is often measured in impressions per minute (ipm).

Infoprint Server
An element of z/OS that supports printing on local printers and remote printers in an Internet Protocol or SNA network. With Infoprint Server, users can submit print requests from remote workstations in an Internet Protocol network, from z/OS UNIX System Services applications, from batch applications, from VTAM applications (such as CICS® or IMS), and from SAP R/3.

initialize

1. In programming languages, to set the starting value of a data object.
2. To set the addresses, switches, or the contents of storage to zero, or to the starting value set by the manufacturer.
3. To prepare a system, device, or program for operation; for example, to initialize a diskette. See also initial program load.

initial machine load (IML)
A procedure that prepares a device for use.

initial microprogram load (IML)
The action of loading microcode into computer storage.

initial program load (IPL)

1. The process that loads the system programs from the system auxiliary storage, checks the system hardware, and prepares the system for user operations.
2. The process of loading the operating system and other basic software into main storage.

inline
Pertaining to spooled input data that is read into a job by a reader. See also inline resource.

inline direction
The direction in which successive characters are added to a line of text.

inline resource
A resource contained in a print file or a print data set.

input/output (I/O)
Pertaining to a device, process, channel, or communication path involved in data input, data output, or both.

installation exit
The means specifically described in an IBM software product's documentation by which an IBM software product can be modified by a customer's system programmers to change or extend the functions of the IBM software product. Such modifications consist of exit routines written to replace one or more existing modules of an IBM software product, or to add one or more modules or subroutines to an IBM software product.

Intelligent Printer Data Stream (IPDS)
An all-points-addressable data stream that lets users position text, images, graphics, and bar codes at any defined point on a printed page. IPDS is the strategic AFP printer data stream generated by PSF.

interface
A shared boundary between independent systems. An interface can be a hardware component used to link two devices, a convention that supports communication between software systems, or a method for a user to communicate with the operating system, such as a keyboard.

interleaving

1. The simultaneous accessing of two or more bytes or streams of data from distinct storage units.
2. The alternating of two or more operations or functions through the overlapped use of a computer facility.

intermediate device
A device that operates on the data stream and is situated between a printer and a presentation services program in the host. Examples include devices that capture and store resources and devices that spool the data stream.

internal copy group
A copy group in a print data set instead of in a form definition. See also copy group.

internal medium map
See internal copy group.

internal object
A structured field that can be included as part of a resource or a print job (data set or file), but that cannot be accessed separately.

Internet Protocol (IP)
A protocol that routes data through a network or interconnected networks. This protocol acts as an intermediary between the higher protocol layers and the physical network. See also Transmission Control Protocol and Transmission Control Protocol/Internet Protocol.

Internet Protocol (IP) address
A unique address for a device or logical unit on a network that uses the IP standard. See also colon hexadecimal notation, dotted decimal notation, and host name.

intervention-required condition
An error that causes printing to stop until an operator performs a required action.

invoke medium map (IMM)
A structured field that is used to switch copy groups within a form definition.

I/O
See input/output.

IOCA
See Image Object Content Architecture.

IP
See Internet Protocol.

IP address
See Internet Protocol address.

IPDS
See Intelligent Printer Data Stream.

IPL
See initial program load.

J

JCL
See job control language.

JES
See Job Entry Subsystem.

JES2
An MVS subsystem that receives jobs into the system, converts them to internal format, selects them for processing, processes their output, and purges them from the system. In an installation with more than one processor, each JES2 processor independently controls its job input, scheduling, and output processing. See also Job Entry Subsystem and JES3.

JES3
An MVS subsystem that receives jobs into the system, converts them to internal format, selects them for processing, processes their output, and purges them from the system. In complexes that have several loosely coupled processing units, the JES3 program manages processors so that the global processor exercises centralized control over the local processors and distributes jobs to them by using a common job queue. See also Job Entry Subsystem and JES2.

job control language (JCL)
A command language that identifies a job to an operating system and describes the job's requirements.

Job Entry Subsystem (JES)
An IBM licensed program that receives jobs into the system and processes all output data that is produced by jobs. See also JES2 and JES3.

job header
A page in printed output that indicates the beginning of a user job. A user job can contain one or more data sets, or one or more copies of a print job. See also data set header.

job separation
The ability to identify job boundaries by placing marks or sheets of paper between successive jobs.

job trailer
A page in the printed output that indicates the end of a user job.

K

Kanji
A graphic character set consisting of symbols used in Japanese ideographic alphabets. Each character is represented by 2 bytes.

L

LAN
See local area network.

landscape page presentation
The position of a printed sheet that has its long edges as the top and bottom and its short edges as the sides. See also portrait page presentation.

LASI
See library access system interface.

LCPC
Load code page control command.

LFE
See load font equivalence.

library

1. A system object that serves as a directory to other objects. A library groups related objects, and allows the user to find objects by name.

2. A data file that contains copies of a number of individual files and control information that allows them to be accessed individually.

3. A partitioned data set or a series of concatenated partitioned data sets.

library access system interface (LASI)
The PSF subcomponent that gets resources from the libraries and stores and obtains records in a message data set.

library member
A named collection of records or statements in a library. See also resource object.

ligature
Two or more characters that are connected so they appear as one character. For example, `ff` and `ffi` are characters that can be presented as ligatures.

line data
Data prepared for printing on a line printer without any data placement or presentation information. Line data can contain carriage-control characters and table-reference characters (TRC) for spacing and font selections. See also record format line data and traditional line data.

line-mode printing
The ability for a printer to operate in line mode as a non-AFP printer. Print jobs that run in line mode do not use AFP resources, such as page definitions or form definitions. Instead, they use 3800 line-mode resources, such as FCBs and GRAPHMODs.

line printer
A device that prints a line of characters as a unit. See also page printer.

lines per inch (lpi)

1. The number of characters that can be printed vertically within an inch.

2. A unit of measurement for specifying the placement of the baseline.

load font equivalence (LFE)
The mapping of a descriptive font name to a font member name in a font library.

local area network (LAN)
A network that connects several devices into a limited area (such as a single building or campus) and that can be connected to a larger network.

local-attached
Pertaining to an SNA-attached device that does not have a communications controller in its configuration. See also communication-attached.

logical page
The defined presentation space on the physical form. All the text and images in the print data must fit within the boundaries of the logical page, which has specified characteristics, such as size, shape, orientation, and offset. See also form and physical page.

logical page origin
The point on the logical page from which positions of images, graphics, page overlays, and text with 0-degree inline direction are measured.

logical unit (LU, L-unit)

1. A unit of linear measurement. For example, in Mixed Object Document Content Architecture (MO:DCA) and AFP data streams, these measurements are used:

   1 L-unit = 1/1440 inch
   1 L-unit = 1/240 inch

2. An access point through which a user or application program accesses the SNA network to communicate with another user or application program. An LU can support at least two sessions, one with an SSCP and one with another LU, and might be capable of supporting many sessions with other LUs.

logical unit (LU) base
The linear measurement base. The value defines, for any object, the meaning of the logical unit values. See also logical unit.

logon mode
In VTAM, a subset of session parameters specified in a logon-mode table for communication with a logical unit.

logon-mode table
In VTAM, a set of entries for macro-generated constants that associate a logon-mode name with a set of session parameters.

lpi
See lines per inch.

