

Print Services Facility for OS/390



# Customization

*Version 3, Release 1.0*



Print Services Facility for OS/390



# Customization

*Version 3, Release 1.0*

**Note**

Before using this information and the product it supports, be sure to read the general information in "Notices" on page xix.

**First Edition (January 1999)**

This edition applies to Version 3 Release 1.0 of the IBM Print Services Facility for OS/390 licensed program, Program Number 5655-B17, and to all subsequent releases and modifications until otherwise indicated in new editions or Technical Newsletters. Be sure to use the correct edition for the level of the product.

See the Summary of Changes for the changes made to this publication. Technical changes or additions to the text and illustrations are indicated by a vertical line to the left of the change.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

The IBM Printing Systems Company welcomes your comments. A form for reader's comments is provided at the back of this publication. If the form has been removed, you may send your comments to the following address:

INFORMATION DEVELOPMENT  
THE IBM PRINTING SYSTEMS COMPANY  
DEPARTMENT H7FE BUILDING 003G  
PO BOX 1900  
BOULDER CO 80301-9191

If you prefer to send comments electronically, use one of the following methods:

- Internet: [printpub@us.ibm.com](mailto:printpub@us.ibm.com)
- Fax: 1-800-524-1519 or 1-303-924-6873

**Internet**

Visit our home pages at <http://www.printers.ibm.com>.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 1985, 1996, 1999. All rights reserved.**

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

---

# Contents

|  |       |
|--|-------|
| <b>Notices</b> . . . . .   | xix   |
| Programming Interfaces . . . . .   | xix   |
| Trademarks . . . . .   | xix   |
| EuroReady . . . . .  | xix   |
| Year 2000 Ready . . . . .  | xx    |
| <br>   |       |
| <b>About This Publication</b> . . . . .  | xxi   |
| Who Should Read This Publication? . . . . .  | xxi   |
| Why Should I Read This Publication? . . . . .  | xxi   |
| How This Publication Is Organized . . . . .  | xxii  |
| Understanding the Syntax Notation Used in This Book . . . . .                              | xxiii |
| Related Information . . . . .  | xxiii |
| <br>   |       |
| <b>Summary of Changes</b> . . . . .  | xxv   |
| <br>   |       |
| <b>Chapter 1. Understanding PSF for OS/390</b> . . . . .                                   | 1     |
| What Is Shipped with PSF? . . . . .  | 1     |
| What Can I Do with PSF? . . . . .  | 1     |
| Summary of PSF Support . . . . .   | 2     |
| Printers Supported by PSF . . . . .  | 3     |
| Printers Attached through an Intermediate Device . . . . .                                 | 4     |
| <br>   |       |
| <b>Chapter 2. Defining Channel-Attached Printers to OS/390</b> . . . . .                   | 5     |
| Prerequisites for Defining Channel-Attached Printers to OS/390 . . . . .                   | 5     |
| Using HCD to Define Channel-Attached Printers . . . . .                                    | 5     |
| Using Switches with Host-Connected, Channel-Attached Printers . . . . .                    | 6     |
| Using an Intermediate Device with Channel-Attached Printers . . . . .                      | 6     |
| Using ESCON Channel-Attached Printers . . . . .  | 6     |
| Defining Channel-Attached Printers with HCD . . . . .                                      | 7     |
| <br>   |       |
| <b>Chapter 3. Establishing Network Nodes for SNA-Attached Printers</b> . . . . .           | 9     |
| Overview of a Network . . . . .  | 9     |
| Local-Attached Devices . . . . .   | 9     |
| Communication-Attached Devices . . . . .   | 10    |
| VTAM Definitions for PSF Network Nodes . . . . .   | 11    |
| SNA Devices, Attachment Mode, and Requirements . . . . .                                   | 12    |
| VTAM Syntax Rules . . . . .  | 14    |
| Defining a PSF Application Node . . . . .  | 14    |
| Defining the Logon-Mode Table Entry . . . . .  | 16    |
| Local-Attached Entries (Excluding PSF/2 and PSF for AIX) . . . . .                         | 16    |
| Communication-Attached Entries (Excluding PSF/2, PSF for AIX, and<br>Token Ring) . . . . . | 16    |
| Defining Local Major Nodes . . . . .   | 19    |
| Defining Switched Major Nodes . . . . .  | 21    |
| Defining Network Control Program (NCP) Major Nodes . . . . .                               | 25    |
| NCP Generation . . . . .   | 26    |
| Switched Line . . . . .  | 26    |
| Nonswitched Line . . . . .   | 28    |
| Multipoint Nonswitched Line . . . . .  | 31    |
| Token-Ring Subsystem Connection . . . . .  | 34    |

|   |           |
|---|-----------|
| General Network Considerations . . . . .                                    | 35        |
| Logon-Mode Table . . . . .  | 35        |
| VTAM Definition Names and PSF . . . . .                                     | 35        |
| LOGAPPL and the LU Statement . . . . .                                      | 35        |
| PSF PRINTDEV Statement Considerations for Switched Lines . . . . .          | 35        |
| 3130, 3160, and 3935 Configuration . . . . .                                | 36        |
| DPF Considerations . . . . .  | 36        |
| PSF Direct Considerations . . . . .   | 36        |
| <br>  |           |
| <b>Chapter 4. Working with Libraries . . . . .</b>                          | <b>37</b> |
| Library Format . . . . .  | 37        |
| Concatenating Defined Libraries to System Libraries . . . . .               | 37        |
| User Libraries in Deferred-Printing Mode . . . . .                          | 38        |
| <br>  |           |
| <b>Chapter 5. Changing the System Environment . . . . .</b>                 | <b>39</b> |
| How to Change the System Environment for PSF 3.1.0 for OS/390 . . . . .     | 39        |
| Default PPT Entry . . . . .   | 40        |
| <br>  |           |
| <b>Chapter 6. PRINTDEV Parameters . . . . .</b>                             | <b>41</b> |
| Selecting PSF Parameters for a Dedicated Network-Attached Printer . . . . . | 51        |
| Selecting PSF Parameters for Shared Printers . . . . .                      | 51        |
| JES Spool-Driven . . . . .  | 51        |
| VTAM RELREQ . . . . .   | 52        |
| Port Switching . . . . .  | 52        |
| Sharing ESCON Channel-Attached Printers . . . . .                           | 53        |
| Recommended Values for PSF Timers . . . . .                                 | 54        |
| <br>  |           |
| <b>Chapter 7. Using Deferred-Printing Mode . . . . .</b>                    | <b>55</b> |
| Defining PSF FSSs and FSAs for Printers . . . . .                           | 55        |
| Sample Initialization Statements for JES2 . . . . .                         | 56        |
| Parameters for the JES2 Initialization Statement . . . . .                  | 56        |
| FSS Statement . . . . .   | 57        |
| PRTnnnn Statement . . . . .   | 57        |
| Sample Initialization Statements for JES3 . . . . .                         | 61        |
| Parameters for the JES3 Initialization Statement . . . . .                  | 62        |
| FSSDEF Statement . . . . .  | 62        |
| DEVICE Statement . . . . .  | 63        |
| PSF Startup Procedures for AFP Printers . . . . .                           | 67        |
| Sample PSF Startup Procedures . . . . .                                     | 67        |
| JCL for PSF Startup Procedures . . . . .                                    | 73        |
| Identifying Output from a Printer . . . . .                                 | 75        |
| Separating Jobs . . . . .   | 75        |
| Job Separator Pages . . . . .   | 75        |
| Additional Distribution Information . . . . .                               | 77        |
| Separator Page Resources . . . . .  | 79        |
| Printing Copies of a Separator Page . . . . .                               | 80        |
| Distributed Print Function (DPF) Considerations . . . . .                   | 80        |
| PSF and DPF on a JES3 System . . . . .                                      | 80        |
| Forms Marking . . . . .   | 81        |
| Forms Marking of Multiple-Page Job Trailers . . . . .                       | 83        |
| Forms Marking of the Interrupt Message Page . . . . .                       | 83        |
| Carrier-Strip Marking . . . . .   | 83        |
| Copy Marking and Offset Stacking . . . . .                                  | 84        |
| Separating Data Sets or Copies of Data Sets . . . . .                       | 85        |

|   |            |
|---|------------|
| Disabled Mechanisms . . . . .   | 86         |
| System Management Facility Type 6 Records . . . . .   | 86         |
| Structure of SMF Type 6 Records . . . . .   | 88         |
| <b>Chapter 8. Using Direct-Printing Mode . . . . .</b>  | <b>95</b>  |
| How to Use Direct-Printing Mode . . . . .   | 96         |
| Defining the DPSS . . . . .   | 97         |
| Application Program JCL for Direct-Printing Mode . . . . .                                    | 97         |
| Requesting Multiple Copies . . . . .  | 100        |
| 3800 Considerations . . . . .   | 100        |
| <b>Chapter 9. Using Cut-Sheet Emulation Mode . . . . .</b>                                    | <b>101</b> |
| How Do I Get Cut-Sheet Emulation Working? . . . . .   | 101        |
| Do I Have Any Control over Using Cut-Sheet Emulation? . . . . .                               | 101        |
| Will Cut-Sheet Emulation Work with 3800 Line-Mode Conversion? . . . . .                       | 102        |
| Things to Consider When Using Cut-Sheet Emulation . . . . .                                   | 102        |
| <b>Chapter 10. Enabling PSF 3800 Line-Mode Conversion . . . . .</b>                           | <b>105</b> |
| What is IBM 3800 Line-Mode Conversion? . . . . .  | 105        |
| How Does 3800 Line-Mode Conversion Work? . . . . .  | 105        |
| How Can PSF 3800 Line-Mode Conversion Be Enabled? . . . . .                                   | 106        |
| Considerations and Limitations of the 3800 Line-Mode Conversion Function . . . . .            | 108        |
| Implementation Checklist for PSF 3800 Line-Mode Conversion . . . . .                          | 109        |
| <b>Chapter 11. Managing Resources and Selecting Special Processing Parameters . . . . .</b>   | <b>111</b> |
| How PSF Selects Special Processing Parameters . . . . .                                       | 111        |
| Selecting Region Sizes . . . . .  | 112        |
| Installation Options . . . . .  | 112        |
| For Each FSA . . . . .  | 113        |
| Redirecting the Message Data Set . . . . .  | 113        |
| Printing PSF Error Messages . . . . .   | 114        |
| Specifying Checkpoint Intervals . . . . .   | 115        |
| How PSF Selects Resources . . . . .   | 115        |
| Selecting a Forms Flash for a 3800 Printer . . . . .  | 116        |
| Selecting a Copy Group . . . . .  | 117        |
| Form Definitions in Deferred-Printing Mode . . . . .  | 117        |
| Form Definitions in Direct-Printing Mode . . . . .  | 118        |
| Page Definitions in Deferred-Printing Mode . . . . .  | 119        |
| Page Definitions in Direct-Printing Mode . . . . .  | 119        |
| Fonts in Deferred-Printing Mode . . . . .   | 120        |
| Fonts in Direct-Printing Mode . . . . .   | 121        |
| Searching for a Resource Specified for a Print Job . . . . .                                  | 121        |
| Using Form Definitions for Basic N_UP Printing . . . . .                                      | 121        |
| Using Form Definitions for Enhanced N_UP Printing . . . . .                                   | 122        |
| Using PSF, the Interface Attachment Feature, and the MICR Postprocessor . . . . .             | 123        |
| Maintaining Resource Integrity . . . . .  | 123        |
| <b>Chapter 12. Handling Job Streams That Are Formatted at Different Resolutions . . . . .</b> | <b>125</b> |
| Multiple Resolution Printers . . . . .  | 126        |
| Printers with Switchable Resolution Mode . . . . .  | 126        |
| Printers with Automatic or Auto-Resolution Mode . . . . .                                     | 127        |
| Resource Libraries Segmented for Multiple Resolution . . . . .                                | 127        |

|   |            |
|---|------------|
| Format Resolution Specified from the Application . . . . .                    | 129        |
| Specifying Format Resolution in JCL . . . . .                                 | 129        |
| Specifying Format Resolution in the Print Data Stream . . . . .               | 129        |
| Specifying Font Fidelity in the Print Data Stream . . . . .                   | 129        |
| Multiple Resolution and Installation Exits . . . . .                          | 130        |
| Format Resolution Order . . . . .   | 130        |
| <br>  |            |
| <b>Chapter 13. Understanding PSF Data Protection and Security . . . . .</b>   | <b>131</b> |
| Security Labeling . . . . .   | 131        |
| Controlling Print Complete Notification . . . . .                             | 132        |
| <br>  |            |
| <b>Chapter 14. Supporting Resident Fonts . . . . .</b>                        | <b>133</b> |
| PSF Resident Font Table (APSRFTBL) . . . . .                                  | 134        |
| Changing APSRFTBL . . . . .   | 134        |
| PSF Resident-Font Utility—APSRMARK . . . . .                                  | 135        |
| APSRMARK Considerations . . . . .   | 136        |
| Using APSRMARK . . . . .  | 137        |
| Syntax of APSRMARK Control Statements . . . . .                               | 137        |
| APSRMARK Control Statement Parameters for Resident Fonts . . . . .            | 138        |
| Using Control Statements to Mark a Font . . . . .                             | 139        |
| APSRMARK Control Statement Parameters (General) . . . . .                     | 140        |
| Sample Control Statements for APSRMARK . . . . .                              | 141        |
| Marking a Member with Resident Font Information . . . . .                     | 141        |
| Generating a Short Report of Resident Font Marking . . . . .                  | 142        |
| Generating a Long Report of Resident Font Marking . . . . .                   | 142        |
| Diagnosing Problems with Printer-Resident Fonts . . . . .                     | 143        |
| <br>  |            |
| <b>Chapter 15. Marking Fonts for Font Capture . . . . .</b>                   | <b>145</b> |
| Single-Byte Raster Font Capture . . . . .                                     | 145        |
| Single-Byte Outline Font Capture . . . . .                                    | 145        |
| Double-Byte Raster Font Capture . . . . .                                     | 146        |
| Double-Byte Outline Font Capture . . . . .                                    | 146        |
| <br>  |            |
| <b>Chapter 16. Using Outline Fonts . . . . .</b>                              | <b>147</b> |
| IBM's Font Strategy . . . . .   | 147        |
| Understanding Outline Fonts . . . . .   | 147        |
| Concatenating Outline Fonts with Other Library Resources . . . . .            | 148        |
| Creating New Fonts . . . . .  | 148        |
| Capturing New Outline Fonts . . . . .   | 149        |
| Font Mapping . . . . .  | 149        |
| Timing of Mapping Operations . . . . .  | 150        |
| Mapping Algorithms . . . . .  | 150        |
| Using the Font Mapping Tables . . . . .                                       | 150        |
| Adding a New Mapping of a Raster Font to an Outline Font . . . . .            | 151        |
| Adding a New Mapping of an Outline Font to a Raster Font . . . . .            | 152        |
| Disabling the Mapping of a Raster Font to an Outline Font . . . . .           | 154        |
| Adding a New Mapping of a GRID to a Raster Font . . . . .                     | 154        |
| Adding a New Mapping of a GRID to an Outline Font . . . . .                   | 155        |
| <br>  |            |
| <b>Chapter 17. Marking Resources for Distributed Print Function . . . . .</b> | <b>157</b> |
| Using APSRMARK to Mark Resources . . . . .                                    | 157        |
| Control-Statement Syntax for APSRMARK . . . . .                               | 158        |
| Control-Statement Parameters for APSRMARK . . . . .                           | 158        |
| Examples of Control Statements for APSRMARK . . . . .                         | 160        |



|  |            |
|--|------------|
| Creating a PSF Resource Library with All Members Marked PUBLIC   | 160        |
| Creating a PSF Resource Library with Some Members Marked PRIVATE | 161        |
| Modifying an Existing PSF Resource Library                       | 162        |
| Printing a Report of Resource Markings in a Partitioned Data Set | 163        |
| Using Page Segments  | 164        |
| Using DPF Fonts and Printer-Resident Fonts with APSRMARK         | 165        |
| Changing How Fonts Are Marked                                    | 166        |
| Fonts Already Marked for Use by DPF Only                         | 166        |
| Fonts Already Marked for DPF and Printer-Resident Use            | 166        |
| Fonts Already Marked PRIVATE                                     | 166        |
| <b>Chapter 18. Using Installation Exits</b>                      | <b>167</b> |
| Do's and Don'ts  | 167        |
| Changing Exit Routines   | 169        |
| Common Control Blocks  | 170        |
| APSGEXTP Area  | 171        |
| Exit-Communication Areas: APSUECA and APSUECE                    | 173        |
| Sequence of Exit Invocation                                      | 178        |
| Control Flow with APSUX07  | 178        |
| Separator-Page Exits: APSUX01, APSUX02, and APSUX03              | 179        |
| Input for Separator-Page Exits                                   | 179        |
| Output for Separator-Page Exits                                  | 179        |
| Processing Multiple-Page Job Trailers with Forms Marking         | 179        |
| Programming with Separator-Page Exits                            | 180        |
| JES Job-Separator-Page Areas—IAZJSPA and IAZJSEP                 | 181        |
| Exit Table, APSUCOM  | 182        |
| Block-Letter Tables, APSUBTBL                                    | 184        |
| Service Routines   | 185        |
| Block-Letter Routine, APSUBLK                                    | 185        |
| Input for APSUBLK  | 185        |
| Output for APSUBLK   | 185        |
| Programming Considerations for APSUBLK                           | 186        |
| Exit-Put Routine, APSUPUT  | 186        |
| Input for APSUPUT  | 187        |
| Output for APSUPUT   | 187        |
| Logical-Record Processing Exit: APSUX04                          | 187        |
| Input for APSUX04  | 187        |
| Output for APSUX04   | 188        |
| Programming Considerations for APSUX04                           | 189        |
| SMF Type 6 Record Exit: APSUX05                                  | 193        |
| Input for APSUX05  | 194        |
| Output for APSUX05   | 194        |
| Programming Considerations for APSUX05                           | 194        |
| PSF Message Exit: APSUX06  | 195        |
| Input for APSUX06  | 196        |
| Output for APSUX06   | 196        |
| Programming Considerations for APSUX06                           | 196        |
| Resource-Management Exit: APSUX07                                | 198        |
| Invoking APSUX07   | 198        |
| Sequence of Events   | 199        |
| APSUX07 Subroutine Calls   | 199        |
| Programming Considerations for APSUX07                           | 200        |
| Page Segment Considerations                                      | 200        |
| Special Performance Considerations                               | 200        |

|   |     |
|---|-----|
| IOCA, GOCA, and BCOCA Considerations . . . . .                            | 201 |
| 3800-Compatible Media Origin Considerations . . . . .                     | 201 |
| Considerations on Suppressing an Interrupt Message Page . . . . .         | 202 |
| Considerations on Redirecting a Message Data Set . . . . .                | 202 |
| Overlay Considerations . . . . .  | 203 |
| Considerations on Converting a Multiple-Celled IM1 Image . . . . .        | 203 |
| Considerations on Eject-to-Front-Facing Function . . . . .                | 203 |
| Considerations on Restartable Abends . . . . .                            | 204 |
| Considerations on Requesting an Acknowledgement . . . . .                 | 204 |
| Considerations on Cut-Sheet Emulation Mode . . . . .                      | 205 |
| Considerations on Forms Marking of an Interrupt Message Page . . . . .    | 205 |
| Considerations Related to Page Definitions and Form Definitions . . . . . | 205 |
| Processing Considerations . . . . .                                       | 206 |
| Selection Hierarchy . . . . .   | 206 |
| Data-Set Termination Considerations . . . . .                             | 206 |
| 3800 Considerations . . . . .   | 207 |
| 3800 Forms Marking . . . . .  | 207 |
| Considerations for Setting a Format Resolution Value . . . . .            | 207 |
| Considerations Related to User Libraries . . . . .                        | 208 |
| Considerations Related to Inline Resources . . . . .                      | 208 |
| Considerations Related to Outline Fonts . . . . .                         | 208 |
| Considerations Related to Mapping to Outline Fonts . . . . .              | 209 |
| Considerations Related to Recover-from-Font-Not-Found Flag . . . . .      | 209 |
| Considerations Related to Inhibiting Recovery . . . . .                   | 210 |
| Considerations Related to Ending the SNA LU1 Conversation . . . . .       | 210 |
| Initialization Call (INIT) . . . . .                                      | 211 |
| Input and Output Indicators for INIT . . . . .                            | 212 |
| Programming Considerations for INIT . . . . .                             | 212 |
| Begin-Data-Set Call (BDSC) . . . . .                                      | 212 |
| Input and Output Indicators for BDSC . . . . .                            | 213 |
| Programming Considerations for BDSC . . . . .                             | 213 |
| Resource-Access Call (RAC) . . . . .                                      | 213 |
| Input and Output Indicators for RAC . . . . .                             | 214 |
| Programming Considerations for RAC . . . . .                              | 214 |
| RAC and Overlays . . . . .  | 215 |
| RAC and Page Segments . . . . .   | 215 |
| RAC and Fonts . . . . .   | 216 |
| Restrictions . . . . .  | 216 |
| RAC, Page Definitions, and Form Definitions . . . . .                     | 217 |
| Using RAC in Processing Page Definitions and Form Definitions . . . . .   | 217 |
| Resource-Load Call (RLC) . . . . .  | 218 |
| Input and Output Indicators for RLC . . . . .                             | 218 |
| Programming Considerations for RLC . . . . .                              | 218 |
| Resource-Deletion Call (RDC) . . . . .                                    | 219 |
| Input and Output Indicators for RDC . . . . .                             | 219 |
| Partial Control of Resource Deletion . . . . .                            | 219 |
| Full Control of Resource Deletion . . . . .                               | 220 |
| Using RDC to Control Resources . . . . .                                  | 220 |
| Termination Call (TERM) . . . . .   | 221 |
| Input and Output Indicators for TERM . . . . .                            | 221 |
| Direct-Printing Line Mode Exit: APSUX14 . . . . .                         | 221 |
| Input for APSUX14 . . . . .   | 221 |
| Output for APSUX14 . . . . .  | 222 |
| Programming Considerations for APSUX14 . . . . .                          | 222 |

|  |            |
|--|------------|
| Download for OS/390 Print Parameter Exit: APSUX15    | 222        |
| Direct-Print Exit: APSUX16                           | 223        |
| Begin-Data-Set Call (BDSC)                           | 223        |
| Input for APSUX16 Begin-Data-Set Call                | 224        |
| Output for APSUX16 Begin-Data-Set Call               | 224        |
| Message Call for APSUX16                             | 224        |
| Input for APSUX16 Message Call                       | 224        |
| Output for APSUX16 Message Call                      | 224        |
| End-Data-Set Call (EDSC)                             | 225        |
| Input for APSUX16 End-Data-Set Call                  | 225        |
| Output for APSUX16 End-Data-Set Call                 | 225        |
| Programming Considerations for APSUX16               | 225        |
| <b>Chapter 19. Understanding AFP Performance</b>     | <b>229</b> |
| The Printer  | 229        |
| Mechanical Speed                                     | 229        |
| Internal Processing Speed                            | 229        |
| Paper Size   | 230        |
| Printer Features                                     | 230        |
| Raster Storage                                       | 230        |
| Enhancements and RPQs                                | 230        |
| Duplex Printing                                      | 230        |
| The Print Job  | 230        |
| Data Density and Complexity                          | 230        |
| Characteristics of Print Data Sets                   | 231        |
| Number and Type of Resources                         | 232        |
| Inline Resources                                     | 232        |
| Resident Fonts                                       | 232        |
| Font Pruning of Raster Fonts                         | 232        |
| Captured Fonts                                       | 233        |
| Size of Print Jobs                                   | 233        |
| Using Outline Fonts                                  | 233        |
| Transmission and Compression of Data                 | 233        |
| The Attachment Hardware                              | 234        |
| Host-Connected, Channel-Attached Printer             | 234        |
| Local-Attached Printer                               | 234        |
| Communication-Attached Printer                       | 235        |
| PSF Direct or Distributed Print Function (DPF)       | 235        |
| PSF Direct   | 235        |
| DPF  | 235        |
| Size of the VTAM Request Unit (RU)                   | 235        |
| VTAM PSNDPAC and DELAY Parameters                    | 236        |
| NCP MAXOUT Value                                     | 236        |
| TCP/IP-Attached Printer                              | 236        |
| The Host System                                      | 237        |
| Processing Capability                                | 237        |
| Amount of Real Storage Available                     | 237        |
| Capacity of the I/O Subsystem                        | 237        |
| System Tuning  | 237        |
| Blank Compression for Line Data (COMPRESS Parameter) | 238        |
| ACK Per Sheet  | 238        |
| Printer System Data Rate Requirements                | 239        |
| Printer Data Checks                                  | 239        |
| Buffers  | 239        |

|   |            |
|---|------------|
| Block Size . . . . .  | 241        |
| Data Separation . . . . .   | 241        |
| Functional Subsystem Interface (FSI) . . . . .                                | 241        |
| Checkpointing . . . . .   | 241        |
| Tracing . . . . .   | 242        |
| <b>Chapter 20. Connecting a TCP/IP-Attached IPDS Printer . . . . .</b>        | <b>243</b> |
| Requirements for Using a TCP/IP-Attached IPDS Printer . . . . .               | 243        |
| Overview . . . . .  | 243        |
| Connecting TCP/IP-Attached Printers . . . . .                                 | 244        |
| Defining the Communications Control Unit to MVS . . . . .                     | 244        |
| Modifying the TCP/IP Profile in MVS . . . . .                                 | 244        |
| Installing the i-data 7913 IPDS Printer LAN Attachment . . . . .              | 246        |
| Pinging the Printer . . . . .   | 246        |
| Ping Is Not Successful . . . . .  | 247        |
| Ping Is Successful . . . . .  | 247        |
| Defining the Printer to JES . . . . .   | 248        |
| Defining the Printer to PSF . . . . .   | 248        |
| Operating a TCP/IP-Attached Printer . . . . .                                 | 251        |
| Overview of the TCP/IP and PSF Interface . . . . .                            | 251        |
| Starting a TCP/IP-Attached Printer . . . . .                                  | 251        |
| Stopping a TCP/IP-Attached Printer . . . . .                                  | 251        |
| <b>Chapter 21. Responding to PSF Errors . . . . .</b>                         | <b>253</b> |
| Accumulating PSF Error Messages . . . . .                                     | 253        |
| Repeating Error Messages . . . . .  | 253        |
| Recovering from PSF Errors . . . . .  | 254        |
| Recovering from Printer-Hardware Problems . . . . .                           | 254        |
| Terminating a Printer FSA . . . . .   | 254        |
| Understanding Restartable Abends . . . . .                                    | 255        |
| Requesting Traces and Dumps . . . . .   | 255        |
| <b>Chapter 22. Supporting AFP Output on Microfilm . . . . .</b>               | <b>257</b> |
| PSF Hardware Attachment Requirements for Microfilm Devices . . . . .          | 257        |
| PSF System Programmer Tasks for Microfilm Devices . . . . .                   | 257        |
| Defining the Microfilm Device as Channel-Attached or ESCON-Attached . . . . . | 259        |
| Defining the Microfilm Device to OS/390 or MVS . . . . .                      | 259        |
| Defining the Microfilm Device to JES2 and JES3 . . . . .                      | 259        |
| Defining the Microfilm Device to PSF . . . . .                                | 260        |
| PRINTDEV Parameters for Microfilm Devices . . . . .                           | 261        |
| Sample JCL Startup Procedure for Microfilm Devices . . . . .                  | 261        |
| Redirecting Message Data Sets for Microfilm Jobs . . . . .                    | 264        |
| Working with Resource Libraries Required by Microfilm Devices . . . . .       | 264        |
| Working with the Microfilm Setup Resource Object Container Library . . . . .  | 264        |
| Working with Font Libraries . . . . .   | 264        |
| Managing Resources and Selecting Special Processing Parameters . . . . .      | 265        |
| Generating Microfilm Setup Resource Object Containers . . . . .               | 265        |
| Using the Font Library Update (FLU) Utility . . . . .                         | 265        |
| Considerations When Using the FLU . . . . .                                   | 266        |
| Running the FLU . . . . .   | 266        |
| FLU Return Codes . . . . .  | 267        |
| Printing to Microfilm . . . . .   | 267        |
| Operating a Microfilm Device . . . . .  | 268        |
| Microfilm Data Protection and Security Considerations for PSF . . . . .       | 268        |

|   |            |
|---|------------|
| Using PSF Installation Exits for Microfilm Devices . . . . .                        | 268        |
| Understanding PSF Performance Issues That Pertain to Microfilm Devices . . . . .    | 268        |
| Responding to PSF Errors Reported by Microfilm Devices . . . . .                    | 269        |
| Recovering or Restarting AFP Print Jobs for Microfilm . . . . .                     | 269        |
| <b>Appendix A. APSRMARK Sample Jobs . . . . .</b>                                   | <b>271</b> |
| <b>Appendix B. APSRMARK Long Report Format . . . . .</b>                            | <b>273</b> |
| Data Categories . . . . .   | 275        |
| Resource Type . . . . .   | 276        |
| Internal ID . . . . .   | 277        |
| Time(Created) . . . . .   | 277        |
| Time(Revised) . . . . .   | 278        |
| Time(Marked) . . . . .  | 278        |
| Change Level . . . . .  | 279        |
| Management . . . . .  | 280        |
| Object Origin . . . . .   | 280        |
| Enabled For . . . . .   | 280        |
| <b>Appendix C. Font Resource Tables Supplied by PSF . . . . .</b>                   | <b>283</b> |
| APSRFTBL Font Resource Table . . . . .  | 283        |
| <b>Appendix D. Field Descriptions for APSUX07 . . . . .</b>                         | <b>295</b> |
| XTP7 Parameter Area (DSECT) . . . . .   | 295        |
| XTP7 Fields . . . . .   | 302        |
| APSLCRL Parameter Area (Concatenation Retrieval List) . . . . .                     | 308        |
| APSLCRL Fields . . . . .  | 310        |
| CRL Data Area . . . . .   | 311        |
| APSURLST Parameter Area (Resource List) . . . . .                                   | 311        |
| APSURLST Fields . . . . .   | 313        |
| RLSTFEXT Parameter Area (Font Extension Area) . . . . .                             | 317        |
| RLSTFEXT Fields . . . . .   | 318        |
| Processing Indicator Cross-Reference for APSUX07 . . . . .                          | 319        |
| <b>Appendix E. APTRCONV Resource Conversion Utility for VSE . . . . .</b>           | <b>323</b> |
| Converting a Resource . . . . .   | 323        |
| <b>Appendix F. Reasonable Resource Loading Values (RRLV) . . . . .</b>              | <b>325</b> |
| <b>Appendix G. Font Conversion Program, 240-pel to 300-pel (APSRCF30) . . . . .</b> | <b>327</b> |
| Understanding APSRCF30 . . . . .  | 327        |
| Using APSRCF30 . . . . .  | 327        |
| Error Handling . . . . .  | 328        |
| APSRCF30 Considerations . . . . .   | 329        |
| APSRCF30 Sample Jobs . . . . .  | 329        |
| Converting All 240-Pel Fonts in Source Library (APSRAL30) . . . . .                 | 329        |
| Input to APSRAL30 . . . . .   | 330        |
| Output from APSRAL30 . . . . .  | 330        |
| APSRAL30 Processing . . . . .   | 331        |
| APSRAL30 Considerations . . . . .   | 331        |
| Example for Selected Font Conversion . . . . .                                      | 332        |
| <b>Appendix H. Installation Verification Program (IVP) Example . . . . .</b>        | <b>335</b> |

|   |     |
|---|-----|
| <b>Appendix I. PSF Programming Interfaces</b> . . . . .                             | 337 |
| Intended Programming Interfaces . . . . .   | 337 |
| Interfaces for Tailoring PSF . . . . .  | 337 |
| Installation-Wide Exits . . . . .   | 337 |
| <br>  |     |
| <b>Appendix J. Mapping Tables for Outline Fonts</b> . . . . .                       | 339 |
| <br>  |     |
| <b>Appendix K. Installing the i-data 7913 IPDS Printer LAN Attachment</b> . . . . . | 347 |
| Understanding How the i-data 7913 IPDS Printer LAN Attachment Works . . . . .       | 347 |
| Checking the i-data 7913 IPDS Printer LAN Attachment . . . . .                      | 348 |
| Configuring the bootp Daemon . . . . .  | 349 |
| Understanding How the bootp Daemon Works . . . . .                                  | 349 |
| Completing the bootp Configuration Worksheet . . . . .                              | 350 |
| Editing the bootptab File . . . . .   | 350 |
| Example 1: Basic bootp Configuration . . . . .                                      | 351 |
| Example 2: More Complex bootp Configuration . . . . .                               | 351 |
| Setting Up the bootp Daemon on AIX . . . . .  | 352 |
| Setting Up the bootp Daemon on OS/2 . . . . .                                       | 353 |
| Configuring a Twinaxial Printer . . . . .   | 354 |
| Setting the Controls for the i-data 7913 IPDS Printer LAN Attachment . . . . .      | 354 |
| IBM Token-Ring Network Model . . . . .  | 354 |
| Ethernet LAN Model . . . . .  | 355 |
| Operating the i-data 7913 IPDS Printer LAN Attachment . . . . .                     | 356 |
| Attaching the i-data 7913 IPDS Printer LAN Attachment . . . . .                     | 356 |
| Starting the i-data 7913 IPDS Printer LAN Attachment . . . . .                      | 356 |
| Shutting Down the i-data 7913 IPDS Printer LAN Attachment . . . . .                 | 357 |
| Testing the i-data 7913 IPDS Printer LAN Attachment . . . . .                       | 358 |
| Troubleshooting the i-data 7913 IPDS Printer LAN Attachment . . . . .               | 360 |
| <br>  |     |
| <b>Glossary</b> . . . . .   | 363 |
| Sources . . . . .   | 363 |
| References . . . . .  | 363 |
| <br>  |     |
| <b>Bibliography</b> . . . . .   | 389 |
| Print Services Facility (PSF) for OS/390 . . . . .                                  | 389 |
| Advanced Function Presentation (AFP) . . . . .                                      | 389 |
| Advanced Function Presentation (AFP) Architecture . . . . .                         | 390 |
| Advanced Function Presentation for Microfilm . . . . .                              | 390 |
| Fonts . . . . .   | 390 |
| Text Processing . . . . .   | 391 |
| InfoPrint Manager for AIX . . . . .   | 391 |
| Print Services Facility (PSF) for AIX . . . . .                                     | 391 |
| Print Services Facility for OS/2 (PSF/2) . . . . .                                  | 391 |
| EDMSuite OnDemand . . . . .   | 392 |
| Printers . . . . .  | 392 |
| i-data . . . . .  | 392 |
| OS/390 Version 2 Release 6 . . . . .  | 393 |
| eNetwork Communications Server . . . . .  | 394 |
| MVS/Enterprise Systems Architecture (ESA) Version 5 . . . . .                       | 394 |
| TCP/IP . . . . .  | 395 |
| TCP/IP for MVS . . . . .  | 395 |
| VTAM and NCP . . . . .  | 395 |
| Systems Network Architecture . . . . .  | 396 |
| AS/400 . . . . .  | 396 |