LU
See logical unit.

L-unit
See logical unit.

LU type 1
An SNA logical unit type that provides a communication protocol among host application programs and terminals. Some printers also use this protocol to communicate with host application programs.

LU type 6.2
An SNA logical unit type that converges functions from existing LU types to provide a single, interchangeable communication protocol. See also Page Printer Communication Component.

M

macro
An instruction that causes the execution of a predefined sequence of instructions.

magnetic ink character recognition (MICR)
The identification of characters through the use of magnetic ink.

mapping
A list, typically in a profile, that establishes a correspondence between items in two groups. For example, a keyboard mapping can establish what character is displayed when a certain key is pressed. See also font mapping.

marking
A method of updating certain structured fields to identify a resource as printer-resident.

maximum transmission unit (MTU)
The largest possible unit of data that can be sent on a given physical medium in a single frame. For example, the maximum transmission unit for Ethernet is 1500 bytes.

media origin
The reference point from which the logical page origin is positioned by the medium map. This point is represented by Xm=0, Ym=0 in the Xm, Ym coordinate system. The media origin is defined relative to the upper-left corner of the form. See also logical page origin. See also logical page origin.

medium

1. The material on which computer information is stored. Examples of media are diskettes, CDs, DVDs, and tape.

2. The physical material, such as paper, on which data is printed. See also form, page, and sheet.

medium map
See copy group.

medium overlay
An electronic overlay that is called by the medium map of a form definition for printing at a fixed position on the form. See also page overlay.

member name
The name under which a file is stored in a library. For example, X1BITR is the member name of a font in the font library.

message data set

1. In PSF, a virtual data set built by the library access system interface (LASI) subcomponent in memory to store error messages for printing at the end of the document.

2. A data set on disk storage that contains queues of messages awaiting transmission to particular terminal operators or to the host system.

metrics
See font metrics.

MICR
See magnetic ink character recognition.

microfiche
A sheet of microfilm containing a photographic record on a reduced scale of printed matter.

microfilm
A film containing a photographic record of printed matter, on a reduced scale.

microfilm device
An output device that presents a hardcopy on microfilm.

microfilm setup resource
A setup file that contains information used to present AFP data on microfilm. See also object container.

microfilm utility
A utility that builds an object container for microfilm setup data and places it in the AFP library.

migration
The movement of data when software is upgraded or the data is transferred to a different hardware server or model.

Mixed Object Document Content Architecture (MO:DCA)
An architected, device-independent data stream for interchanging documents.

Mixed Object Document Content Architecture for Presentation (MO:DCA-P)
The subset of MO:DCA that defines presentation documents. PSF supports MO:DCA Presentation Interchange Set data streams.

MO:DCA
See Mixed Object Document Content Architecture.

MO:DCA AFP/Archive (MO:DCA AFP/A)
An AFP document architecture interchange set that is used for long-term preservation and retrieval. This subset ensures page independence and eliminates images without clearly specified resolution, device default fonts, and external resources.

MO:DCA AFP/A
See MO:DCA AFP/Archive.

MO:DCA AFP/A, IS/3
An AFP document architecture interchange set that complies with the rules and restrictions of both the AFP/Archive and IS/3 interchange sets.

MO:DCA data
Print data that has been composed into pages. Text-formatting programs (such as DCF) can produce composed text data consisting entirely of structured fields. ACIF or AFP Download Plus can transform line data or XML data to MO:DCA data.

MO:DCA data definition
A resource containing a set of formatting controls for printing logical pages of data. Includes controls for the number of lines per printed sheet, font selection, and print direction, and for mapping individual fields in the data to position on the printed sheets.

MO:DCA document
Data composed entirely of stuctured fields and containing a Begin Document structured field and an End Document structured field.

MO:DCA GA
See MO:DCA Graphic Arts Function Set.

MO:DCA Graphic Arts Function Set (MO:DCA GA)
An extension of MO:DCA IS/3 that provides support for PDF presentation object containers.

MO:DCA IS/1
See MO:DCA Presentation Interchange Set 1.

MO:DCA IS/3
See MO:DCA Presentation Interchange Set 3.

MO:DCA-P
See Mixed Object Document Content Architecture for Presentation.

MO:DCA Presentation Interchange Set 1 (MO:DCA IS/1)
A subset of MO:DCA that defines an interchange format for presentation documents.

MO:DCA Presentation Interchange Set 3 (MO:DCA IS/3)
A subset of MO:DCA that defines an interchange format for presentation documents. The MO:DCA IS/3 data stream includes structured fields that are not found in MO:DCA IS/1.

MTU
See maximum transmission unit.

multiple-entry font
A font with multiple entries in the Map Coded Font (MCF) structured field. The only fonts that have multiple entries are double-byte fonts that are defined dynamically. (The MCF points directly to a set of code page and font character set pairs.) See also single-entry font.

multiple up
The printing of more than one page on a single surface of a sheet of paper.

Multiple Virtual Storage (MVS)
An IBM operating system that accesses multiple address spaces in virtual storage.

multiplexer
A device that takes several input signals and combines them into a single output signal in such a manner that each of the input signals can be recovered.

multiplexing
In data transmission, a function that enables two or more data sources to share a common transmission medium so that each source has its own channel.

multipoint network
More than two devices sharing the same transmission line at the same time.

MVS
See Multiple Virtual Storage.

MVSCP
MVS Configuration Program.

N

NACK
See negative acknowledgment reply.