Information/System . . . . . 396

**Index** . . . . . 397





---

## Figures

|     |   |     |
|-----|---|-----|
| 1.  | Example of a Channel-Attached Printer Configuration                             | 6   |
| 2.  | HCD Define Device Parameters/Features Panel                                     | 7   |
| 3.  | Sample PSF Configurations for Local-Attached Devices                            | 10  |
| 4.  | Sample PSF Configurations for Communication-Attached Devices                    | 10  |
| 5.  | Sample PSF Configurations for Communication Attachment to DPF or PSF Direct     | 11  |
| 6.  | PSF Application Node  | 15  |
| 7.  | Sample APPL Statement   | 15  |
| 8.  | Sample Logon-Mode Table Entry for Local-Attached Devices                        | 16  |
| 9.  | Sample Logon-Mode Table Entry for Communication-Attached Devices                | 17  |
| 10. | Sample Logon-Mode Table Entry for Token-Ring Devices                            | 17  |
| 11. | Example Showing Three Local Major Nodes   | 19  |
| 12. | Sample Local Major Node Definitions for Different LU Protocols                  | 20  |
| 13. | Sample Configuration of Switched Major Nodes                                    | 22  |
| 14. | Sample Definitions of Switched Major Nodes                                      | 22  |
| 15. | Sample NCP Major Node   | 25  |
| 16. | Sample NCP Macroinstructions for Switched lines                                 | 27  |
| 17. | Sample NCP Macro instructions for Nonswitched Lines                             | 28  |
| 18. | Sample NCP Macroinstructions for Multipoint Lines                               | 31  |
| 19. | Sample JES2 Initialization Statements for Three Printers                        | 56  |
| 20. | Sample JES3 Initialization Statements for Three Printers                        | 61  |
| 21. | Sample Initialization Statements for JES3                                       | 62  |
| 22. | Sample PSF Startup Procedure for Three Printers, Showing Multiple FSAs          | 69  |
| 23. | Sample PSF Startup Procedures for Three Separate Printers                       | 71  |
| 24. | Example of a Default PSF Job-Header Page (APSUX01)                              | 76  |
| 25. | Example of a Fast-Printing Job-Header Page for PSF (APSUX01P)                   | 77  |
| 26. | Sample Header Page  | 78  |
| 27. | Sample Header Page  | 79  |
| 28. | Example of Forms Marking for Cut-Sheet Paper                                    | 81  |
| 29. | Forms Marking for Continuous Forms Not Printed on a 3800 Printer                | 82  |
| 30. | Example of Forms Marking  | 82  |
| 31. | Example of Carrier-Strip Marking on the 3800 Printer                            | 83  |
| 32. | Example of Copy Marking to Simulate Offset Stacking                             | 84  |
| 33. | Map of SMF Type 6 Records   | 88  |
| 34. | Job Stream for Creating an IEFSSNBB Member of SYS1.PARMLIB                      | 97  |
| 35. | Sample JCL to Enable Direct Printing  | 98  |
| 36. | A Printer That Supports Resident Fonts  | 133 |
| 37. | Sample JCL for Marking Resident Fonts   | 139 |
| 38. | Example of the Use of JCL Statements to Mark a Member of a Data Set             | 141 |
| 39. | Example of the Use of JCL Statements in Generating a Short Report               | 142 |
| 40. | Printed Output of Printer-Resident Short Report                                 | 142 |
| 41. | Example of the Use of JCL Statements in Generating a Long Report                | 143 |
| 42. | Comparison of an Outline Font Character with a Raster Font Character            | 148 |
| 43. | Access to Resources by DPF  | 157 |
| 44. | Sample JCL for Creating a PSF Resource Library and Marking all Members PUBLIC   | 160 |
| 45. | Sample JCL for Creating a PSF Resource Library and Marking Some Members PRIVATE | 161 |
| 46. | Sample JCL for Modifying an Existing PSF Resource Library                       | 162 |
| 47. | Sample JCL for Reporting Resource Markings in a Partitioned Data Set            | 163 |

|     |   |     |
|-----|---|-----|
| 48. | Sample Output from a Resource-Marking Report                                | 164 |
| 49. | Use of STEPLIB Statement for Exit Routine in PSF Startup Procedure          | 168 |
| 50. | Format of the APSGEXTP Area   | 171 |
| 51. | Format of Exit-Communication Area, APSUECA                                  | 173 |
| 52. | Format of Exit-Communication Area Extension, APSUECE                        | 175 |
| 53. | Format of JES Job-Separator-Page Area, IAZJSPA.                             | 182 |
| 54. | Format of JES Job-Separator-Page Area Extension, IAZJSEP                    | 182 |
| 55. | Format of the Output in the Work Buffer, ECAWKBUF                           | 186 |
| 56. | APSUX04/APSUX14 Indicators in XTP4FLAG Field                                | 192 |
| 57. | Format of JES Index Table, IAZIDX   | 193 |
| 58. | Appearance of the TCP/IP Connection between an MVS Host and a Printer       | 243 |
| 59. | Actual Physical TCP/IP Connection between an MVS Host and a Printer         | 244 |
| 60. | Sample Parameters in the TCP/IP 3.2 Profile                                 | 245 |
| 61. | Sample JCL Startup Procedure for a TCP/IP-Attached Printer (APSWPROT)       | 249 |
| 62. | Example of AFP Data Being Transmitted to a Microfilm Device                 | 258 |
| 63. | Sample JCL Startup Procedure for a Microfilm Device (APSWPROM)              | 262 |
| 64. | APSWRFLU Sample Job Provided in SYS1.SAMPLIB                                | 267 |
| 65. | Printed Output of Printer-Resident Long Report                              | 273 |
| 66. | APSRFTBL Font Resource Table  | 284 |
| 67. | XTP7 Parameter Area Format  | 296 |
| 68. | APSLCRL Parameter Area Format   | 309 |
| 69. | Format of the CRL Data Area   | 310 |
| 70. | APSURLST Parameter Area Format  | 312 |
| 71. | Format of the RLSTFEXT Parameter Area                                       | 318 |
| 72. | Sample JCL for Converting the MVS PAGEDEF Resource for Use by VSE           | 323 |
| 73. | Sample JCL to Route Job to Reader   | 323 |
| 74. | IVP Output Printed with 240-Pel Resolution                                  | 335 |
| 75. | Global Resource Identifiers (GRIDs) to Font Names                           | 340 |
| 76. | Font Names to Global Resource Identifiers (GRIDs)                           | 343 |
| 77. | i-data 7913 IPDS Printer LAN Attachment in a TCP/IP LAN Environment         | 348 |
| 78. | Data Flow between the i-data 7913 IPDS Printer LAN Attachment and the LAN   | 349 |
| 79. | IBM Token-Ring Network Model of the i-data 7913 IPDS Printer LAN Attachment | 355 |
| 80. | Ethernet LAN Model of the i-data 7913 IPDS Printer LAN Attachment           | 355 |
| 81. | Status indicators on the i-data 7913 IPDS Printer LAN Attachment            | 357 |
| 82. | Sample Test Page for IBM 7913 Token-Ring Network Model                      | 359 |
| 83. | Sample Test Page for Ethernet Model   | 359 |

---

## Tables

|     |  |     |
|-----|--|-----|
| 1.  | SNA Devices: Requirements  | 12  |
| 2.  | Initialization Parameters for the PRINTDEV Statement: Summary    | 41  |
| 3.  | JCL DD Allocation vs. Dynamic Allocation                         | 99  |
| 4.  | Minimum Storage Required for FSS and FSAs                        | 112 |
| 5.  | Hierarchy of Flash Selection in Deferred-Printing Mode           | 116 |
| 6.  | Hierarchy of Flash Selection in Direct-Printing Mode             | 116 |
| 7.  | Hierarchy of Form Definition Selection in Deferred-Printing Mode | 118 |
| 8.  | Hierarchy of Form Definition Selection in Direct-Printing Mode   | 118 |
| 9.  | Hierarchy of Page Definition Selection in Deferred-Printing Mode | 119 |
| 10. | Hierarchy of Page Definition Selection in Direct-Printing Mode   | 119 |
| 11. | Hierarchy of Font Selection in Deferred-Printing Mode            | 120 |
| 12. | Hierarchy of Font Selection in Direct-Printing Mode              | 121 |
| 13. | Resident Font Diagnosis  | 143 |
| 14. | PSF Exit Routines in SYS1.SAMPLIB                                | 169 |
| 15. | Summary of APSUX07 Subroutine Calls                              | 199 |
| 16. | Approximate Maximum Attachment Data Rates                        | 235 |
| 17. | Approximate Maximum Printer Data Rates                           | 239 |
| 18. | Recommended BUFNO values   | 240 |
| 19. | Sample Jobs in SYS1.SAMPLIB                                      | 271 |
| 20. | Data Areas on a APSRMARK Long Report                             | 275 |
| 21. | Enable For Settings  | 281 |
| 22. | Input and Output Summary for APSUX07 Processing Indicators       | 319 |
| 23. | Reasonable Resource Loading Values (RRLV)                        | 325 |
| 24. | GRID-to-Font-Name Table, APSRGTM                                 | 339 |
| 25. | Font-Name-to-GRID Table, APSRNTGM                                | 339 |
| 26. | Configuration Worksheet for bootp                                | 350 |
| 27. | Problems in Installing the 7913 Attachment                       | 361 |
| 28. | Problems in Operating the 7913 Attachment                        | 362 |



---

## Notices

References in this publication to products or services of IBM do not suggest or imply that IBM will make them available in all countries where IBM does business or that only products or services of IBM may be used. Noninfringing equivalents may be substituted, but the user must verify that such substitutes, unless expressly designated by IBM, work correctly. No license, expressed or implied, to patents or copyrights of IBM is granted by furnishing this publication. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785, USA.

---

## Programming Interfaces

This publication includes documentation of intended Programming Interfaces that let the customer write programs to obtain the services of Print Services Facility (PSF) for OS/390.

---

## Trademarks

The following are trademarks or registered trademarks of the IBM Corporation:

|                                 |                         |
|---------------------------------|-------------------------|
| ACF/VTAM                        | MVS                     |
| Advanced Function Presentation  | MVS/ESA                 |
| AFCCU                           | MVS/SP                  |
| AFP                             | NetSpool                |
| AIX                             | Operating System/2      |
| APL2                            | Operating System/400    |
| Application System/400          | OS/2                    |
| AS/400                          | OS/390                  |
| eNetwork                        | OS/400                  |
| ESCON                           | Personal System/2       |
| Extended Services               | PrintManager            |
| GDDM                            | Print Services Facility |
| IBM                             | Proprinter              |
| InfoPrint                       | RACF                    |
| Intelligent Printer Data Stream | RS/6000                 |
| IPDS                            | S/370                   |
| IP PrintWay                     | S/390                   |
| JES                             | VTAM                    |

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

---

## EuroReady

PSF 3.1.0 for OS/390 is capable of processing data containing the euro sign. Font character sets and code pages that contain and map the euro sign consistently with the application must be present either in a host library or in the printer. AFP fonts that support the euro sign are included in the AFP Font Collection (Program Number 5648-B33).

---

## Year 2000 Ready

PSF 3.1.0 for OS/390 does not have date dependencies and is therefore Year 2000 ready. When used in accordance with its associated documentation, PSF is capable of correctly processing, providing, and receiving date data within and between the twentieth and twenty-first centuries, provided all other products used with PSF (including software, hardware, and firmware) properly exchange accurate date data with it.

---

## About This Publication

This publication provides information about changing and maintaining Print Services Facility (PSF) 3.1.0 for OS/390 licensed program (Program Number 5655-B17), hereafter referred to as PSF.

This publication assumes 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 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 should 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 should also be familiar with PSF on the level described in *PSF for OS/390: 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, read *PSF for OS/390: Program Directory* before continuing. The program directory contains most of the instructions for installation. Information about modifying the host system for PSF is presented in this book.

If you need a description of the job control language (JCL) used to print jobs on AFP printers, a description of resources 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 OS/390: 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.

The beginning sections of this publication, through Chapter 1, describe:

- Concepts underlying PSF
- Prerequisites for system programmers
- How to use this publication
- What applications are supported by PSF

Chapter 2 through Chapter 13 are more specific. They describe:

- How to define channel-attached printers to OS/390
- How to create network nodes for SNA-attached printers
- How to install library data sets
- How to change the system environment
- How to use the PRINTDEV parameters
- How to use deferred-printing mode
- How to use direct-printing mode
- How to take advantage of the cut-sheet emulation capability of some AFP printers
- How to use 3800 line-mode conversion
- How to manage resources and select special processing parameters
- How to handle job streams formatted at different resolutions
- How PSF protects data

Chapter 14 through Chapter 22, 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 outline fonts
- How to use the APSRMARK utility to mark resources for DPF
- How to use installation exits
- How to refine PSF to affect system and printer performance
- How to connect and operate a TCP/IP printer
- How to respond to PSF error messages
- How to enable users to print to microfilm

Appendixes provide the following reference information:

- Lists of sample jobs 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 the APSUX07 installation exit
- Information about APTRCONV, the Virtual Storage Extended (VSE) (IBM System/370) resource conversion utility program
- Information about reasonable resource loading values (RRLV)
- Information about APSRCF30, the font conversion utility program (to convert 4028 fonts)



- An example of output obtained after running 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 IPDS Printer LAN Attachment

A bibliography lists the publications containing additional information about the system, hardware, or software used by PSF, or about installation and maintenance.

---

## Understanding the Syntax Notation Used in This Book

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

- The uppercase letters are to be coded as shown.
- The variable data you must provide is printed in italics. Enter specific data to replace the characters in italics; for example, for *PRTnnnn* you could enter PRT2.
- Do not enter any of the following symbols as part of a parameter or an option:

|              |     |
|--------------|-----|
| Vertical bar |     |
| Underscore   | —   |
| Braces       | { } |
| Brackets     | [ ] |

- A vertical bar between two values means that you are to select only one of the values.
- If one value in a set of values is underscored, that value is the default—that is, the value that PSF uses if no option is specified.
- If values are enclosed in braces {}, one or the other of the values in the braces must be selected.
- If a value is enclosed in brackets [], you do not have to select that value.

---

## Related Information

Publications that are referred to in this book or that contain additional information about Advanced Function Presentation (AFP), the MVS operating system, PSF, and related products are listed in the “Bibliography” on page 389.

For additional information about OS/390 and PSF for OS/390, refer to these Web pages:

<http://www.ibm.com/s390/os390>

<http://www.printers.ibm.com/pbin-id/go/?pdocs/psf390/home.html>

To obtain the latest documentation updates for OS/390 base elements and optional features that result from DOC APARs and PTFs, refer to the DOC APARs and ++HOLD DOC Web page:

[http://www.s390.ibm.com/os390/bkserv/new\\_tech\\_info.html](http://www.s390.ibm.com/os390/bkserv/new_tech_info.html)

To obtain the latest documentation updates for PSF for OS/390, refer to these members in SYS1.SAMPLIB:

| <b>Member</b> | <b>Publication</b>                                    |
|---------------|---|
| APSGCUSU      | <i>PSF for OS/390: Customization, S544-5622</i>       |
| APSGDGNU      | <i>PSF for OS/390: Diagnosis, G544-5623</i>           |
| APSGDLGU      | <i>PSF for OS/390: Download for OS/390, G544-5624</i> |
| APSGMACU      | <i>PSF for OS/390: Messages and Codes, G544-5627</i>  |
| APSGSECU      | <i>PSF: Security Guide, S544-3291</i>                 |
| APSGUSRU      | <i>PSF for OS/390: User's Guide, S544-5630</i>        |

---

# Summary of Changes

## Summary of Changes for *PSF for OS/390: Customization*, S544-5622

This publication contains additions and changes to information previously presented in *PSF/MVS System Programming Guide*, S544-3672-03, which supports PSF/MVS Version 2 Release 2.0. The technical additions and changes are marked with a “|” in the left margin.

These general changes were made throughout the book:

- The name PSF/MVS was changed to PSF.
- The name MVS Download was changed to Download for OS/390.
- References to MVSCP were deleted.
- References to downlevel MVS and JES releases were deleted.
- References to the Huffman compression were deleted.
- References to APARS or PTFS were deleted.

These new sections were added:

- Chapter 12, “Handling Job Streams That Are Formatted at Different Resolutions” on page 125
- Appendix B, “APSRMARK Long Report Format” on page 273

The section formally called “Related Publications” has been moved from the Appendix to the “Bibliography” on page 389.

The following information is new or updated:

- The optional programs shipped with PSF have been updated: AFP API and PPMP have been removed and Compatability Fonts have been added. See “What Is Shipped with PSF?” on page 1. The related IBM product that is now used instead of AFP API is AFP Toolbox for MVS.
- The list of attachments not supported through DPF has been updated. See “DPF Considerations” on page 36.
- A naming convention for identifying each type of resource in the page segment library has been added. See Chapter 4, “Working with Libraries” on page 37.
- These PRINTDEV parameters have been added to Table 2 on page 41: COLORMAP, FONT240, FONT300, NORESP, OVERLAY, OVLY240, OVLY300, PRERROR, PSEG240, PSEG300, RESPTIME. These parameters have been updated: BUFNO, CONNINTV, DISCINTV, FONTDD, OBJCONDD, OVLYDD, PSEGDD, SETUP.
- A new section, “Selecting PSF Parameters for a Dedicated Network-Attached Printer” on page 51, has been added to Chapter 6 that lists what PRINTDEV parameters to specify.
- RELMODE=NONE has been added to the list of PRINTDEV parameters you specify to share printers by the JES spool-driven method. See “JES Spool-Driven” on page 51.
- The port switching method of sharing network-attached printers has been updated. See “Port Switching” on page 52.

- Two new sections, “Sharing ESCON Channel-Attached Printers” on page 53 and “Recommended Values for PSF Timers” on page 54, have been added to Chapter 6.
- These JES2 initialization statement parameters have been updated: NPRO and PRMODE. See “Parameters for the JES2 Initialization Statement” on page 56. These JES3 initialization statement parameters have been updated: NPRO and PM; this parameter is new: PDEFAULT. See “Parameters for the JES3 Initialization Statement” on page 62.
- In the sample PSF startup procedures, the CHARS value has been changed to 60DB and the TRACE value has been changed from NO to YES. See Figure 22 on page 69.
- A new section has been added to Chapter 7 that indicates how to identify all output from a printer with an overlay to be printed on every sheet of output. See “Identifying Output from a Printer” on page 75.
- The information in SMF type 6 records has been updated. See “System Management Facility Type 6 Records” on page 86. The map of SMF type 6 records has also been added. See “Structure of SMF Type 6 Records” on page 88.
- The statement for defining DPSS has been updated. See “Defining the DPSS” on page 97.
- A new OUTPUT JCL statement for enabling direct-printing mode has been added in Chapter 8. See page 99.
- Duplex printing capability has been added as a standard AFP function with line-mode conversion. See “What is IBM 3800 Line-Mode Conversion?” on page 105.
- PSF Exit 14 in direct-printing mode has been added as a way that the line-mode conversion function is activated. The information about enabling 3800 Line-Mode Conversion has also been updated. See “How Can PSF 3800 Line-Mode Conversion Be Enabled?” on page 106.
- PSF resource exit, APSUX07, has been added to the list of where PSF resources can be specified. The description of how PSF deletes resources after processing a data set has also been updated. See Chapter 11, “Managing Resources and Selecting Special Processing Parameters” on page 111.
- The description of how the printer output buffer is calculated has been updated. See the bullet on page 113.
- The procedure for how PSF selects a copy group has been updated to include a print data set with an internal copy group. See “Selecting a Copy Group” on page 117.
- The hierarchies of form definition selection in deferred-printing mode and direct-printing mode have been updated. See “Form Definitions in Deferred-Printing Mode” on page 117 and “Form Definitions in Direct-Printing Mode” on page 118.
- The descriptions of how page definitions and fonts in deferred-printing mode are selected under JES3 have been updated. See “Page Definitions in Deferred-Printing Mode” on page 119 and “Fonts in Deferred-Printing Mode” on page 120.

- The procedure in “Searching for a Resource Specified for a Print Job” on page 121 has been updated.
- A statement that PSF tells the printer not to capture resources found inline or in user libraries has been added in several places in the publication. See “Maintaining Resource Integrity” on page 123, Chapter 13, “Understanding PSF Data Protection and Security” on page 131, and “APSRMARK Considerations” on page 136.
- These APSRMARK control statement parameters for resident fonts have been updated: RRDATE and RRTIME. See “APSRMARK Control Statement Parameters for Resident Fonts” on page 138.
- SHORT|LONG has been added as an APSRMARK control statement parameter. See “APSRMARK Control Statement Parameters (General)” on page 140 and “Control-Statement Parameters for APSRMARK” on page 158.
- Two sections describing the short and long APSRMARK reports have been added. See “Generating a Short Report of Resident Font Marking” on page 142 and “Generating a Long Report of Resident Font Marking” on page 142.
- These exit routines have been added in SYS1.SAMPLIB: APSUX07U, APSUX07V, and APSUX14. See Table 14 on page 169.
- The format of the APSGEXTP area, the APSUECE exit-communication area extension, and the IAZJSPA JES job-separator-page area have been updated. See Figure 50 on page 171, Figure 52 on page 175, and Figure 53 on page 182.
- XTP4LMRG and XTP4DPLX have been added to the input and output call for APSUX04. See “Input for APSUX04” on page 187 and “Output for APSUX04” on page 188.
- Two considerations have been added to the bullet about requesting 3800 line-mode conversion of 3800 line-mode resources to AFP print resources. See “Programming Considerations for APSUX04” on page 189.
- These indicators have been added to the XTP4FLAG field in the APSGEXTP area: XTP4WINA and XTP4LMRG. See Figure 56 on page 192.
- These programming considerations for resource-management exit APSUX07 have been added:
  - “IOCA, GOCA, and BCOCA Considerations” on page 201
  - “Considerations on Requesting an Acknowledgement” on page 204
  - “Considerations for Setting a Format Resolution Value” on page 207
  - “Considerations Related to Inhibiting Recovery” on page 210
  - “Considerations Related to Ending the SNA LU1 Conversation” on page 210

These programming considerations for APSUX07 have been updated:

- “3800-Compatible Media Origin Considerations” on page 201
- “Considerations on Suppressing an Interrupt Message Page” on page 202
- “Overlay Considerations” on page 203
- “Considerations Related to Inline Resources” on page 208

This information has been updated:

- “Initialization Call (INIT)” on page 211
- “Begin-Data-Set Call (BDSC)” on page 212

- Default and auxiliary processing for the resource-access call (RAC); see “Using RAC in Processing Page Definitions and Form Definitions” on page 217
- The section about “Direct-Printing Line Mode Exit: APSUX14” on page 221 has been added to Chapter 18.
- Information about the “Begin-Data-Set Call (BDSC)” on page 223 in the direct-print exit APSUX16 has been updated.
- A description about “ACK Per Sheet” on page 238 has been added to Chapter 19.
- New printer data has been added to Table 17 on page 239, which shows the approximate maximum printer data rates.
- The information about “Buffers” on page 239 has been updated with information about BUFNO values.
- “Requirements for Using a TCP/IP-Attached IPDS Printer” on page 243 has been updated in Chapter 20.
- The sample JCL startup procedure for a TCP/IP-attached printer (APSWPROT) has been updated. See Figure 61 on page 249.
- “Overview of the TCP/IP and PSF Interface” on page 251 has been added.
- The sample JCL startup procedure for a microfilm device (APSWPROM) has been updated. See Figure 63 on page 262.
- The APSRFTBL font resource tables have been updated with EuroReady code pages for impact printers. See Figure 66 on page 284.
- These fields in XTP7 (a DSECT in APSGEXTP that contains information that APSUX07 needs to determine what action to take) have been added or updated: XTP7HCA, XTP7WIDE, XTP7IHIB, XTP7ENDC, XTP7FMT, XTP7CRF3, and XTP7CRP2. See Figure 67 on page 296.
- These fields have been added to Table 22 on page 319, which shows an input and output summary for APSUX07 processing indicators: XTP7WIDE, XTP7IHIB, XTP7ENDC, XTP7FMT, XTP7CRF3, and XTP7CRP2.

---

## Chapter 1. Understanding PSF for OS/390

Print Services Facility (PSF) 3.1.0 for OS/390 (hereafter referred to as PSF) is a licensed printer-driver program that manages and controls the input and output data streams required by supported AFP printers or microfilm devices. PSF runs on OS/390 or MVS/ESA systems. PSF combines print data with other resources and printing controls to produce AFP data that can be printed.

PSF can be used in either of two ways. In *deferred-printing mode*, it can serve as the output writer under Job Entry Subsystem (JES). In *direct-printing mode*, it can serve as the access method to process an output data set. Printers attached by means of Systems Network Architecture (SNA) or Transmission Control Protocol/Internet Protocol (TCP/IP) cannot be used in direct-printing mode.

---

### What Is Shipped with PSF?

The following optional programs are shipped with PSF on their own tapes, with their own installation instructions in separate program directories, and with their own separate publications:

- Advanced Function Presentation Conversion and Indexing Facility (ACIF)
- AFP Upload
- Compatibility Fonts
- Download for OS/390
- IP PrintWay
- NetSpool

These programs can assist you in:

- Converting and indexing data
- Transmitting data to and from your OS/390 system

For details, refer to the publications for these programs.

---

### What Can I Do with PSF?

You can use PSF to do the following:

- Receive print jobs, access resources for print jobs, and send print commands to a printer. PSF supports both deferred printing, under job entry subsystem (JES); and direct printing, when it is specified in an application program.
- Manage resources required for the print job, such as form definitions, page definitions, fonts, page segments, overlays, and object containers.
- Identify (mark) resources as printer-resident or as managed by the Distributed Print Function (DPF) of PSF/2.
- Provide operator control of printers in deferred-printing mode.
- Provide problem diagnosis and error recovery.

You can use exits provided with PSF to do the following:

- Modify header or trailer separator pages.
- Modify, add, or suppress output records.
- Modify System Management Facility (SMF) type 6 records.

- Inspect messages for deferred and direct printing.
- Manage resources.

---

## Summary of PSF Support

PSF is the IBM printer-driver program that supports AFP printers and Computer Output Microfilm (COM) in the OS/390 environment. Using the Advanced Function Presentation (AFP) data stream and the Intelligent Printer Data Stream (IPDS), the printers supported by PSF print data that is already composed into pages. If print data has not already been composed, PSF can compose pages from line data. For more information about using PSF to print, refer to *PSF for OS/390: User's Guide*. For more information about AFP and AFP products, refer to *Guide to Advanced Function Presentation*. For information about printing to microfilm, see Chapter 22, Supporting AFP Output on Microfilm. For information about submitting jobs to print to microfilm, refer to *PSF for OS/390: User's Guide*.

In addition to the functions available for printing output, PSF provides the following:

- **3800 line-mode conversion:** PSF supports the printing of 3800 and line data in compatibility mode, providing a dynamic translation of 3800-1 print resources to enable users to print 3800 print data sets on other AFP printers.
- **Diagnostic aids:** PSF supplies diagnostic aids that maintain a record of events within PSF. If an abnormal termination occurs or if the formatted output is not as expected, information in the diagnostic aids helps you in determining and diagnosing problems.
- **Deferred-printing mode (JES):** In deferred-printing mode, PSF is an output writer, controlled by JES, that processes the spooled output from JES and sends a data stream to a page printer.
- **Direct-printing mode:** In direct-printing mode, PSF acts as a method for access to the printer.
- **Disabled-mechanism support:** A disabled mechanism, as the phrase is used here, is a printer part that is not functioning correctly—the part that controls duplex printing, for example. PSF provides disabled-mechanism support for some printers, printing the data set while ignoring the affected function. If a printer mechanism is disabled, output may not be as expected. To avoid unexpected results, you can reassign output to another printer until the disabled mechanism is repaired. For more details about which AFP printers provide this support, refer to *Advanced Function Presentation: Printer Information*.
- **Dynamic reconfiguration management (DRM):** PSF provides DRM for AFP1 and 3820 channel-attached printers. With DRM, these devices can be dynamically added to or deleted from the I/O configuration, or changed. For more information, see Chapter 2, Defining Channel-Attached Printers to OS/390.
- **Error recovery:** PSF provides for error recovery. PSF maintains internal checkpoints for each print data set. If an error or an operator command interrupts the processing of a data set, recovery routines use this checkpoint data to continue printing.
- **Hardware configuration definition (HCD):** The HCD function configures AFP1 and 3820 channel-attached printers. For more information, see Chapter 2, “Defining Channel-Attached Printers to OS/390” on page 5.



- **Operator controls:** PSF processes spooled output from JES before it sends a data stream to a printer. The operator control statements are presented in the operations guides for JES2 and JES3. For titles and order numbers, see the “Bibliography” on page 389.
- **Output record access:** PSF has an installation exit routine that can add, change, or suppress output records.
- **Print complete notification (NOTIFY):** The user can optionally request that the system supply a notification message when printing is completed.
- **Resource management:** PSF makes sure that the resources needed for printing a data set (form definitions, page definitions, fonts, page segments, overlays, and object containers) are available. For descriptions of these resources, refer to *PSF for OS/390: User's Guide*.
- **Restarting from a checkpoint:** When requested, PSF takes checkpoints at a specified interval and transmits a request to JES to record checkpoint data. A smaller checkpoint interval causes more checkpoints to be taken during the processing of a data set, and thus reduces the amount duplicated when printing is restarted from a checkpoint.
- **Separator pages:** PSF prints separator pages that show the start and end of a print job. Separator pages can be replaced, changed, or deleted through the installation exit routine and through the printer configuration.
- **System Management Facility (SMF) Type-6 Records:** PSF writes the SMF type-6 record and has an installation exit routine that can change SMF type-6 records.
- **User libraries:** PSF enables users to specify libraries containing resources (fonts, page segments, overlays, object containers, page definitions, and form definitions) for use when the print data set is processed.
- **Virtual storage constraint relief:** To allow more active printers for each address space, PSF obtains storage above the 16 MB line in 31-bit addressing mode, except where system constraints force PSF to obtain the storage below the 16 MB line.

A small part of the virtual storage constraint relief is related to the functional subsystem interface (FSI) between JES and PSF. PSF allows the FSI to operate above the 16 MB line in 31-bit addressing mode, where it has additional storage space. If, however, any PSF exit is run in the 24-bit addressing mode, PSF forces the FSI to run below the 16 MB line. For more information, see Chapter 18, “Using Installation Exits” on page 167.

---

## Printers Supported by PSF

The information in this publication applies to IBM printers supported by PSF; not all models of those printers are supported. You should be familiar with the specific printers installed on your system. For the most current list of printers and printer models supported, and information about PSF-supported printers, refer to *Advanced Function Presentation: Printer Summary* or to *Advanced Function Presentation: Printer Information*.

If you are installing a 300-pel printer, and you do not have a 300-pel font library, consider using APSRCF30 to migrate your 240-pel fonts to the 300-pel format. For

more information, see Appendix G, Font Conversion Program, 240-pel to 300-pel (APSRCF30).

**Note:** The microfilm device can operate as either a 240-pel device or a 300-pel device. For more information, see Chapter 22, Supporting AFP Output on Microfilm.

If you plan to use the PSF print-labeling function, refer to *Print Services Facility: Security Guide*. The publication you are now reading does not contain specific information about the print-labeling function.

---

## Printers Attached through an Intermediate Device

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

- Distributed Print Function (DPF) of PSF/2, running on a personal computer
- PSF Direct function of PSF/2, running on a personal computer
- PSF Direct function of PSF for AIX or InfoPrint Manager for AIX, running on an IBM RS/6000

If you are using an intermediate device, you should be familiar with the publications listed in the “Bibliography” on page 389.

The microfilm device is not supported by intermediate devices for printing to microfilm. For a list of printers supported by intermediate devices, refer to *Advanced Function Presentation: Printer Information*.

With DPF you can store (cache) 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 not to store resources found in user libraries or inline resources. The PSF Direct function does not store resources in the intermediate device.

---

## Chapter 2. Defining Channel-Attached Printers to OS/390

This chapter describes the prerequisites, methods, and steps for defining host-connected, channel-attached printers and microfilm devices to OS/390 or MVS.

---

### Prerequisites for Defining Channel-Attached Printers to OS/390

When you are installing a basic version of PSF for a particular set of printers, you must define the printers to OS/390 for the specific configuration of hardware being used. For this task you must know what printers are attached, how they are attached, and what release of OS/390 you are using.

Your hardware configuration can contain 3800 printers, 3820 printers, or channel-attached printers and microfilm devices of the AFP1 family; the definition is slightly different for each.

Your printers can be attached as follows:

- Host-connected, channel-attached printers:
  - Channel-attached by linking directly to the host
  - Channel-attached with switches providing multiple paths to the host
- SNA-attached channel printers:
  - Channel-attached to an intermediate device, to communicate with PSF through SNA

This chapter describes only host-connected, channel-attached printers. For information about defining SNA-attached printers, see Chapter 3, Establishing Network Nodes for SNA-Attached Printers.

### Using HCD to Define Channel-Attached Printers

To define host-connected, channel-attached printers, use a hardware configuration definition (HCD). See “Defining Channel-Attached Printers with HCD” on page 7.

For more information on how to configure your printers by HCD, refer to the following publications:

- *MVS/ESA Migration Planning: Dynamic I/O Configuration or OS/390 HCD Planning.*
- *MVS/ESA Hardware Configuration: Using the Dialog or OS/390 HCD User's Guide.*

Most channel-attached printers, when directly linked to the host system through an IBM S/390 channel, do not use SNA protocol. The only exceptions are 3820 channel-attached printers. Figure 1 on page 6 shows an example of a channel-attached printer configuration that is directly linked to the host system.

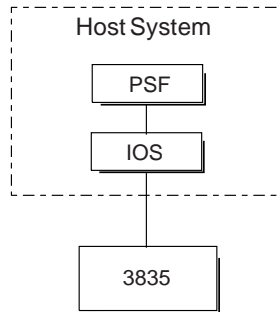


Figure 1. Example of a Channel-Attached Printer Configuration

## Using Switches with Host-Connected, Channel-Attached Printers

By using the optional two-channel switch, you can define a second channel interface for a printer attached to a single host, or define the second channel interface to another host. PSF supports this configuration only in manual mode (operator control). Using this switch can provide manual printer sharing or increased reliability if a channel fails. You must perform an I/O configuration to define all the channel paths.

## Using an Intermediate Device with Channel-Attached Printers

When you connect to channel-attached printers through an intermediate device, PSF uses Systems Network Architecture (SNA) protocol in communicating with PSF/2 or PSF for AIX. PSF/2 and PSF for AIX communicate with printers by means of S/390 channel commands. When setting up communications between PSF and an intermediate device, use Chapter 3, Establishing Network Nodes for SNA-Attached Printers and the publications for the intermediate device, listed in the “Bibliography” on page 389.

## Using ESCON Channel-Attached Printers

The Enterprise Systems Connection (ESCON) attachment improves data rate, physical path length, space requirements, and overall system connectivity.

With an ESCON card in the printer, you can use an ESCON channel, rather than the S/390 parallel channel attachment, to attach printers. The 3900 Duplex printer can be attached by this method; additional printers will be attached by this method as they become available.

The ESCON attachment also makes it possible to exploit the dynamic pathing function. Because dynamic pathing uses multiple paths, it improves reliability and performance. For information about how to define multiple paths to the device, refer to the hardware manual for your printer.

When an ESCON channel-attached printer is shared, PSF only supports manual (operator-controlled) printer-sharing; PSF does not support dynamic printer-sharing similar to automatic tape switching. See “Sharing ESCON Channel-Attached Printers” on page 53 for more details on sharing an ESCON channel-attached printer among multiple PSF hosts.