NCP
See Network Control Program.

NCP generation
The process by which the host processor assembles and link-edits a macroinstruction to produce a network control program.

negative acknowledgment reply (NACK)
A reply from a printer to a host indicating that an exception has occurred. See also positive acknowledgment reply

nested resource
A resource mapped in an overlay.

Network Control Program (NCP)
An IBM licensed program that provides communication controller support for single-domain, multiple-domain, and interconnected network capability.

nonguaranteed print labeling
Some printers that do not support guaranteed print labeling can still print identification labels, but the integrity of the labels cannot be guaranteed by using PSF. See also guaranteed print labeling.

nonimpact printer
A printer in which printing is not the result of mechanical impacts, for example, a thermal printer, an electrostatic printer, and a photographic printer. See also impact printer.

nonprocess runout (NPRO)
An operation that moves paper through the paper path without printing.

nonswitched line
A connection between computers or devices that does not have to be made by dialing. The line is permanently connected and always available. See also switched line.

normal duplex
Pertaining to printing on both sides of the paper such that the top of one side is at the same end as the top of the other side. Normal duplex printing is used for forms that are bound on the long edge of the paper, regardless of whether the printing is portrait or landscape. See also duplex and tumble duplex.

notify subtask (NST)

1. A PSF subcomponent that returns processed data sets to JES or to the DPSS and performs checkpoint processing on data sets as they are printed.

2. An external trace that contains information pertaining to the releasing and checkpointing of data sets by PSF.

NPRO
See nonprocess runout.

NST
See notify subtask.

null value
A parameter position for which no value is specified.

N_UP
The partitioning of a side of a sheet into a fixed number of equal size partitions. For example, N_UP 4 divides each side of the sheet into four equal partitions. In enhanced N_UP printing, the sheet can be divided into 8 partitions, each of which can be anywhere on a single side of the sheet.

O

object
In AFP architecture, a collection of structured fields, bounded by a begin-object function and an end-object function. The object can contain other structured fields containing data elements of a particular type. Examples of objects are text, fonts, graphics, images, and bar codes.

object container
A MO:DCA structure that carries object data, which might or might not be defined by a presentation architecture.

offset stacking
A function that allows the printed output pages to be offset for easy separation of the print jobs.

OGL
See Overlay Generation Language.

OpenType font
An extension of the TrueType font format that adds support for PostScript outlines and more support for international character sets and advanced typographic control.

option
A specification in a statement that can influence the running of the statement.

orientation
In printing, the number of degrees an object is rotated relative to a reference; for example, the orientation of an overlay relative to the logical page origin, or the orientation of printing on a page relative to the page coordinates. Orientation typically applies to blocks of information, whereas character rotation applies to individual characters. See also character rotation.

origin

1. A position from which the placement and orientation of an element is specified.

2. The point in a coordinate system where the axes intersect. Examples of origins are the addressable position in an X_m,Y_m coordinate system where both coordinate values are zero and the character reference point in a character coordinate system.

outline font
A font whose graphic character shapes are defined by mathematical equations rather than by raster patterns. See also raster font.

overlay
1. A resource object that contains predefined presentation data, such as text, image, graphics, and bar code data, that can be merged with variable data on a page or form while printing. See also page overlay and medium overlay.
2. The final representation of a collection of predefined presentation data on a physical medium.

Overlay Generation Language (OGL)
An IBM licensed program used for designing objects (such as lines, boxes, shadings, and irregular shapes) for electronic overlays.

P

packet
In data communication, a sequence of binary digits, including data and control signals, that is transmitted and switched as a composite whole.

page
1. A collection of data that can be printed on one side of a sheet of paper or a form.
2. A data stream object delimited by a Begin Page structured field and an End Page structured field. A page can contain presentation data such as text, image, graphics, and bar code data. See also logical page and physical page.

page definition
An AFP resource object used by PSF that defines the rules for transforming line data and XML data into MO:DCA data and text controls, such as width of margins and text orientation.

page mode
The mode of operation in which a page printer can accept an entire page of data from a host processor to be printed on an all-points-addressable output medium. A page of data can consist of text, images, overlays, and page segments.

page origin
See logical page origin.

page overlay
An electronic overlay that can be called for printing and positioned at any point on the page by an Invoke Page Overlay structured field in the print data. See also medium overlay.

page position
A control in the copy group to assign the upper-left boundary point of the logical page on a sheet for a data set. The page position is determined from the media origin.

page printer
1. In AFP support, any of a class of printers that accepts composed pages, constructed of composed text and images, among other things. See also line printer.
2. A device that prints one page at a time.

Page Printer Communication Component (PPCC)
The access method that provides the SNA communication interface between printers and PSF.

Page Printer Formatting Aid (PPFA)
An IBM licensed program with which to create and store form definitions and page definitions, which are resource objects used for print-job management. These stored objects are used to format printed output.

page segment
An AFP resource object containing text, image, graphics, or bar code data that can be positioned on any addressable point on a page or an electronic overlay.

paging
The process of transferring instructions, data, or pages between real storage and external page storage.

parameter
A value or reference passed to a function, command, or program that serves as input or controls actions. The value is supplied by a user or by another program or process.

partition
In basic N_UP printing, the division of the medium presentation space into a specified number of equal-sized areas in a manner determined by the current physical medium.

partitioned data set (PDS)
A data set in direct-access storage that is divided into partitions, called members, each of which can contain a program, part of a program, or data. See also sequential data set.

pattern storage (PST)

1. An area of storage that holds the raster patterns for fonts and images.

2. In printers, storage that contains the fonts and images used for printing the character data, overlays, and segments contained in the page buffer.

PCL
See Printer Control Language.

PDS
See partitioned data set.

pel
See picture element.

physical medium
A physical entity on which information is presented; for example, a sheet of paper, a roll of paper, microfilm, an envelope, label, or display screen.

physical page
A single surface (front or back) of a form. See also form, logical page, and page.

physical unit (PU)
In SNA, one of three types of network addressable units. A PU exists in each node of an SNA network to manage and monitor, at the request of an SSCP logical unit session, the resources (such as attached links and adjacent link stations) of a node.

picture element (pel, pixel)

1. An element of a raster pattern about which a toned area on the photoconductor might appear. When used with a number, *pel* indicates resolution. Examples include 240-pel and 300-pel.