**Note:** The microfilm device can be ESCON-attached for printing to microfilm.

## Defining Channel-Attached Printers with HCD

Figure 2 shows an example of an HCD Define Device Parameters/Features panel.

**Define Device Parameters/Features**

Specify or revise the values below.

Configuration ID . . : CONFIG1  
Device number . . . : 0399                      Number of devices : 1  
Device type . . . . : 3835

| Parameter/<br>Feature | Value | P | Req. | Description                                |
|-----------------------|-------|---|------|--|
| OFFLINE               | No    |   |      | Device considered online or offline at IPL |
| <b>DYNAMIC</b>        | Yes   |   |      | Device supports dynamic configuration      |
| BURSTER               | No    |   |      | Burster-Trimmed-Stacker                    |

Figure 2. HCD Define Device Parameters/Features Panel

In the Device type option, you can specify either AFP1 or the IBM device type for channel-attached AFP1 printers. This example is for the 3835 printer.

AFP1 channel-attached printers support the DYNAMIC parameter. This parameter uses dynamic reconfiguration management (DRM) to dynamically add, delete, or change a printer in the I/O configuration.

For more information about how to use HCD, refer to *MVS/ESA Hardware Configuration Definition: Using the Dialog* or *OS/390 HCD User's Guide*.

In defining channel-attached 3820 printers, specify 3820 for the device type and the control unit type. When adding an ESCON channel-attached printer, you must attach it to an ESCON channel (type=CNC).

Following are the steps for adding an AFP1 or 3820 channel-attached printer to an I/O configuration by using HCD and DRM and then using PSF to drive the printer. Use these steps if you are using HCD and DRM in adding an AFP1 or 3820 channel-attached printer to an I/O configuration. Details not described here can be found in the publications for HCD and DRM; for titles and order numbers, see the "Bibliography" on page 389.

1. Define the printer to your I/O configuration, using HCD.
2. Activate the I/O configuration, using the MVS ACTIVATE command.
3. If the JES2 INIT deck does not already contain a printer definition for the printer that is to be started, define one, using the JES2 \$ADD command.

If a printer definition does exist but needs to be updated, use the JES2 \$T command.

For JES3, if the INIT deck needs to be updated (whether to add a new printer definition or to update an existing one), update the INIT deck and START JES3.

4. Vary the printer online, using the MVS VARY command.

You can create a new PRINTDEV statement in a new startup procedure, or you can use an existing PRINTDEV statement. To update the current startup procedure, stop the WRITER PROC:

- a. For JES2, issue the \$P command for all printers in the WRITER PROC. For JES3, issue the \*V command.
  - b. Issue the MVS CANCEL command.
5. Start the printer as you usually do.

**Note:** Be sure to update the JES INIT deck before you start PSF; otherwise you will probably encounter a PSF abend 024 with a return code of 203 or 104.

If you want to remove a printer from the I/O configuration being used by PSF, and only one printer is active in the WRITER PROC, cancel the WRITER PROC to change the device to an unallocated state. (If several printers are active in the WRITER PROC, issue the JES2 \$P command or the JES3 \*V command, followed by a \*C command, to change the device to an unallocated mode.) Use the VARY command to vary the device offline. Use the MVS ACTIVATE command to remove the printer from the I/O configuration.

---

## Chapter 3. Establishing Network Nodes for SNA-Attached Printers

This chapter 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 chapter because they cannot be configured as SNA-attached printers.

For information on defining more complex networks, refer to the publications listed under "VTAM and NCP" on page 395 in the "Bibliography" on page 389. For information on network configuration for AFCCU printers (for example: 3130, 3160, and 3935), refer to *Attachment Configuration Handbook for Printers that Use the Advanced Function Common Control Unit*.

Do not use this chapter when configuring printers supported through the Distributed Print Function (DPF) of PSF/2 or the PSF Direct function of PSF/2, InfoPrint for AIX, or PSF for AIX. Instead, refer to the PSF/2, InfoPrint for AIX, or PSF for AIX publications listed in the "Bibliography" on page 389. The graphics present 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

Define the SNA-attached printers (or the devices to which they are attached) to Advanced Communication Function/Virtual Telecommunications Access Method (VTAM) as either local-attached or communication-attached.

### Local-Attached Devices

Figure 3 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 PSF/2 or PSF for AIX. To define devices that are local-attached (printer or control units), you must use a hardware configuration definition (HCD).

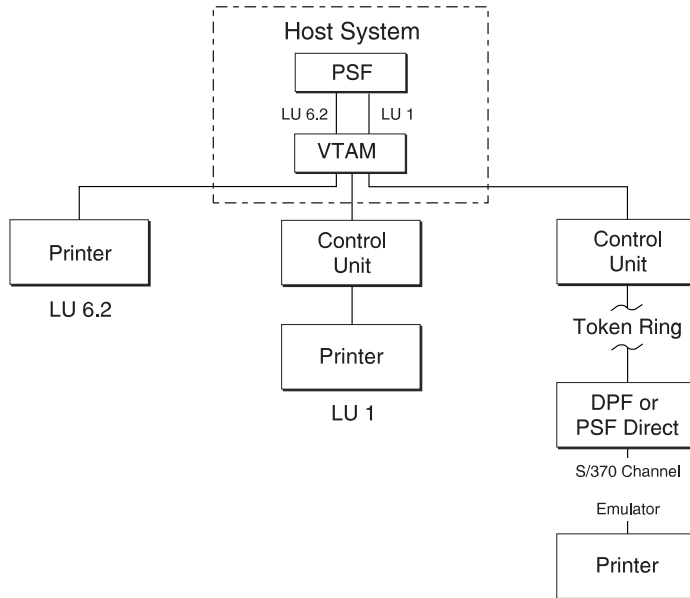


Figure 3. Sample PSF Configurations for Local-Attached Devices

## Communication-Attached Devices

Figure 4 shows three sample configurations for communication-attached devices. Here, 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 either directly, by way of a control unit, or by way of PSF/2 or PSF for AIX.

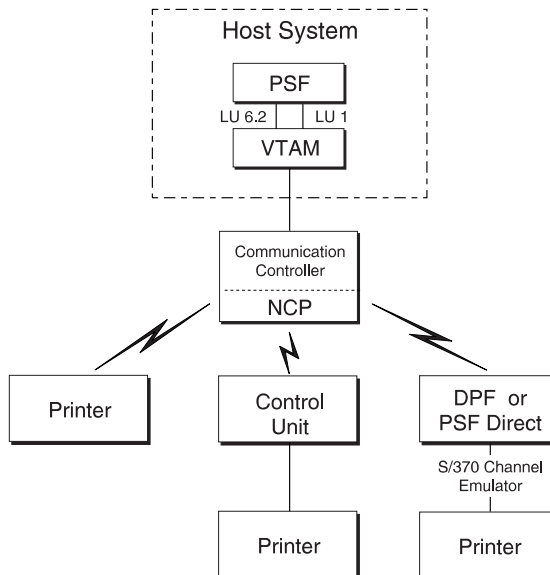


Figure 4. Sample PSF Configurations for Communication-Attached Devices

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.)



Figure 5 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

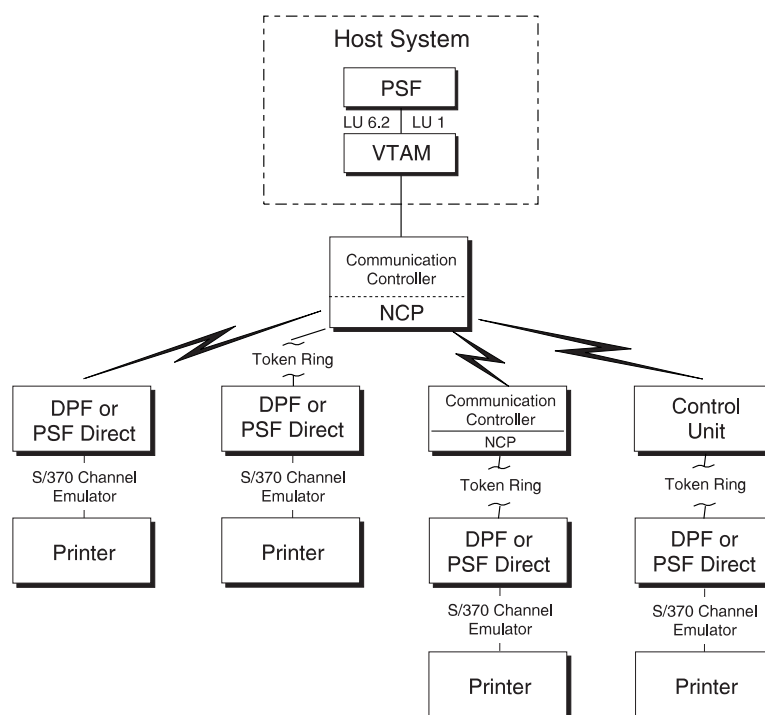


Figure 5. Sample PSF Configurations for Communication Attachment to DPF or PSF Direct

## VTAM Definitions for PSF Network Nodes

Before using an SNA-attached printer, identify the following:

- 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 26.

**Note:** The definition samples and explanations in this section are not intended to be a complete description of the statements and operands. For more information, refer to your publications for VTAM.

## SNA Devices, Attachment Mode, and Requirements

Table 1 gives an overall picture of SNA devices, the mode of attachment, and the details about the requirements.

| SNA DEVICES  | REQUIREMENTS                            |   |   |                                    |                                       |                             |   |   |   |
|--|---|---|---|------------------------------------|---------------------------------------|-----------------------------|---|---|---|
|  | Define a Local-Attached Device (page 9) | Define a PSF application node (page 14) | Define the logon-mode table entry (page 16) | Define local major nodes (page 19) | Define switched major nodes (page 21) | Define NCP Major Node       |   |   |   |
| Switched line (page 26)  |   |   |   |                                    |                                       | Non-switched line (page 28) | Multi-point non-switched line (page 31) | Token-ring subsystem connection (page 34) |   |
| SNA devices are attached in one of the following ways, each of which requires definitions for NCP. |   |   |   |                                    |                                       |                             |   |   |   |
| <b>Local-Attached</b>  | X                                       | X                                       | X   | X                                  |                                       |                             |   |   |   |
| <b>Communication-Attached: Token Ring through:</b>   |   |   |   |                                    |                                       |                             |   |   |   |
| • Communication Controller Token-Ring Subsystem  |   | X                                       | X   |                                    | X                                     |                             |   |   | X |
| • Remote Communication Controller Token-Ring Subsystem   |   | X                                       | X   |                                    | X                                     |                             | X                                       |   | X |
| • Remote Control Unit Token-Ring Gateway   |   | X                                       | X   |                                    |                                       |                             |   | X   |   |
| <b>Communication-Attached: Switched Line</b>   |   | X                                       | X   |                                    | X                                     | X                           |   |   |   |
| <b>Communication-Attached: Nonswitched Line</b>  |   | 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 14.
- Logon-mode table entry. See “Defining the Logon-Mode Table Entry” on page 16.
- Physical unit (PU) and logical unit (LU) statements in a local major node. See “Defining Local Major Nodes” on page 19.

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.) See “Defining Local Major Nodes” on page 19.

### Communication-attached devices (switched line)

- PSF application node. See “Defining a PSF Application Node” on page 14.

- Logon-mode table entry. See “Defining the Logon-Mode Table Entry” on page 16.
- PU, PATH, and LU statements in a switched major node. See “Defining Switched Major Nodes” on page 21.
- LINE and PU statements in a switched line GROUP under an NCP major node. An NCP generation is required. See “Switched Line” on page 26.

#### **Communication-attached devices (nonswitched line)**

- PSF application node; see “Defining a PSF Application Node” on page 14.
- Logon-mode table entry; see “Defining the Logon-Mode Table Entry” on page 16.
- 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 28.

#### **Communication-attached devices (token ring)**

Through a communication controller token-ring subsystem:

- PSF application node. See “Defining a PSF Application Node” on page 14.
- Logon-mode table entry. See “Defining the Logon-Mode Table Entry” on page 16.
- PU, PATH, and LU statements in a switched major node. See “Defining Switched Major Nodes” on page 21.
- 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 34.

Through a remote communication controller token-ring subsystem:

- PSF application node. See “Defining a PSF Application Node” on page 14.
- Logon-mode table entry. See “Defining the Logon-Mode Table Entry” on page 16.
- PU, PATH, and LU statements in a switched major node. See “Defining Switched Major Nodes” on page 21.
- LINE and PU statements in a nonswitched line GROUP under two NCP major nodes. An NCP generation is required. See “Nonswitched Line” on page 28.
- 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 34.

Through a remote control unit token-ring gateway:

- PSF application node. See “Defining a PSF Application Node” on page 14.
- Logon-mode table entry. See “Defining the Logon-Mode Table Entry” on page 16.
- 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 31.

## 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 from 1 to 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 should 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 will not receive an error message, but processing will stop at the end of the first line and will be unable to continue with the second line of the statement. The printer, therefore, will be incorrectly initialized.

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

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 on the sift-down effect, refer to *OS/390 eNetwork Communications Server: SNA Resource Definition Reference*.

The network configuration examples that follow in this chapter do not follow this 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 associated with LU type 6.2 are used; for a 3812 printer, the parameters and values 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. See Figure 6 on page 15.

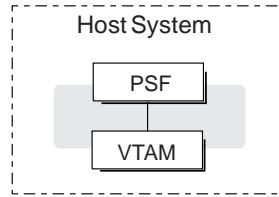


Figure 6. PSF Application Node. One logical unit (LU) is shown. The APPL statement defines an LU.

This APPL statement also defines some of the communication capabilities of the application program. You should 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 7 is used.

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

Figure 7. Sample APPL Statement. Commonly found in SYS1.VTAMLST.

**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, refer to your publications for VTAM.

The application program node must also be specified in the APPLID parameter of the PRINTDEV statement in the PSF startup procedure. 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 PSF/2 and PSF for AIX)

#### For an LU type 6.2 device

```
IBM3820C MODEENT LOGMODE=IBM3820C,FMPROF=X'13',TSPROF=X'07',
PRIIPROT=X'B0',SECIPROT=X'B0',
COMIPROT=X'D0B1',RUSIZES=X'85C7',
PSERVIC=X'060200000000000000002000',
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',
PRIIPROT=X'B1',SECIPROT=X'B0',
COMIPROT=X'7080',RUSIZES=X'85C7',
PSERVIC=X'014000010000000001000000',
PSNDPAC=X'02',SRCVPAC=X'02',
SSNDPAC=X'00'
```

Figure 8. Sample Logon-Mode Table Entry for Local-Attached Devices. Continuation characters in column 72 are not shown. These statements are commonly found in SYS1.VTAMLIB(MODETAB1).

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

### Communication-Attached Entries (Excluding PSF/2, PSF for AIX, and 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'060200000000000000002000',
          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'014000010000000001000000',
          PSNDPAC=X'03',SRCVPAC=X'03',
          SSNDPAC=X'00'
```

Figure 9. Sample Logon-Mode Table Entry for Communication-Attached Devices. Continuation characters in column 72 are not shown. These statements are commonly found in SYS1.VTAMLIB(MODETAB1).

For VTAM definitions for DPF, refer to the PSF/2 and PSF for AIX publications listed in the “Bibliography” on page 389.

**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'060200000000000000002000',
          PSNDPAC=X'xx',SRCVPAC=X'yy',
          SSNDPAC=X'00'
```

Figure 10. Sample Logon-Mode Table Entry for Token-Ring Devices. Continuation characters in column 72 are not shown in the above statements. These statements are commonly found in SYS1.VTAMLIB(MODETAB1).

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, refer to 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, either with the DLOGMOD operand of the LU statement for a printer or with the LOGMODE option of the PRINTDEV statement in the PSF startup procedure. 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 on setting the pacing value, refer to 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 2 leftmost hexadecimal digits apply to data sent by the secondary LU (printer). The 2 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:

$$\text{LENGTH} = m \times 2^n$$

Thus, RUSIZES=X'8585' specifies length = 256 ( $8 \times 2^5$ ) 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 via 3745 with SDLC modem: X'86F8'
  - Communication-attached via 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 by means of 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 connected to an LU type 1 control unit, the connection between VTAM and the control unit is defined with a PU statement. Each printer attached to the control unit is defined with an LU statement. Figure 11 shows an example.

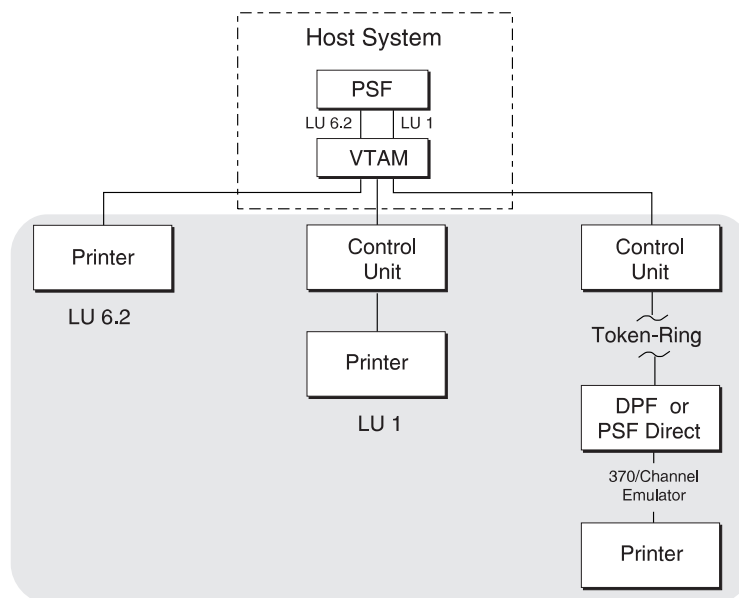


Figure 11. Example Showing Three Local Major Nodes. A node can be for either a printer or a control unit.

Figure 12 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 12. Sample Local Major Node Definitions for Different LU Protocols. These statements are commonly found in SYS1.VTAMLST.

**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 activating the PU. 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 on customizing this value, refer to your publications for VTAM.

**LU**

Describes the logical unit (LU). The name specified on the LU statement must also be specified on the LUNAME parameter in the PRINTDEV statement of the PSF startup procedure for the printer.

**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. This parameter is required when MGMTMODE=DIALIN is specified in the PRINTDEV statement. For more information, see "Defining the Logon-Mode Table Entry" on page 16. If LOGMODE is specified in the PRINTDEV statement, it overrides the name specified here when either MGMTMODE=IMMED or MGMTMODE=OUTAVAIL is specified.

**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 coming

from the control unit. For example, for a 3174 control unit 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 specified in the APPLID parameter in the PRINTDEV statement. This parameter is required only when MGMTMODE=DIALIN in the PRINTDEV statement. For more information, see the note on page 15.

**Note:** When LOGAPPL is specified, the host application program node must be active before the device LU node is activated. Also, if MGMTMODE=DIALIN, 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 supplied by IBM is used.

#### **PACING**

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

## **Defining Switched Major Nodes**

Figure 13 shows major nodes switched to link VTAM with a communication controller.

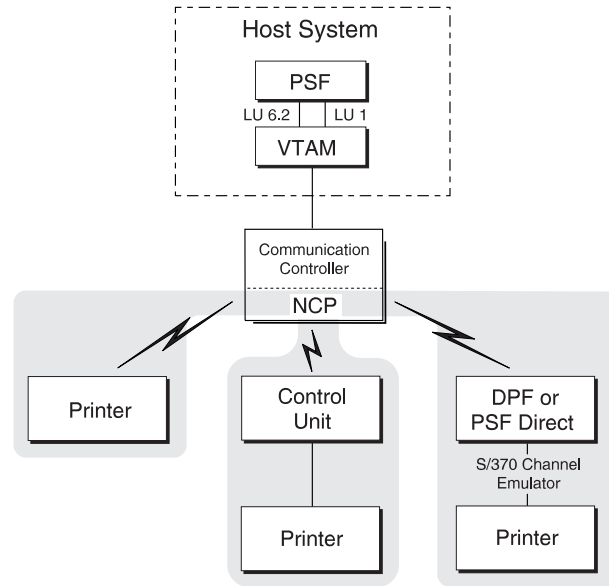


Figure 13. Sample Configuration of Switched Major Nodes. An NCP generation is required.

Figure 14 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,
           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 14. Sample Definitions of Switched Major Nodes. Continuation characters in column 72 are not shown. These statements are commonly found in SYS1.VTAMLST(member name).

An NCP generation is required. To do the NCP generation, see “Defining Network Control Program (NCP) Major Nodes” on page 25.

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 should physically disconnect the physical unit when the logical unit terminates 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, refer to 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 building 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. The bits in the station ID are defined as follows:

|                    |                             |
|--------------------|-----------------------------|
| <b>Bits 0–3:</b>   | Reserved                    |
| <b>Bits 4–7:</b>   | Contain the value of PUTYPE |
| <b>Bits 8–15:</b>  | Contain X'00'               |
| <b>Bits 16–27:</b> | Contain the value of IDBLK  |
| <b>Bits 28–47:</b> | 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 should 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 being defined.

**LU**

Defines the logical unit. The name specified in the LU statement must match the name specified for the LUNAME option in the PRINTDEV statement of the PSF startup procedure.

**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. This parameter is required when MGMTMODE=DIALIN is specified in the PRINTDEV statement. See “Defining the Logon-Mode Table Entry” on page 16 for more information. If LOGMODE is specified in the PRINTDEV statement, it overrides the name specified here when either MGMTMODE=IMMED or MGMTMODE=OUTAVAIL is specified.

**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 specified in the APPLID parameter in the PRINTDEV statement. This parameter is required only when MGMTMODE=DIALIN in the PRINTDEV statement. For more information, see the note on page 15.

**Note:** When LOGAPPL is specified, the host application program node must be active before the device LU node is activated.

When MGMTMODE=DIALIN, 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 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 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 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 listed in “NCP Generation” on page 26 should also be supplied to VTAM as part of the NCP major node definition. For a diagram of a sample definition, see Figure 15. For an overview of the VTAM and NCP definitions that PSF requires see Table 1 on page 12.

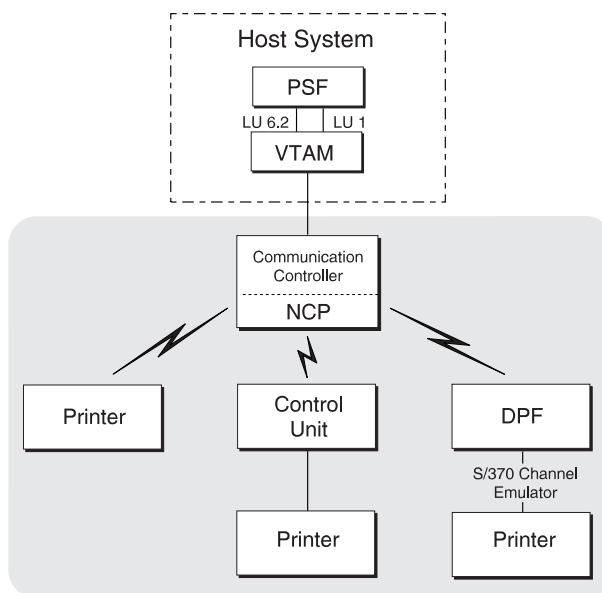


Figure 15. Sample NCP Major Node. Lines can be switched or nonswitched.

## 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. Some of the operands, however, 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 used by both NCP and VTAM are verified during NCP generation but not during VTAM network definition. Therefore, the NCP should 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; the following 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 connecting 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 in this section are not intended to be a complete description of the macroinstructions and operands. For more information, refer to your publications for NCP.

### Switched Line

Figure 16 on page 27 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 16. Sample NCP Macroinstructions for Switched lines. These statements are commonly found in SYS1.VTAMLST(member name), and are entered during the generation of NCP.*

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 time-out 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 17 shows sample NCP macro instructions for three nonswitched lines.

|   |         |   |  |
|---|---------|---|--|
| <b><u>For an LU type 6.2 device</u></b> |         |   |  |
| 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=zzzzzzz, MODETAB=MODETAB1           |  |
| <b><u>For an LU type 1 device</u></b>   |         |   |  |
| 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 17. Sample NCP Macro instructions for Nonswitched Lines. Continuation characters in column 72 are not shown. These statements are commonly found in SYS1.VTAMLST(member name), and are input during the generation of NCP.

This example consists of the following instructions:

**GROUP**

Describes a group of communication lines:

**ACTIVTO**

Specifies the time-out value that the secondary NCP waits for communication from the primary NCP before initiating 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 time-out 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 respectively 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 will send to the physical unit before requesting a response. The recommended range for this value is from 3 to 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 specified in the LU statement must match the name specified for the LUNAME parameter for the printer FSA in the PRINTDEV statement of the PSF startup procedure.

**DLOGMOD**

Required when MGMTMODE=DIALIN is specified in the PRINTDEV statement. 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 16. If LOGMODE is specified in the PRINTDEV statement, it overrides the name specified here when either MGMTMODE=IMMED or MGMTMODE=OUTAVAIL.

**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 specified in the APPLID parameter in the PRINTDEV statement. Required only when MGMTMODE=DIALIN in the PRINTDEV statement. For more information, see the note on page 15.

**Note:** When LOGAPPL is specified, the host application program node must be active before the device LU node is activated.

Also, if MGMTMODE=DIALIN, 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 supplied by IBM is used.

## Multipoint Nonswitched Line

Figure 18 shows an example of macroinstructions for generating NCP for multipoint lines.

| <u>For an LU type 6.2 device</u> |         |   |  |
|----------------------------------|---------|---|--|
| 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 |  |
| <u>For an LU type 1.0 device</u> |         |   |  |
| LN2804                           | LINE    | ADDRESS=(004,HALF),CLOCKNG=EXT,RETRIES=(5,1,2),PACING=3,    |  |
|                                  | SERVICE | VPACING=3,SPEED=9600  |  |
|                                  | ORDER   | (PU2804A,PU2804B)   |  |
| PU2804A                          | PU      | ADDR=C1,ANS=STOP,I_RETRY=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,I_RETRY=NO,MAXDATA=265,MAXOUT=3,           |  |
|                                  |         | MAXLU=10,PUTYPE=2   |  |
| LU38204B                         | LU      | LOCADDR=01,MODETAB=MODETAB1,DLOGMOD=IBM3820,                |  |
|                                  |         | LOGAPPL=PSFFSA1   |  |

Figure 18. Sample NCP Macroinstructions for Multipoint Lines. Continuation characters in column 72 are not shown.

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, 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 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 will again poll a secondary station before advancing 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 requesting a response. The recommended range for this value is from 3 to 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 specified in the LU statement for the device must match the name specified for the LUNAME parameter for the printer FSA in the PRINTDEV statement of the PSF startup procedure. 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 16. If LOGMODE is specified in the PRINTDEV statement, it overrides the name specified here when either MGMTMODE=IMMED or MGMTMODE=OUTAVAIL.

**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.

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 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 coming from the control unit.

**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 a USS definition table. If USSTAB is not specified, a USS definition table supplied by IBM is searched when VTAM receives character-coded input from the logical unit.

**Note:** The USSTAB operand may 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 should 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 will 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 will be 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 page 18.

#### 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 establishing a network, you should consider the following:

- 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 two places:

- The LOGMODE parameter of the PRINTDEV statement.
- 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 option is specified in the PRINTDEV statement, VTAM uses it. If that option is not specified, VTAM uses the DLOGMOD operand. If neither LOGMODE nor DLOGMOD is specified, the default DLOGMOD entry for VTAM is used.

If the LOGMODE option is invalid, an error message is issued even if the DLOGMOD operand is valid. Conversely, if the DLOGMOD operand is invalid but the LOGMODE option 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 MGMTMODE=DIALIN.

- The APPLID name specified in the PRINTDEV statement

## LOGAPPL and the LU Statement

When MGMTMODE=DIALIN is specified in the PRINTDEV statement, 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.

---

## PSF PRINTDEV Statement Considerations for Switched Lines

The kind of printing session PSF establishes depends on the parameter specified for MGMTMODE in the PRINTDEV statement. For communication-attached devices on switched lines, choose:

- MGMTMODE=DIALIN
- MGMTMODE=OUTAVAIL
- MGMTMODE=IMMED

MGMTMODE=DIALIN is 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 should also specify DISCNT=YES. For

more information, see the DISCNT statement, described in Defining Switched Major Nodes, and Figure 14 on page 22.

The CALL=IN | OUT | INOUT parameter determines whether PSF waits for the printer operator to dial in or prompts the VTAM operator to dial out.

MGMTMODE=OUTAVAIL and MGMTMODE=IMMED are 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.

**Note:** When a printer has not been dialed, message APS925I is issued to remind the host operator that the control unit needs to be dialed.

The CALL=OUT | INOUT parameter should be used. LOGAPPL is not required.

---

## 3130, 3160, and 3935 Configuration

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

---

## DPF Considerations

When setting up communication between DPF and a printer, use the PSF/2 publications listed in the “Bibliography” on page 389.

**Note:** The following are not supported through DPF:

- Attachment to a microfilm device.
- Attachment to a printer in automatic resolution mode.
- Attachment to a printer that supports N-UP (PSF disables the N-UP function).
- Attachment to a printer that supports extended overlays or page segments (PSF disables the extended overlay or page segment function).
- Attachment to a printer that supports the x'84' triplet on the Activate Resource IPDS command (PSF disables the x'84' triplet support).
- Attachment to a printer in cut-sheet emulation mode (PSF disables the cut-sheet emulation function).

---

## PSF Direct Considerations

When setting up communication between PSF Direct and a printer, use the PSF/2 and PSF for AIX publications listed in the “Bibliography” on page 389.

**Note:** Attachment to a microfilm device is not supported through PSF Direct.

---

## Chapter 4. Working with Libraries

This chapter 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.

You can define the following libraries that contain the resources needed for printing:

- Font library
- Form definition library
- Object container library
- Overlay library
- Page definition library
- Page segment library

Because multiple resource types are now contained in the page segment library and PSF does not enforce a prefix for the eight-character resource name, you should define a naming convention that identifies each type of resource in the page segment library. IBM recommends a two-character prefix:

|           |                        |
|-----------|------------------------|
| <b>B1</b> | BCOCA resources        |
| <b>G1</b> | GOCA resources         |
| <b>I1</b> | IOCA resources         |
| <b>S1</b> | Page segment resources |

**Note:** When 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.

For more information about installing resources in libraries, refer to the PSF program directory provided with your PSF tapes.

---

### Library Format

Libraries must be partitioned data sets (PDS) with record formats of variable-blocked machine (VBM), variable-blocked ANSI <sup>1</sup> (VBA), variable machine (VM), or variable ANSI (VA) controls.

---

### 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 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 it finds first. When PSF is started, libraries are searched in the order in which they were concatenated. For information about concatenating libraries, refer to the JCL user's guide for your operating system.

With user library support, you may not want to concatenate all such libraries to the system libraries. You should not concatenate to the system libraries any libraries

---

<sup>1</sup> ANSI stands for American National Standards Institute.

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 processing the print data set, PSF accesses user libraries for AFP resources (fonts, page segments, overlays, page definitions, form definitions, object containers, BCOCA, GOCA, and IOCA). Print-job submitters specify user libraries by specifying the USERLIB parameter in their JCL OUTPUT statement, as in this example:

```
//OUT1  OUTPUT  USERLIB=(USER.IMAGES.AFP,USER.FONT.S.AFP)
```

This example directs PSF to search for AFP resources in the two specified user libraries. If unable to find the resources in the user libraries, PSF searches the system libraries defined in the startup procedure.

You can use RACF to control access to user libraries. 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 has finished printing, PSF deletes the resource from the printer.

Because PSF runs as a started task and must be permitted access to a user library specified with the USERLIB parameter, the RACF-Started Procedures Table (ICHRIN03) must include an entry for PSF, declaring it privileged or trusted. For information about the Started Procedures Table, refer to the *RACF System Programmer's Guide*.

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 may necessitate reconfiguring shared volumes or storage classes, and allocating the resource libraries to such volumes or storage classes.
- PSF does not support the USERLIB parameter 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 all tasks in an address space serializes on SYSZTIOT, which could 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.

For more information about user libraries, refer to *PSF for OS/390: User's Guide*.

---

## Chapter 5. Changing the System Environment

After you install PSF, you must change the system environment—system libraries, JES, and VTAM—before you can use PSF 3.1.0 for OS/390. The changes you make depend on whether PSF has already been installed, which printers are being used, and whether printers are controlled with JES.

---

### How to Change the System Environment for PSF 3.1.0 for OS/390

To change the system environment for PSF 3.1.0 for OS/390:

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 7, “Using Deferred-Printing Mode” on page 55.

You should understand the hierarchy by which resources are selected when the printer is used with JES. The JES FCB and UCS parameters and work selection criteria are described in Chapter 11, “Managing Resources and Selecting Special Processing Parameters” on page 111.

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 H, Installation Verification Program (IVP) Example. These jobs are placed in SYS1.PROCLIB when PSF is installed.

For more information, see Chapter 7, Using Deferred-Printing Mode.

- To set up the printer for use in direct-printing mode (host-connected, channel-attached non-SNA printers only), see Chapter 8, Using Direct-Printing Mode. 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. Ensure that PSF has access to RACF-protected resource libraries, by adding an entry for PSF to the RACF-Started Procedures Table (ICHRIN03) and declaring PSF to be privileged or trusted. This ensures that PSF has access to RACF-protected resource libraries. For information about the Started Procedures Table, refer to *RACF System Programmer's Guide* for your operating system, listed in the “Bibliography” on page 389.

3. Set up for VTAM/NCP (SNA only):

An SNA-attached printer is a networked device that uses ACF/VTAM alone or with ACF/NCP to communicate with your system. Before you can use the printer, you must update the ACF/VTAM network definition or perform an ACF/NCP generation.

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

For more information, see Chapter 3, Establishing Network Nodes for SNA-Attached Printers.

For more information about host-connected printers that are channel-attached, see “Defining Channel-Attached Printers with HCD” on page 7. For information about printers attached with TCP/IP, see Chapter 20, “Connecting a TCP/IP-Attached IPDS Printer” on page 243.

---

## Default PPT Entry

IBM supplies an internal default program properties table (PPT) entry with MVS 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, refer to the *Initialization and Tuning Reference* for your operating system, listed in the “Bibliography” on page 389.

Specifying NODSI (nonexclusive use of data sets) in the PPT enables other programs to change the resources while PSF is running. Specifying DSI and running PSF with DISP=SHR causes other programs to wait until PSF is no longer running before the other programs can update the PSF libraries. Modification of any other values may cause JES to abend, or other unexpected results to occur.

## Chapter 6. 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 7, Using Deferred-Printing Mode, and Chapter 8, Using Direct-Printing Mode.