2. The smallest printable or displayable unit that can be displayed. A common measurement of device resolution is picture elements per inch. Typical monitors display between 72 and 96 pixels per inch. Characters and graphics are created by turning pixels on or off.

ping
To verify that a host system can communicate with a TCP/IP-attached printer.

pitch
A unit of measurement for the width of type (or a printed character), based on the number of characters that can be set (or printed) in one linear inch; for example, 10-pitch has 10 characters per inch. Uniformly spaced fonts are measured in pitch. See also point.

pixel
See picture element.

PLR
See priority level resource.

point
A unit of measurement used mainly for describing type sizes. Each pica has 12 points, and an inch has approximately 72 points. See also pitch.

point size
The height of a font in points.

portrait page presentation
The position of a printed sheet that has its short edges as the top and bottom and its long edges as the sides. See also landscape page presentation.

positive acknowledgment reply (ACK)
A reply from a printer to a host in which no exception condition is reported. See also negative acknowledgment reply.

PostScript
A page description language developed by Adobe Systems, Incorporated that describes how text and graphics are presented on printers and display devices.

PPCC
See Page Printer Communication Component.

PPFA
See Page Printer Formatting Aid.

presentation text
See composed text.

Presentation Text Object Content Architecture (PTOCA)
An architecture that provides a collection of constructs used to interchange and present presentation text data, such as printing text data on a page, page segment, or overlay.

printable area
The area on a sheet of paper where print can be placed.

print data set
A data set created by an application program that contains the actual information to be printed and, optionally, some of the data that controls the format of the printing. The types of print data sets are composed text, line format, XML data, and mixed format. See also auxiliary data set and print file.

print data stream
The data stream that is created by PSF and transmitted to the printer.

print direction
A combination of the inline direction and the baseline direction.

Printer Control Language (PCL)
The Hewlett Packard page description language that is used in laser and ink-jet printers.

Printer Inventory
In Infoprint Server, a set of files that contain information about printers. The Printer Inventory includes such objects as printer definitions, functional subsystem (FSS) definitions, and job selection rules for IP PrintWay.

print file
A file that is created for the purpose of printing data. A print file includes information to be printed and, optionally, some of the data that controls the format of the printing. See also print data set.

print job
One or more documents submitted in the same job to be printed on the same printer.

print labeling
A controlled method of placing identification labels on each page of PSF printed output.

print position
Any location on a medium where a character can be printed.

print quality
The measure of the quality of printed output relative to existing standards and in comparison with jobs printed previously.

Print Services Facility (PSF)
An IBM licensed program that manages and controls the input data stream and output data stream required by supported page printers.

priority level resource (PLR)
A marked resource known to PSF and deleted according to a modified deletion algorithm. When resources are loaded according to reasonable resource loaded value, PSF deletes all non-PLR resources before deleting any PLR resources. A frequently used resource can be marked as PLR with the Resource Exit; this improves its chances of being retained at the data set end.

processor
In a computer, the part that interprets and processes instructions. Two typical components of a processor are a control unit and an arithmetic logic unit.

programming interface for customers
Any product method that lets a customer-written program obtain the services of the product (for example, CSECT names, data areas or control blocks, data sets or files, exits, macros, parameter lists, and programming languages). Not all products have programming interfaces for customers; some products provide their services through graphical user interfaces, while others provide their services only to other products. See also graphical user interface.

programming request for price quotation (PRPQ)
A customer request for a price quotation on alterations or additions to the functional capabilities of system control programming or licensed programs. The PRPQ can be used in conjunction with computing system RPQs to solve unique data processing problems. See also computing system RPQ.

program temporary fix (PTF)
For System i®, System p, and IBM Z products, a package containing individual or multiple fixes that is made available to all licensed customers. A PTF resolves defects and might provide enhancements.

protocol
A set of rules controlling the communication and transfer of data between two or more devices or systems in a communications network.

PRPQ
See programming request for price quotation.

PSF
See Print Services Facility.

PSF direct
A function of an AIX or Windows print server that enables another PSF program to print remotely.

PST
See pattern storage.

PTF
See program temporary fix.

PTOCA
See Presentation Test Object Content Architecture.

PU
See physical unit.

R

RACF
See Resource Access Control Facility.

raster font
A font in which the characters are defined directly by the raster bit map. See also outline font.

raster pattern
A series of picture elements (pels) arranged in scan lines to form an image. The toned or untoned status of each pel creates an image. A digitized raster pattern is an array of bits. The on or off status of each bit determines the toned or untoned status of each pel.

raster pattern storage (RPS)
An area of storage that holds raster patterns for fonts and images.

RAT
See resource access table.

RDW
See record descriptor word.

real storage
The main storage in a virtual storage system. Physically, real storage and main storage are identical. Conceptually, however, real storage represents only part of the range of addresses available to the user of a virtual storage system.

reasonable resource loaded value (RRLV)
A value used by PSF to keep the number of resources loaded at a manageable level. At data set end, PSF deletes resources for a particular resource type until this value is reached.

received page counter
See channel counter.

record descriptor word (RDW)
Data preceding a variable record or a structured field that specifies the length of the entire record including the RDW.

record format line data
A form of line data where each record is preceded by a 10-byte identifier. See also line data.

region area
Storage above the 16 MB line and below the 2 GB bar. By default, inline resources are stored in the region area during job processing. See also above the bar storage.

region size
The amount of main storage available for a program to run.

relative metrics
Measurement information that is defined in relation to some other units. Relative values are expressed as fractional parts of a unit-square design space (em square), whose sides correspond to the vertical size of the font. See also fixed metrics and font metrics.

reload function
A process in which PSF loads a resource from a host library instead of using an existing version of the resource.

repositioning
A process in which PSF, following an indication from the printer of a potentially recoverable error, locates the proper spool record for recomposing one or more pages for printing.

Request for Comments (RFC)
In Internet communication, one of a series of numbered documents that describe Internet communication protocols.

request for price quotation (RPQ)
A customer request for a price quotation on alterations or additions to the functional capabilities of a hardware product for a computing system or a device. See computing system RPQ and programming request for price quotation.

request unit (RU)
In SNA, a message unit that contains data, control information, or both (for example, data and indicators).

resident resource
A resource, such as a font, symbol set, page segment, or overlay, that resides in a printer or an intermediary device, such as a personal computer.

resolution
A measure of the sharpness of an image, expressed as the number of lines per unit of length or the number of points per unit of area discernible in that image.

resource

A collection of printing instructions used, in addition to the print data set, to produce the printed output. Resources include coded fonts, font character sets, code pages, page segments, overlays, form definitions, and page definitions.