Table 2 describes the initialization parameters that can be included in a PRINTDEV statement.

| <i>Table 2 (Page 1 of 11). Initialization Parameters for the PRINTDEV Statement: Summary</i> |   |
|--|---|
| Parameter  | Description   |
| ACQINTV<br>(optional)  | <p>Specifies the acquire interval in seconds. This is the time period between releasing a printer and attempting to acquire it again. ACQINTV is used only when RELMODE=TIME.</p> <p><b>Syntax:</b> ACQINTV=<i>nn</i>, where <i>nn</i> is a value from 0 to 86400.</p>  |
| APPLID<br>(required for SNA-attached printers only)  | <p>Specifies the name of the VTAM application-program node for a functional subsystem application (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 the following two values in the VTAM printer-network definition:</p> <ul style="list-style-type: none"> <li>The name of the APPL statement in the PSF node definition for the application program</li> <li>The value for the LOGAPPL parameter in the node definition for the printer</li> </ul> <p>For more information, see “Defining a PSF Application Node” on page 14, and “Defining Switched Major Nodes” on page 21.</p> <p><b>Syntax:</b> APPLID=<i>name</i></p> <p><b>Note:</b> Only for SNA-attached printers (required).</p>  |
| BUFNO<br>(optional)  | <p>Specifies a factor for allocating real, fixed storage used in printer I/O operations. (The BUFNO value is multiplied by 32 KB (32768 bytes) to determine the total amount of reserved storage.) The value of BUFNO can range from 1 to 10000.</p> <p><b>Syntax:</b> BUFNO=<i>nnnnn</i></p> <p><b>Notes:</b></p> <ol style="list-style-type: none"> <li>Only for channel-attached printers that do not use RPM, DPF, or PSF Direct.</li> <li>For more information, see “Buffers” on page 239.</li> </ol> <p><b>Attention:</b> The I/O buffer expansion factor has changed from 4 KB (PSF/MVS 2.2.0 or earlier) to 32 KB (PSF 3.1.0 for OS/390). This means that use of a startup procedure specifying BUFNO values defined for PSF/MVS 2.2.0 or earlier is now reserving eight times as much I/O buffer space. This might not concern you because a majority of BUFNO space is now reserved above the 16 MB line instead of below it. However, a review of the BUFNO values defined in your startup procedure is recommended.</p> |
| CHARS<br>(optional)  | <p>Specifies up to four default font names. A font name must contain 1 to 4 alphanumeric or national characters. For information on selecting fonts, see “How PSF Selects Resources” on page 115.</p> <p><b>Syntax:</b> CHARS=(<i>fontname1</i> [, <i>fontname2</i>] [, <i>fontname3</i>] [, <i>fontname4</i>])</p>   |

Table 2 (Page 2 of 11). Initialization Parameters for the PRINTDEV Statement: Summary

| Parameter  | Description  |
|--|--|
| <p>COLORMAP<br/>(optional)</p>                           | <p>Specifies the member name of the printer default color mapping table (1 to 8 alphanumeric or national characters). The color mapping table is an object container resource used to tell the printer what colors to use for various identifiable parts of the data stream. IBM recommends a prefix of <b>M1</b> for color mapping table resources. The full member name must be specified because PSF does not add the prefix. For example,</p> <p><b>COLORMAP=M1SYSTEM</b></p> <p>PSF uses a hardcoded color mapping table named M1RESET to reset all color mapping table specifications in the printer when these conditions are in place:</p> <ul style="list-style-type: none"> <li>• PSF is attached to a device that supports the color mapping table object container.</li> <li>• The COLORMAP parameter is not specified in the OUTPUT JCL statement for the submitted job (see <i>PSF for OS/390: User's Guide</i> for more information about the OUTPUT JCL statement).</li> <li>• The PRINTDEV JCL statement does not contain a COLORMAP parameter.</li> </ul> <p>For printers not supporting color mapping table object container resources, this parameter is ignored.</p> <p><b>Syntax:</b> <b>COLORMAP</b>=(membername)</p> |
| <p>COMPRESS<br/>(optional)</p>                           | <p>Specifies whether PSF should compress data (blanks) in line data sent to printers, including printers attached to PSF/2 or PSF for AIX. Selecting YES causes PSF to use an IPDS command to compress contiguous blanks. (The printer will decompress the data.)</p> <p><b>Syntax:</b> <b>COMPRESS=</b><u>YES NO</u></p> <p><b>Notes:</b></p> <ol style="list-style-type: none"> <li>1. This parameter will probably not improve data transmission for channel-attached printers.</li> <li>2. Blank compression is a data-compression function in PSF. It reduces the amount of data that has to be sent through the attachment (host-connected channel, SNA, or TCP/IP) to the printer.</li> <li>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.</li> <li>4. In certain conditions, this support may improve the throughput of SNA-attached or TCP/IP-attached printers. For more information, see "Blank Compression for Line Data (COMPRESS Parameter)" on page 238.</li> </ol>   |
| <p>COMSETUP<br/>(required for microfilm device only)</p> | <p>Specifies the member name (1 to 8 alphanumeric or national characters) of the printer default microfilm setup resource object container generated with the utility provided with your microfilm device. The COMSETUP parameter is used only when output is being sent to a microfilm device. IBM recommends a prefix of <b>H1</b> for microfilm setup resources, and the full member name must be specified, because PSF does not add a prefix. For an example of a startup procedure that uses the COMSETUP parameter with a full member name, see Figure 63 on page 262. For example,</p> <p><b>COMSETUP=H1SETUPS</b></p> <p>When PSF is attached to an microfilm device, if the COMSETUP parameter is missing from an AFP job submitted to that microfilm device, or if no member name has been specified or provided in the PRINTDEV, PSF terminates the job. For more information about printing to microfilm, see Chapter 22, Supporting AFP Output on Microfilm.</p> <p>For non-microfilm devices, this parameter is ignored.</p> <p><b>Syntax:</b> <b>COMSETUP</b>=membername</p>   |



Table 2 (Page 3 of 11). Initialization Parameters for the PRINTDEV Statement: Summary

| Parameter              | Description   |          |                               |            |                    |
|------------------------|---|----------|-------------------------------|------------|--------------------|
| CONNINTV<br>(optional) | <p>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 terminates the FSA.</p> <p>For all attachments except TCP/IP, if CONNINTV is not specified, PSF attempts to connect for an unlimited time. If the attachment is TCP/IP and CONNINTV is not specified, PSF defaults to a 10-minute connection interval (CONNINTV=600). For all attachments, if CONNINTV=0, PSF attempts to connect for an unlimited time.</p> <p>For SNA, the connect interval is set when a session request is queued after a busy or not available condition.</p> <p>For TCP/IP, the connect interval is set when attempting to connect to the printer (connect, open session, assign).</p> <p>For IPDS, the connect interval is set when a dialog is restarted after a release (RELMODE=IDLE TIME).</p> <p><b>Syntax: CONNINTV=0 nn</b>, where <i>nn</i> is a value from 0 to 86400. The default value is:</p> <table border="0"> <tr> <td><b>0</b></td> <td>All attachments except TCP/IP</td> </tr> <tr> <td><b>600</b></td> <td>TCP/IP attachments</td> </tr> </table>  | <b>0</b> | All attachments except TCP/IP | <b>600</b> | TCP/IP attachments |
| <b>0</b>               | All attachments except TCP/IP   |          |                               |            |                    |
| <b>600</b>             | TCP/IP attachments  |          |                               |            |                    |
| DATAACK<br>(optional)  | <p>Specifies the data-check exceptions the printer reports to PSF. Printers report two data-check exceptions:</p> <ul style="list-style-type: none"> <li>• Print-positioning exceptions, which occur when attempts are made to print outside the valid printable area or off the logical page. Some printers use exception highlighting to mark print-positioning exceptions on the printed page.</li> <li>• Invalid-character exceptions, which occur when attempts are made to use a code point that is not assigned to a character.</li> </ul> <p><b>Syntax: DATAACK= BLOCK UNBLOCK BLKCHAR BLKPOS</b></p> <p><b>DATAACK=BLOCK</b><br/>Specifies that the printer is not to report data-check exceptions to PSF.</p> <p><b>DATAACK=UNBLOCK</b><br/>Specifies that the printer is to report all data-check exceptions to PSF.</p> <p><b>DATAACK=BLKCHAR</b><br/>Specifies that the printer is not to report invalid-character exceptions to PSF. (The printer reports only print-positioning exceptions.)</p> <p><b>DATAACK=BLKPOS</b><br/>Specifies that the printer is not to report print-positioning exceptions to PSF. (The printer reports only invalid-character exceptions.)</p> <p>Regardless of what the PRINTDEV statement specifies, PSF defaults to DATAACK=UNBLOCK when printing security resources. For more information on security resources, refer to <i>Print Services Facility: Security Guide</i>.</p> <p><b>Notes:</b></p> <ol style="list-style-type: none"> <li>1. Unless PIMSG=NO, data-check exceptions also produce error messages. (PIMSG is described on page 49.)</li> <li>2. For microfilm devices, see Chapter 22, Supporting AFP Output on Microfilm.</li> </ol> |          |                               |            |                    |

Table 2 (Page 4 of 11). Initialization Parameters for the PRINTDEV Statement: Summary

| Parameter              | Description   |
|------------------------|---|
| DISCINTV<br>(optional) | <p>Specifies the disconnect interval in seconds. When no output is available from JES for this time period, PSF ends the session with the printer. PSF makes sure that all pages from the last job reach the stacker before ending the session and allowing that job to be purged from the spool.</p> <p>If DISCINTV=0, a lack of output does not cause PSF to end the printer output. (See Note 2 for additional information.)</p> <p>If RELMODE=IDLE or TIME, DISCINTV is ignored. For more information, see “Selecting PSF Parameters for Shared Printers” on page 51.</p> <p><b>Syntax:</b> DISCINTV=<i>nnn</i>, where <i>nnn</i> is a value from 0 to 86400. The default value is 0.</p> <p><b>Notes:</b></p> <ol style="list-style-type: none"> <li>1. For SNA-attached and TCP/IP-attached printers only. TCP/IP attachment includes printers attached with the i-data 7193 IPDS Printer LAN Attachment (see Appendix K, “Installing the i-data 7913 IPDS Printer LAN Attachment” on page 347).</li> <li>2. Some printers have a network inactivity timer. If the timer is enabled, the DISCINTV value must be less than that of the inactivity timer and greater than 0. Otherwise, PSF abends when the inactivity timer expires.<br/><br/>The inactivity timer can be set with the printer-configuration procedure.</li> <li>3. When some printers, such as the 3820 printer, become inoperative, a <i>local intervention timer</i> is started. When the timer expires, an action code X'22' negative acknowledgment (NACK) is sent to PSF. PSF then issues message APS923I and performs the action specified in the FAILURE 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.</li> </ol> |
| DPAGELBL<br>(optional) | Refer to <i>Print Services Facility: Security Guide</i> for a description of this parameter.  |
| DSHDR<br>(optional)    | <p>Identifies an OUTPUT statement that specifies the form definition and page definition used to format and print data set header pages.</p> <p><b>Syntax:</b> DSHDR=<i>*.label</i></p> <p><b>Note:</b> Not valid for direct-printing mode.</p>   |
| DUMP<br>(optional)     | <p>Specifies that a conditional dump of the PSF address space is to be taken when a specific reason code, restartable abend, or message occurs.</p> <p><b>Syntax:</b> DUMP=(<i>[reascode ABD0nnn][,msgid,count]</i>)</p> <p><b>Note:</b> For a description of the syntax of this parameter and an explanation of restartable abends, refer to <i>PSF for OS/390: Diagnosis</i>. To determine whether an abend is restartable, refer to <i>PSF for OS/390: Messages and Codes</i>.</p>   |
| FAILURE<br>(optional)  | <p>Specifies PSF action after a printer failure, an SNA session failure, or a TCP/IP network failure. If FAILURE=STOP, PSF must be restarted by an operator command. If FAILURE=WCONNECT and the printer is connected to another host when PSF attempts to establish connection, PSF waits for the printer as follows:</p> <ul style="list-style-type: none"> <li>• For SNA, PSF queues a session request in VTAM, which will respond when the printer becomes available. PSF waits up to the time specified by CONNINTV.</li> <li>• For TCP/IP, PSF continuously retries until the printer becomes available. PSF retries up to the time specified by CONNINTV.</li> </ul> <p><b>Syntax:</b> FAILURE=STOP WCONNECT</p>   |

Table 2 (Page 5 of 11). Initialization Parameters for the PRINTDEV Statement: Summary

| Parameter  | Description   |
|--|---|
| FDEFDD<br>(required)                                   | Identifies the DD statement that specifies the form definition libraries.<br><b>Syntax:</b> FDEFDD=*.label  |
| FONTDD<br>(required)                                   | Identifies the DD statement that specifies the default font libraries.<br><b>Syntax:</b> FONTDD=*.label   |
| FONT240<br>(optional)                                  | Identifies the DD statement that specifies the 240-pel font libraries.<br><b>Syntax:</b> FONT240=*.label  |
| FONT300<br>(optional)                                  | Identifies the DD statement that specifies the 300-pel font libraries.<br><b>Syntax:</b> FONT300=*.label  |
| FORMDEF<br>(required)                                  | Specifies a default form definition. For information on how PSF selects a form definition, see “How PSF Selects Resources” on page 115.<br><b>Syntax:</b> FORMDEF=fdefname  |
| IPADDR<br>(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. Specify the IP address in the dotted-decimal format or specify the host name in place of the dotted-decimal address. Enclose the value in single quotation marks.<br><b>SYNTAX:</b> IPADDR='ip-address'<br><b>Examples:</b> IPADDR='9.99.12.33'; IPADDR='jdoe.penn.boulder.ibm.com'  |
| JOBHDR<br>(optional)                                   | Identifies the OUTPUT statement that specifies the form definition and page definition used to format and print job-header separator pages.<br><b>Syntax:</b> JOBHDR=*.label<br><b>Note:</b> Not valid for direct-printing mode.  |
| JOBTRLR<br>(optional)                                  | Identifies the OUTPUT statement that specifies the form definition and page definition used to format and print job-trailer separator pages.<br><b>Syntax:</b> JOBTRLR=*.label<br><b>Note:</b> Not valid for direct-printing mode.  |
| LOGMODE<br>(optional)                                  | Specifies the 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 16 for more information.)<br><br>Although LOGMODE is an optional parameter, it must not be used if the printer is in a cross-domain system and if the PRINTDEV statement specifies MGMTMODE=DIALIN. (For the PRINTDEV statement, see the discussion of the MGMTMODE parameter on page 46.)<br><br>If LOGMODE is used, the name must conform to standard JCL naming conventions. If LOGMODE is omitted or if 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.<br><b>Syntax:</b> LOGMODE=name<br><b>Notes:</b><br><ol style="list-style-type: none"><li>1. Only for SNA-attached printers.</li><li>2. If LOGMODE is omitted or the name is left blank, the name defaults to one of the following:<ul style="list-style-type: none"><li>• The name specified by the DLOGMOD parameter of the related LU statement of the VTAM network definition</li><li>• The name passed by the secondary LU in the CINIT, if that LU resides in another domain</li></ul></li></ol> |

Table 2 (Page 6 of 11). Initialization Parameters for the PRINTDEV Statement: Summary

| Parameter   | Description   |
|---|---|
| LUNAME<br>(required for<br>SNA-attached<br>printers only) | <p>Specifies the unique 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 21.</p> <p><b>Syntax:</b> LUNAME=<i>name</i></p> <p><b>Note:</b> Only for SNA-attached printers (required).</p>  |
| MESSAGE<br>(optional)                                     | <p>Identifies the OUTPUT statement that specifies the form definition and page definition used to format and print messages.</p> <p><b>Syntax:</b> MESSAGE=<i>*.label</i></p> <p><b>Note:</b> For microfilm devices, see Chapter 22, Supporting AFP Output on Microfilm.</p>  |
| MGMTMODE<br>(optional)                                    | <p>Specifies how PSF controls an SNA-attached or TCP/IP-attached printer in deferred-printing mode. MGMTMODE=OUTAVAIL requests that PSF start a communication session with the printer only when output is available on the JES spool. (See Note 2 on page 44).</p> <p>MGMTMODE=DIALIN requests that PSF start a session with the printer when the switched line is connected. LOGAPPL must be applied in the PSF network definition. DIALIN is recommended for some printers that use a switched line. For more information about DIALIN, see “PSF PRINTDEV Statement Considerations for Switched Lines” on page 35. For TCP/IP-attached printers, MGMTMODE=DIALIN is identical with MGMTMODE=IMMED.</p> <p>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.</p> <p>MGMTMODE=IMMED requests that PSF start a communication session with this printer immediately. PSF then looks for output available on the JES spool. (See Note 2 on page 44).</p> <p><b>Syntax:</b> MGMTMODE= <u>OUTAVAIL</u> DIALIN IMMED</p> <p><b>Note:</b> Only for SNA-attached and TCP/IP-attached printers.</p> |

Table 2 (Page 7 of 11). Initialization Parameters for the PRINTDEV Statement: Summary

| Parameter                    | Description  |
|------------------------------|--|
| <p>NORESP<br/>(optional)</p> | <p>Specifies the action that PSF takes when the time specified by the <b>RESPTIME</b> keyword expires before a response is received from the printer.</p> <p><b>Syntax: NORESP=NOTIFY UNBLOCK(<i>node.userid</i>) NOTIFY(OPERATOR) TERMINATE</b></p> <p><b>NORESP=NOTIFY</b><br/>Specifies that when the <b>RESPTIME</b> time expires, PSF is to notify JES (if JES supports this notification) that an expected response was not received from the printer. This condition is also displayed by SDSF.</p> <p><b>NORESP=NOTIFY(<i>node.userid</i>)</b><br/>Specifies that when the <b>RESPTIME</b> time expires, PSF is to send 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.</p> <p><b>NORESP=(OPERATOR)</b><br/>Specifies that when the <b>RESPTIME</b> time expires, PSF is to send a message to the system operator indicating that an expected response was not received from the printer. PSF also notifies JES of the missing response.</p> <p><b>NORESP=TERMINATE</b><br/>Specifies that when the <b>RESPTIME</b> time expires, PSF FSA for the printer is to terminate. The system operator must issue a command to restart the FSA.</p> <p><b>Notes:</b></p> <ol style="list-style-type: none"> <li>1. Only for SNA-attached and TCP/IP-attached printers.</li> <li>2. When <b>NORESP</b> is specified, <b>RESPTIME</b> must also be specified.</li> <li>3. When the FSA is terminated as requested by <b>NORESP=TERMINATE</b> and the FSA is started again, the data set active when termination occurs is restarted from the last checkpoint.</li> </ol> |
| <p>NPRO<br/>(optional)</p>   | <p>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.</p> <p>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.</p> <p>If NPRO is 1, PSF initiates an NPRO.</p> <p><b>Syntax: NPRO=0 1</b></p> <p><b>Notes:</b></p> <ol style="list-style-type: none"> <li>1. This parameter is used only for direct-printing mode with channel-attached printers.</li> <li>2. In deferred-printing mode, the value for NPRO is derived from the JES NPRO definition. The PSF NPRO parameter described above has no effect.</li> <li>3. The only valid values are 0 and 1. Any other value will result in job scheduler messages (IEF), indicating that the numeric exceeds the maximum allowed in the NPRO field and that the job failed with a JCL error.</li> <li>4. In direct-printing mode, a SETPRT macro issued within an application program may cause an NPRO, even though data set processing is not complete.</li> <li>5. For microfilm devices, see Chapter 22, Supporting AFP Output on Microfilm.</li> </ol>   |