Resource Access Control Facility (RACF)

An IBM licensed program that provides for access control by identifying users to the system, verifying users of the system, authorizing access to protected resources, logging unauthorized attempts to enter the system, and logging accesses to protected resources.

resource access table (RAT)

An array of data that is used to map a resource name specified in the MO:DCA data stream to information used to find and process the resource on a given system.

resource-constrained condition

The situation that occurs when the printer does not have enough storage for the resources required to print the current page.

resource manager

An application, program, or transaction that manages and controls access to shared resources, such as memory buffers and data sets.

resource name

The name under which an AFP resource object is stored, the first 2 characters of which indicate the resource type.

resource object

In AFP, a collection of printing instructions, and sometimes data to be printed, that consists entirely of structured fields. A resource object is stored as a member (or file) of a library and can be called for by PSF when needed. The different resource objects include: coded font, font character set, code page, page segment, overlay, form definition, and page definition. See also library member.

RFC

See Request for Comments.

root

Pertaining to the user name for the system user with the most authority.

rotation

The number of degrees a graphic character is turned relative to the page coordinates. See character rotation. See also orientation.

router

1. A computer that determines the path of network traffic flow. The path selection is made from several paths based on information obtained from specific protocols, algorithms that attempt to identify the shortest or best path, and other criteria such as metrics or protocol-specific destination addresses.

2. An attaching device that connects two LAN segments at the reference model network layer. The LAN segments can use similar or different architectures.

routine

1. A set of statements in a program that causes the system to perform an operation or a series of related operations.

2. A program or sequence of instructions called by a program. Typically, a routine has a general purpose and is frequently used.

RPQ

See request for price quotation.

RPS

See raster pattern storage.

RRB

Record resource block.

RRLV

See reasonable resource loaded value.

RU

See request unit.

S

SDLC

See Synchronous Data Link Control.

SDSF

See System Display and Search Facility.

security label

In a trusted computing base, a label used to maintain multiple levels of security on a system. This label is a combination of a security class and a security level. See also identification label.

security overlay

An overlay, such as one created by use of Overlay Generation Language, that resides in a secure library and is used to place security resources on a page.

segment

A collection of composed text and images, prepared before formatting and included in a document when it is printed. See also page segment.

sense data

In printers, information used to indicate the causes of command-stream and device exceptions and to direct the host program to the appropriate exception-recovery actions.

sequence number

A 2-byte field in the structured field introducer that identifies the position of the structured field in the data set.

sequential data set

A data set whose records are organized on the basis of their successive physical positions, such as on magnetic tape. See also partitioned data set.

server

A software program or a computer that provides services to other software programs or other computers. The program or computer making the request of the server is typically called the client. See also client and host.

service program

See utility program.

sheet

A division of the physical medium; multiple sheets can exist on a physical medium. For example, a roll of paper might be divided by a printer into rectangular pieces of paper, each representing a sheet. Envelopes are an example of a physical medium that comprises only one sheet. The IPDS architecture defines four types of sheets: cut-sheets, continuous forms, envelopes, and computer output on microfilm. Each type of sheet has a top edge. A sheet has two sides, a front side and a back side. See also form.

shift-out, shift-in (SOSI)

Special EBCDIC or ASCII characters that exist in the data stream to indicate the switches between double-byte fonts and single-byte fonts.

Simple Network Management Protocol (SNMP)

A set of protocols for monitoring systems and devices in complex networks. Information about managed devices is defined and stored in a Management Information Base (MIB).

simplex

Pertaining to printing on only one side of the paper. See also duplex, normal duplex, and tumble duplex.

single-byte coded font
A font in which the characters are defined by a 1-byte code point. A single-byte coded font has only one coded font section. See also double-byte coded font.

single-entry font
Fonts with a single entry in the Map Coded Font structured field. This includes all single-byte fonts and those double-byte fonts being defined as coded fonts. See also multiple-entry font.

skip
1. A move of the current print position to another location.
2. To ignore one or more instructions in a sequence of instructions.
3. To pass over one or more positions on a data medium; for example, to perform one or more line feed operations.

SMF
See System Management Facilities.

SMF type 6 record
A record that PSF uses to record data for each print data set.

SMP/E
See System Modification Program/Extended.

SNA
See Systems Network Architecture.

SNA-attached
Pertaining to a device that is linked to the host system through VTAM or ACF/VTAM and uses an SNA protocol to transfer data. It does not need to be physically connected to the host; some printers are attached to a control unit, a communication controller, or both, and they can transfer data over telecommunication lines. For example, an IBM 3825 Page Printer attached to a communication controller that uses the LU 6.2 communication protocol to transfer data to a communication controller is considered an SNA-attached printer. See also channel-attached and TCP/IP-attached.

SNMP
See Simple Network Management Protocol.

softcopy
One or more files that can be electronically distributed, manipulated, and printed by a user. See also hardcopy.

soft page segment
A resource that is not declared in the Map Page Segment structured field but is sent to the printer inline as part of the overlay or page that includes it. See also hard page segment.

soft resource
A resource that is not declared in a Map structured field but is sent to the printer inline with data. It cannot be reused during the job without being reloaded to the printer. See also hard resource.

SOSI
See shift-out, shift-in.

specified resolution
See format resolution.

SSCP
See system services control point.

SSI
See subsystem interface.

stacker
An enclosure in a printer in which printed media is stacked.

startup procedure
A program used to start an application and to specify initialization parameters, libraries that contain system resources, and routing-control information.

storage

1. A functional unit in which data can be placed and retained, and from which it can be retrieved. See also virtual storage.

2. The location of saved information.

structured field

1. A self-identifying string of bytes and its data or parameters.

2. A mechanism that permits variable length data to be encoded for transmission in the data stream.

structured field introducer
The first 8 bytes of a structured field that indicate its length, type, and number.

subgroup
A set of modifications in a copy group that applies to a certain number of copies of a form. A copy group can contain more than one subgroup.

subnet mask
For internet subnetworking, a 32-bit mask used to identify the subnetwork address bits in the host portion of an IP address.

subsystem interface (SSI)
The means by which system routines request services of the master subsystem, a job entry subsystem, or other subsystems defined to the subsystem interface. See also functional subsystem interface.

supervisor call (SVC)
An instruction that interrupts the program being run and passes control to the supervisor so that it can perform the specific service indicated by the instruction.

suppression
A method used to prevent presentation of specified data. In AFP support, a page- and form-definition function that is used to identify fields in a print record that are not printed on selected pages of a document. See also text suppression.

SVC
See supervisor call.

swapping
A process that interchanges the contents of an area of real storage with the contents of an area in auxiliary storage.

switched line
In data communications, a connection between computers or devices that is established by dialing. See also nonswitched line.

switched major node
In VTAM, a major node whose minor nodes are physical units and logical units attached by switched SDLC links.

symbol set
A type of font that resides in a printer but has fewer attributes than can be specified for resident coded fonts. See also character set.

synchronous

1. Pertaining to two or more processes that depend on the occurrences of specific events, such as common timing signals. See also asynchronous.

2. Occurring with a regular or predictable time relationship.

Synchronous Data Link Control (SDLC)
A protocol for managing synchronous information transfer over a data link connection.

SYSIN
See system input stream.

SYSOUT
    See system output stream.

System Display and Search Facility (SDSF)
    An IBM licensed program that provides a menu-driven, full-screen interface that is used to obtain detailed information about jobs and resources in a system.

system input stream (SYSIN)
    A data definition (DD) statement used to begin an in-stream data set. See also system output stream.

system library
    A collection of data sets or files in which one or more system resources are stored. See also user library.

System Management Facilities (SMF)
    A component of z/OS that collects and records a variety of system and job-related information. Examples of information collected by SMF are statistics, accounting information, and performance data.

System Modification Program/Extended (SMP/E)
    An IBM licensed program that is used to install software and software changes on z/OS operating systems. In addition to providing the services of SMP, SMP/E consolidates installation data, it allows more flexibility in selecting changes to be installed, provides a dialog interface, and supports dynamic allocation of data sets.

system output stream (SYSOUT)
    A data definition (DD) statement used to identify a data set as a system output data set. See also system input stream.

system path library
    A path or set of paths for system UNIX files that contain font objects. See also user path library.

system services control point (SSCP)
    A focal point in an SNA network for managing configuration, coordinating network-operator and problem-determination requests, and providing directory support or other session services for network users. Multiple SSCPs, cooperating as peers, can divide the network into domains of control, with each SSCP controlling the physical and logical units in its domain.

Systems Network Architecture (SNA)
    The description of the logical structure, formats, protocols, and operational sequences for transmitting information units through the networks and controlling the configuration and operation of networks. The layered structure of SNA allows the ultimate origins and destinations of information (the users) to be independent of and unaffected by the specific SNA network services and facilities that are used for information exchange. See also Page Printer Communication Component.

T

table reference character (TRC)
    A numeric character corresponding to the order in which font character sets have been specified. The TRC is used to select a font character set during printing.

TCP
    See Transmission Control Protocol.

TCP/IP
    See Transmission Control Protocol/Internet Protocol.

TCP/IP-attached
    Pertaining to a device that is linked to an operating system through an Internet Protocol network and receives data from the system by using an application-layer protocol for IPDS printers. Some TCP/IP-attached printers require the i-data 7913 IPDS Printer LAN Attachment. See also channel-attached and SNA-attached.

telecommunication line
    The part of a data circuit external to the equipment that connects to a data-switching exchange. See also data link.

terminate
To stop the operation of a system, a device, or a program. Do not use. Use *end* or *stop* instead.

text
A sequence of characters that can be read by a person and encoded into formats such as ASCII that can be interpreted by a computer.

text control
Structured field data that control the format, placement, and appearance of text.

text orientation
A description of the appearance of text as a combination of print direction and character rotation.

text suppression
The intentional omission of portions of text in copy groups specified in the form definition.

throughput

1. The measure of the amount of work performed by a device, such as a computer or printer, over a period of time, for example, the number of jobs per day.

2. In data communications, the total traffic between stations over a period of time.

token ring
A network configuration in which tokens are passed in a circuit from node to node. A node that is ready to send data can capture the token and insert data for transmission.

trace

1. A record of the processing of a computer program or transaction. The information collected from a trace can be used to assess problems and performance.

2. A Db2® for z/OS facility that provides the ability to collect monitoring, auditing, performance, accounting, statistics, and serviceability (global) data.

traditional line data
A form of line data that is prepared for printing on a line printer. See also line data.

transaction identifier
A unique name that is assigned to a transaction and is used to identify the actions associated with that transaction.

transmission
The sending of data from one place for reception elsewhere.

Transmission Control Protocol (TCP)
A communications protocol used in the Internet and in any network that follows the Internet Engineering Task Force (IETF) standards for internetwork protocol. TCP provides a reliable host-to-host protocol in packet-switched communications networks and in interconnected systems of such networks. See also Internet Protocol.

Transmission Control Protocol/Internet Protocol (TCP/IP)
An industry-standard, nonproprietary set of communications protocols that provide reliable end-to-end connections between applications over interconnected networks of different types.

tray
See bin.

TRC
See table reference character.

TrueType font
A font format based on scalable outline technology in which the graphic character shapes are based on quadratic curves. The font is described with a set of tables contained in a TrueType font file.

tumble duplex
Pertaining to printing on both sides of the paper such that the top of one side is at the same end as the bottom of the other side. Tumble duplex printing is used for forms that are bound on the short edge of the paper, regardless of whether the printing is portrait or landscape. See also duplex, normal duplex, and simplex.

two-channel switch
A hardware feature with which an I/O device can be attached to two channels. A dynamic switch can be added, which makes it possible for both interfaces to be enabled at the same time with channel selection determined by programming.

typeface
All characters of a single type family or style, weight class, width class, and posture, regardless of size. An example is Helvetica bold condensed italic, in any point size. See also font.

U

UCS
See universal character set.

unbounded-box font
A font designed to use unbounded-character boxes. See also bounded-box font.

unformatted print records
Traditional line data made up of fields of data that have not been formatted into print lines. PSF uses a page definition to format these records for printing on page printers.

Unicode
A character encoding standard that supports the interchange, processing, and display of text that is written in the common languages around the world, plus some classical and historical texts. For example, the text name for $ is *dollar sign* and its numeric value is X'0024'. The Unicode standard has a 16-bit character set defined by ISO 10646.

universal character set (UCS)
A printer feature that permits the use of a variety of character arrays. See font.

UPA
See user printable area.

user library
A private print-resource library owned by an individual user, accessed only when the name is specified by the owner in a JCL statement. See also system library.

user path library
A private font library owned by an individual user, accessed only when the path name is specified by the owner in a JCL statement. See also system path library.

user printable area (UPA)
The area within the valid printable area (VPA) where user-generated data can print without causing an exception condition. See also valid printable area.

utility program
A computer program in general support of computer processes; for example, a diagnostic program, a trace program, or a sort program.

V

valid printable area (VPA)
The intersection of the current logical page or current overlay with the physical page in which printing is allowed. See also user printable area.

value
In programming, the alphabetic or numeric contents of a variable, parameter, special register, field, or storage location.

virtual storage
The storage space that can be regarded as addressable main storage by the user of a computer system in which virtual addresses are mapped into real addresses. The size of virtual storage is limited by the addressing scheme of the computer system and by the amount of auxiliary storage available, not by the actual number of main storage locations. See also storage.

Virtual Telecommunications Access Method (VTAM)
An IBM licensed program that controls communication and the flow of data in an SNA network. It provides single-domain, multiple-domain, and interconnected network capability.

VPA
See valid printable area.

VTAM
See Virtual Telecommunications Access Method.

W

workstation print manager (WPM)
A distributed print function, which is available in Japan only, that can be installed and used to print jobs from PSF.

write to operator (WTO)
An optional user-coded service that allows a message to be written to the system console operator informing the operator of errors and unusual system conditions that might need to be corrected.

WTO
See write to operator.

X

X-axis
In printing, an axis perpendicular to the direction in which the paper moves through the printer. See also Y-axis.

X-extent
A measurement along the X-axis.

XML data
Data identified with the Extensible Markup Language (XML), which is a standard metalanguage for defining markup languages that is based on Standard Generalized Markup Language (SGML). For printing on page printers, a page definition is required to provide the data placement and presentation information. The XML data processed by PSF can be encoded in EBCDIC, ASCII, UTF-8 or UTF-16.

Xm, Ym coordinate system
The media coordinate system.

Xp, Yp coordinate system
The logical page coordinate system that describes the size, position, and orientation of a logical page.

Y

Y-axis
In printing, an axis parallel with the direction in which the paper moves through the printer. See also X-axis.

Y-extent
A measurement along the Y-axis.

Z

zFS
See z/OS File System.

z/OS
An IBM mainframe operating system that uses 64-bit real storage.

z/OS File System (zFS)
A type of file system that resides in a Virtual Storage Access Method (VSAM) linear data set (LDS). zFS contains files and directories that can be used by z/OS UNIX System Services to provide data access over IP networks.

z/OS Font Collection

A base element of z/OS V2R2 or later that contains a comprehensive set of fonts, including AFP outline fonts, AFP raster fonts, and WorldType fonts (TrueType and OpenType fonts). The recommended source of AFP fonts for printing with PSF.

Bibliography

This bibliography lists the titles of publications containing additional information about PSF, AFP, the z/OS operating system, and related products.

The titles and order numbers might change from time to time. To verify the current title or order number, consult your IBM marketing representative.

You can obtain many of the publications listed in this bibliography from the AFP Consortium Publications (afpcinc.org/publications), the IBM Documentation (www.ibm.com/docs/en), and the IBM Publications Center (www.ibm.com/e-business/linkweb/publications/servlet/pbi.wss).

Advanced Function Presentation (AFP)

The AFP publications in Table 49 on page 535 are available from AFP Consortium Publications (afpcinc.org/publications).

Table 49. AFP publications available from the AFP Consortium	
Publication	**Order Number**
Advanced Function Presentation: Programming Guide and Line Data Reference	S544-3884
AFP Consortium: AFP Color Management Architecture (ACMA)	AFPCC
Bar Code Object Content Architecture Reference	AFPC-0005
Color Management Object Content Architecture Reference	AFPC-0006
Font Object Content Architecture Reference	AFPC-0007
Graphics Object Content Architecture for AFP Reference	AFPC-0008
Guide to Advanced Function Presentation	G544-3876
IBM AFP Fonts: Font Summary for AFP Font Collection	S544-5633
IBM AFP Fonts: Type Transformer User's Guide	G544-3796
IBM Infoprint Fonts: Font Summary	G544-5846
IBM Infoprint Fonts: Introduction to Type Transformer and Utilities for Windows	G544-5853
Image Object Content Architecture Reference	AFPC-0003
Intelligent Printer Data Stream Reference	AFPC-0001
Mixed Object Document Content Architecture Reference	AFPC-0004
Overlay Generation Language/370 User's Guide and Reference	S544-3702
Presentation Text Object Content Architecture Reference	AFPC-0009
Using OpenType Fonts in an AFP System	G544-5876

The AFP publications in Table 50 on page 535 are available from the IBM Documentation (www.ibm.com/docs/en) and the IBM Publications Center (www.ibm.com/e-business/linkweb/publications/servlet/pbi.wss). For best results, search the IBM Publications Center by order number.

Table 50. AFP publications available from the IBM Documentation and the IBM Publications Center	
Publication	**Order Number**
z/OS Font Collection	GA32-1048

The AFP publications in Table 51 on page 536 are available from the IBM Publications Center (www.ibm.com/e-business/linkweb/publications/servlet/pbi.wss). For best results, search the IBM Publications Center by order number.

Table 51. AFP publications available from the IBM Publications Center

Publication	Order Number
AFP Toolbox User's Guide	S544-5292
Guide to Advanced Function Presentation	G544-3876
IBM AFP Fonts: Font Summary for AFP Font Collection	S544-5633
IBM AFP Fonts: Type Transformer User's Guide	G544-3796
IBM Infoprint Fonts: Font Summary	G544-5846
IBM Infoprint Fonts: Introduction to Type Transformer and Utilities for Windows	G544-5853

i-data

Publication	Order Number
i-data 7913 IPDS Printer LAN Attachment for Ethernet Installation Guide	none
i-data 7913 IPDS Printer LAN Attachment for Token Ring Installation Guide	none

Index

M

macro-call routine 284
macroinstructions, NCP 30
macros (ENQ or DEQ) 295, 305, 310
macros for programming interfaces 465
magnetic ink character recognition printing 181
major nodes, defining
    local 25
    NCP 30
    switched 27
Management mode parameter 62
managing resources 173
Map to outline fonts parameter 62
MAP2OLN parameter 102
mapping, fonts 235
mark form carrier-strip (MFCS) 155
MARK parameter, JES2 125
mark-interrupt-message-page parameter 62
marking
    carrier-strips 155
    copies 156
    font resources 223
    resources for DPF 246
maximum transmission unit in TCP/IP profile 44
MAXPROC parameters, updating 7
MEMLIMIT parameter 195
message call, Exit 16 320
message data set redirection 328
message exit (Exit 6) 292
MESSAGE parameter 102
message text for APS954I 299
message-count-before-dump parameter 62
messages, PSF 327
MFCS command 155
MGMTMODE parameter 102
MICR postprocessor 181
microfilm
    supporting AFP output on 341
migrating
    channel-attached printers to TCP/IP 42
    PSF parameters to the Printer Inventory 353
MO:DCA interchange set auxiliary pages, setting up 116
MODE parameter
    JES2 125
    JES3 130
MODEENT statement 22
MODIFY command 203
MTU size in TCP/IP profile 44
multiple resolution modes 214
multipoint nonswitched line, NCP macroinstructions 34

N

N_UP printing
    basic 180
    enhanced 181
navigation
    keyboard 491
NCP
    macroinstructions 30
    major node, defining 30
    MAXOUT value 192
    performance 192

NCP *(continued)*
    SNA set up for 5
nested resource 297, 312
network printers, PSF parameters for 118
networks, communication 17
No response action parameter 62
No response action: Notify parameter 62
no-response-notify parameter 62
nodes, major
    local 25
    NCP 30
    switched 27
nonswitched line
    communication-attached devices, requirements 19
    multipoint, NCP macroinstructions 34
    NCP macroinstructions 32
NORESP parameter 102
NOTIFY JCL keyword 219
NPRO
    JES2 parameter 125
    JES3 parameter 130
    PRINTDEV statement 102
NST trace dsname parameter 58
number of FSSs and FSAs supported 123
number, concatenation 311

O

OBJCONDD parameter 102
OBJCPATH parameter 102
object container resources
    data objects in UNIX files 255
Object containers parameter 62
objects, resource 173
offset stacking
    overview 156
    separating from copy marking 158
Offset stacking parameter 62
offset-interrupt-message-page parameter 62
OpenType fonts
    capture 232
    obtaining 243
    using 243
operating TCP/IP-attached printers 46
optional message text for APS954I 299
optional programs for PSF 2
outline font
    capture
        double-byte 232
        single-byte 231
    mapping
        to raster 238
outline fonts
    capture
        downloaded 235
    concatenating 234
    creating 234
    description 233
    mapping
        from GRID 240
        from raster 237
    mapping tables, using 237
    performance, FOCA 189
    recovering from font not found 236

resolution modes, multiple 214
Resolution parameter 62
resolutions, printing at multiple 213
resource
 access call (RAC) 306
 buffers 194
 color management 255
 control with RDC 312
 definition of 173
 deletion call (RDC) 311
 deletion, expediting 189
 Exit 7 294
 font pruning 188, 429
 full control of deletion 312
 hard page segment 308
 hierarchy of 174
 integrity considerations 173
 libraries for multiple resolution 215
 libraries, PSF 49
 library for DPF 245
 library for PSF 245
 list (APSURLST) 442
 list control block (RLST) 308, 309
 load call (RLC) 310
 management 173
 marking 223, 245
 microfilm setup resource 347
 nested 297, 312
 Open Type fonts 243
 partial control of deletion 311
 performance considerations 188
 PLR 311
 resident fonts 188
 searching for in print job 179
 selection tables and hierarchy 174
 soft page segment 308
 TrueType fonts 243
Resource Access Control Facility
 access to user libraries 51
 for sensitive resources 219
 security for PSF 5
resource access table (RAT 243
resource deletion call (RDC) 311
Resource deletion ratio parameter 62
resource deletion, expediting 189
resource list control block (RLST) 308, 309
resource load call (RLC) 310
resource table, APSRFTBL font 411
resource-access call (RAC) 306
resource-deletion-ratio parameter 112
resource-management exit (Exit 7) 294
resources used to print documents, determining 181
Response timeout parameter 62
RESPTIME parameter 102
restartable abends
 controlling in Exit 7 298
 understanding 330
Restrict printable area parameter 62
restrictions
 exception to resource list control block 309
 for Exit 7 442
 for printers not supporting selective deletion 312
 for system macros ENQ or DEQ 311
 with RAC 309

retained resource counts
 Fonts parameter 62
 Form definitions parameter 62
 Object containers parameter 62
 Page definitions parameter 62
 Page segments parameter 62
retained-fonts parameter 62
retained-form-definitions parameter 62
retained-object-containers parameter 62
retained-page-definitions parameter 62
retained-page-segments parameter 62
retrieval lists (CRLs), concatenation 301, 305, 311
return codes for font conversion 458
RLC 310
RLST 308, 309
RLSTFEXT parameter area 446
routines, service 281
RRLV 311, 455

S

sample jobs, APSRMARK 399
samples included with PSF 3
saving printer information 201
SDEF240 parameter 102
SDEF300 parameter 102
SDEFDD parameter 102
SDSF notification 326
search for specified resources 179
seconds between checkpoint 125, 130
section ID 311
security labeling 219
selective deletion restriction 312
Send default character parameter 62
Send msgs to SYSOUT parameter 62
send-messages-to-sysout parameter 62
SENDGARBAGE parameter for keep-alive transmissions 44
sending to IBM
 reader comments xxix
SEP parameter, JES2 125
separating copy marking and offset stacking 158
separating jobs or copies of jobs
 carrier-strip marking 155
 copy marking 156
 edge marking 156
 header pages 158
separating users' jobs
 blank pages after separator pages 153
 forms marking 153
 separator pages 148
 using MFCS 155
separator page exits (Exits 1, 2, 3) 276
separator pages
 DPF considerations 153
 for job header and trailer 148
 printing multiple copies of 153
 resources 152
SEPDS parameter, JES2 125
SERVICE instruction 32, 34
service routines 281
set-3800-dataset-header-origin parameter 62
set-3800-dataset-origin parameter 62
set-3800-job-header-origin parameter 62
set-3800-job-trailer-origin parameter 62

IBM.

Product Number: 5655-M32