Table 2 (Page 8 of 11). Initialization Parameters for the PRINTDEV Statement: Summary

| Parameter   | Description   |
|---|---|
| OBJCONDD<br>(required with<br>COMSETUP and<br>COLORMAP) | <p>Identifies the DD statement that specifies the object container library. The object container library contains:</p> <ul style="list-style-type: none"> <li>• Microfilm setup resources specified in the COMSETUP parameter for printing on a microfilm device. For more information, see Chapter 22, Supporting AFP Output on Microfilm.</li> <li>• Color mapping table resources specified in the COMSETUP parameter for printing to color printers that support color mapping. For more information, see: <ul style="list-style-type: none"> <li>– Chapter 4, “Working with Libraries” on page 37.</li> <li>– <i>InfoPrint Hi-Lite Color Application Design Reference</i>, available through your IBM marketing representative.</li> <li>– <i>Mixed Object Document Content Architecture Reference</i>, SC31-6802-03 or later.</li> <li>– <i>InfoPrint 4000 HC Introduction and Planning Guide</i>, G544-5420</li> </ul> </li> </ul> <p><b>Syntax:</b> <b>OBJCONDD</b>=*.label</p> |
| OVERLAY<br>(optional)                                   | <p>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 1 to 8 alphanumeric or national characters; the first character must be alphabetic or national.</p> <p><b>SYNTAX:</b> <b>OVERLAY</b>=overlayname</p> <p><b>Example:</b> OVERLAY=01GLOBAL</p> <p><b>Notes:</b></p> <ol style="list-style-type: none"> <li>1. PSF does not prefix the name given with O1.</li> <li>2. This overlay is always loaded from the system library. If multiple system overlay libraries are defined for different resolutions, the overlay should be placed in each library concatenation list.</li> </ol>  |
| OVLYDD<br>(optional)                                    | <p>Identifies the DD statement that specifies the default overlay libraries.</p> <p><b>Syntax:</b> <b>OVLYDD</b>=*.label</p>  |
| OVLY240<br>(optional)                                   | <p>Identifies the DD statement that specifies the 240-pel overlay libraries.</p> <p><b>Syntax:</b> <b>OVLY240</b>=*.label</p>   |
| OVLY300<br>(optional)                                   | <p>Identifies the DD statement that specifies the 300-pel overlay libraries.</p> <p><b>Syntax:</b> <b>OVLYDD</b>=*.label</p>  |
| PAGEDEF<br>(required)                                   | <p>Specifies the default page definition. (This page definition is also used if none is specified in JOBHDR, JOBTRLR, DSHDR, and MESSAGE parameters.) For information on how PSF selects a page definition, see “How PSF Selects Resources” on page 115.</p> <p><b>Syntax:</b> <b>PAGEDEF</b>=pdefname</p>  |
| PDEFDD<br>(required)                                    | <p>Identifies the DD statement that specifies the page definition libraries.</p> <p><b>Syntax:</b> <b>PDEFDD</b>=*.label</p>  |

Table 2 (Page 9 of 11). Initialization Parameters for the PRINTDEV Statement: Summary

| Parameter             | Description  |
|-----------------------|--|
| PIMSG<br>(optional)   | <p>Controls the handling of PSF messages.</p> <p>Y or YES specifies that all message groups 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 generated up to this error are printed, including the message group describing the terminating error. YES is the default.</p> <p>N or NO specifies that no message groups are to be printed unless an error occurs that forces the premature termination of the printing of the data set. If that happens, the message group describing the terminating error is printed.</p> <p><b>Note:</b> Reaching the message count value specified for PIMSG is one error that forces a premature termination of printing. If you do not code a message count value, PSF uses the default value of 16.</p> <p><i>nnn</i> specifies that after PSF has generated <i>nnn</i> 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 may be more than <i>nnn</i> message groups if the message groups are generated for errors reported by the printer for pages sent before the message count is reached. A value of 0 allows data set printing to continue, regardless of the number of message groups generated, unless a terminating error occurs. The default is 16.</p> <p><b>Note:</b> Messages are printed depending on whether YES or NO is specified in this parameter. For more information on PIMSG, refer to <i>PSF for OS/390: User's Guide</i>.</p> <p><b>Syntax:</b> <b>PIMSG=(YES</b> [<i>,nnn</i>] <b> NO</b> [<i>,nnn</i>])</p> <p><b>Note:</b> The PIMSG specified on the OUTPUT statement in the Writer-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 113.</p> |
| PORTNO<br>(optional)  | <p>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. The default port is 5001.</p> <p><b>Syntax:</b> <b>PORTNO= 5001 nnnnn</b>, where <i>nnnnn</i> is a value between 5001 to 32767.</p> <p><b>Note:</b> For TCP/IP-attached printers only.</p>   |
| PRERROR<br>(optional) | <p>Controls whether the PRERROR keyword is honored on the OUTPUT JCL statement.</p> <p><b>Syntax:</b> <b>PRERROR=HONOR NOTHONOR</b></p> <p><b>PRERROR=HONOR</b> specifies that PSF honors PRERROR if specified on the OUTPUT JCL statement for data sets printed on this printer.</p> <p><b>PRERROR=NOTHONOR</b> specifies that PSF ignores PRERROR if specified on the OUTPUT JCL statement for data sets printed on this printer. This is the default.</p>   |
| PSEGDD<br>(optional)  | <p>Identifies the DD statement that specifies the default page segment libraries. In addition to page segments, the page segment library might contain BCOCA, GOCA, and IOCA resources.</p> <p><b>Syntax:</b> <b>PSEGDD=*.label</b></p>  |
| PSEG240<br>(optional) | <p>Identifies the DD statement that specifies the 240-pel page segment libraries.</p> <p><b>Syntax:</b> <b>PSEG240=*.label</b></p>   |
| PSEG300<br>(optional) | <p>Identifies the DD statement that specifies the 300-pel page segment libraries.</p> <p><b>Syntax:</b> <b>PSEG300=*.label</b></p>   |

Table 2 (Page 10 of 11). Initialization Parameters for the PRINTDEV Statement: Summary

| Parameter              | Description  |
|------------------------|--|
| RELINTV<br>(optional)  | <p>Specifies the release interval, in seconds. This is the time period after which a request to release the printer is honored. The timer for the release interval is not started until PSF is at the next job boundary.</p> <p><b>Syntax:</b> RELINTV=<i>nn</i>, where <i>nnn</i> is a value from 0 to 86400.</p>   |
| RELMODE<br>(optional)  | <p>Specifies the manner in which 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 51.</p> <p><b>Syntax:</b> RELMODE=IDLE TIME NONE</p> <p><b>IDLE</b> specifies to release the printer when (1) a request to release has been received and (2) there is no output on the spool for this printer for the time specified by the release interval.</p> <p><b>TIME</b> specifies to begin the release interval when a request to release has been received, even when there is more output on the spool.</p> <p><b>NONE</b> specifies that the printer is not to be released. This is the default.</p>   |
| RESPTIME<br>(optional) | <p>Specifies the maximum number of seconds PSF should wait for a response from the printer. If RESPTIME is not specified or RESPTIME=0 is specified, PSF waits indefinitely for a response.</p> <p><b>Syntax:</b> RESPTIME= <u>0</u> <i>nnnnn</i>, where <i>nnnnn</i> is a value from 0 to 86400. The default is 0.</p> <p><b>Notes:</b></p> <ol style="list-style-type: none"> <li>1. Only for SNA and TCP/IP-attached printers.</li> <li>2. The RESPTIME value is only effective with TCP/IP once the TCP/IP session has been established (CONNECT, ASSIGN, and START SESSION have completed successfully).</li> <li>3. The RESPTIME value is only effective with SNA when PSF is actually sending IPDS data (this does not include items such as BIND, ATTACH, DETACH, and UNBIND).</li> <li>4. When RESPTIME is specified, NORESP must also be specified to tell PSF what action to take when no response is received within the time specified.</li> <li>5. RESPTIME might not be appropriate for faster printers. This function adds some overhead to start and stop the timers.</li> <li>6. If the printer has an intervention timer, consider how that timer interacts with this one when choosing the time values for both timers.</li> </ol> |
| SDEFDD<br>(optional)   | Refer to <i>Print Services Facility: Security Guide</i> for a description of this parameter.   |
| SETUP<br>(optional)    | <p>Specifies the setup parameters for which JES should issue 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.</p> <p><b>Syntax:</b> SETUP=FORMS BURST (FORMS,BURST)</p> <p><b>Note:</b> Only for SNA-attached or TCP/IP-attached printers. Setup messages are always issued for local channel-attached printers.</p>  |
| SFONTDD<br>(optional)  | Refer to <i>Print Services Facility: Security Guide</i> for a description of this parameter.   |
| SOVLYDD<br>(optional)  | Refer to <i>Print Services Facility: Security Guide</i> for a description of this parameter.   |
| SPAGELBL<br>(optional) | Refer to <i>Print Services Facility: Security Guide</i> for a description of this parameter.   |
| SPSEGDD<br>(optional)  | Refer to <i>Print Services Facility: Security Guide</i> for a description of this parameter.   |



Table 2 (Page 11 of 11). Initialization Parameters for the PRINTDEV Statement: Summary

| Parameter          | Description   |
|--------------------|---|
| SYSAREA (optional) | Refer to <i>Print Services Facility: Security Guide</i> for a description of this parameter.  |
| TIMEOUT (optional) | <p>Specifies the action that PSF is to take after a timeout when no output is available from JES. The DISCINTV parameter (described on page 44) specifies the timeout interval.</p> <p>TIMEOUT=REDRIVE requests that PSF redrive the printer FSA according to specifications in the MGMTMODE parameter (described on page 46).</p> <p>TIMEOUT=STOP requests that PSF stop the printer FSA, which can then be restarted only by an operator command.</p> <p><b>Syntax:</b> TIMEOUT= REDRIVE STOP</p> <p><b>Note:</b> Only for SNA-attached and TCP/IP-attached printers.</p> |
| TRACE (optional)   | <p>Specifies PSF tracing. For more information on using traces, refer to <i>PSF for OS/390: Diagnosis</i>. TRACE=YES specifies PSF tracing. TRACE=NO specifies no PSF tracing.</p> <p><b>Syntax:</b> TRACE=YES NO</p>   |

## Selecting PSF Parameters for a Dedicated Network-Attached Printer

When your printer is attached to a single PSF on an SNA or TCP/IP network, specify the following PRINTDEV parameters:

- MGMTMODE=IMMED
- CONNINTV=0 (Unless you want to detect when session startup is delayed excessively.)
- DISCINTV=0
- 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 RELREQ
- Port switching

PSF releases a printer only at the end of a job, so that output is never split.

### JES Spool-Driven

This method is valid for SNA-attached or TCP/IP-attached printers. A session with the printer is maintained while there is output on the JES spool and the printer is available. When there is no more output on the spool, and the disconnect interval expires, PSF ends the session with the printer. When there is more work on the spool for this printer, PSF attempts to restart the session. After the session is restarted, PSF must reload the resources required for following print jobs. With this method, there is no means for PSF to determine whether another print driver is active.

This method is requested by specifying the following PRINTDEV parameters:

- MGMTMODE=OUTAVAIL
- DISCINTV=*nn*
- TIMEOUT=REDRIVE
- FAILURE=WCONNECT
- RELMODE=NONE

## VTAM RELREQ

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 has requested a session with the printer. The session with the printer is then ended according to the RELMODE specified. After the session is restarted, PSF must reload any resources required for following print jobs.

If RELMODE=IDLE, PSF ends the session when a VTAM request to release has been received and there is no output on the spool for the printer for the release interval specified. When there is more work on the spool for this printer, PSF attempts to restart the session.

This method is requested by specifying the following PRINTDEV parameters:

- MGMTMODE=OUTAVAIL
- RELMODE=IDLE
- RELINTV=*nn*

If RELMODE=TIME, when a VTAM request to release is received PSF waits for the time 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 MGMTMODE specified.

This method is requested by specifying the following PRINTDEV parameters:

- RELMODE=TIME
- RELINTV=*nn*
- ACQINTV=*nn*

## Port Switching

This method is valid for printers that support multiple active ports and IPDS dialogs, such as the 3130 printer with release 2 microcode. This is the most efficient method 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 has requested control, PSF ends its dialog in accordance with the RELMODE specified.

**Note:** Some printers, such as the IBM Network Printer Models 12, 17, and 24 support multiple ports but do not use IPDS dialogs. These printers may do port switching without informing PSF. For these printers, RELMODE and RELINTV have no effect.

If RELMODE=IDLE, PSF ends the dialog when an IPDS request to release has been received and there is no output on the spool for the printer for the time specified by the release interval. PSF attempts to restart the dialog when there is more work on the spool for this printer.

This method is requested by specifying the following PRINTDEV parameters:

- RELMODE=IDLE

- RELINTV=*nn*

If RELMODE=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 following PRINTDEV parameters:

- RELMODE=TIME
- RELINTV=*nn*
- ACQINTV=*nn*

**Note:** When a release or acquire interval of 0 is specified, the related action is taken immediately.

---

## Sharing ESCON Channel-Attached Printers

An ESCON channel-attached printer can be shared among multiple PSF hosts, but it can only be in use by one PSF at a time. This sharing is manually controlled by the operator using the MVS VARY operator command. If used in conjunction with the printer multi-host support, the printer allows only one PSF to use the printer at a time.

Some ESCON channel-attached printers have the multi-host support that, when enabled, enforces the use of the printer by only one PSF at a time. Enabling the multi-host support, requires a successful printer ASSIGN before PSF can use the printer.

Enable the multi-host support only if systems connected to the printer have OS/390 1.3.0 or higher installed; otherwise, do not enable the support. The OS/390 systems can be guests of VM/ESA. If these conditions are met and multi-host is enabled, the printer can be used by only one PSF at a time.

**Attention:** If this support is enabled and one of the hosts does not have the correct software installed, PSF will not be able to use the printer. For more information on the multi-host support, see the Introduction and Planning Guide for the appropriate printer.

Exclusive access to the printer can also be given at IPL time. By enabling the multi-host support and configuring the printer to come online at IPL time, a printer ASSIGN is attempted during the IPL. If multiple systems, which share a printer with multi-host support enabled and defined to come online at IPL time, are IPLed concurrently, the first system to successfully issue the printer ASSIGN gets access to the printer.

When the printer is configured in multi-host mode, it's important to follow proper operational procedures, such as draining PSF and varying the printer offline before re-IMLing the printer. If not, the printer ASSIGN could be lost, the printer boxed, and PSF could probably issue an Abend024 RC617. For more information on reconfiguration and recovery operational procedures for a printer ASSIGN, such as ASSIGN lost conditions and device boxing, refer to the *OS/390 MVS Recovery and Reconfiguration Guide*.

---

## Recommended Values for PSF Timers

When a PSF printer is shared with other printer drivers, the desired values for the PSF timers vary depending on the:

- Job mix expected through the printer drivers
- Share capabilities of the other printer drivers
- Priorities 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 has finished printing or after a designated period of time.

As a starting point, use the following values when these timers are appropriate:

- CONNINTV=300 (specified in most cases)
- DISCINTV=240
- RELINTV=240
- ACQINTV=30

---

## Chapter 7. 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. PSF-attached printers, however, 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.

Under JES2, a PSF FSA can start in either of two ways:

- An operator can enter the START DEVICE JES2 command.
- When JES2 is started, PSF starts automatically if the PRTnnnn statement for the printer FSA contains the START parameter.

Under JES3, a PSF FSA can start in either of two ways:

- An operator issues the CALL WRITER JES3 command.
- When printing is scheduled for a PSF printer FSA, 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.

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.

**Note:** Stop the printer before you enter JES repositioning commands.

To use a PSF printer as a system output device for deferred printing, you must first do the following:

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.”
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 67.

**Note:** For configurations that include PSF attached to the Distributed Print Function (DPF) of PSF/2, JES deletes the print data set from the host spool after PSF has transmitted a data set to the DPF spool. The host operator cannot control the print data set after it has been 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 is defined by a PRTnnnn statement. For JES3, the FSS is defined by an FSSDEF statement, and each printer FSA is defined by a DEVICE statement. For information on defining the FSS and printer FSA to JES, see “Sample Initialization Statements for JES2” on page 56 and “Sample Initialization Statements for JES3” on page 61.

The following number of FSSs and FSAs are supported:

- A maximum of 2000 FSSs
- A maximum of 64 FSAs per FSS (see note)

**Note:** The actual number of FSAs per FSS depends on several factors, some of which are as follows:

- The virtual region size below the 16-megabyte line
- The number and the size of form definitions and page definitions
- The number of buffers specified (BUFNO) for channel-attached printers
- The block sizes of the font, page segment, overlay, form definition, and page definition libraries

For more information, see “Selecting Region Sizes” on page 112.

---

## Sample Initialization Statements for JES2

Figure 19 shows sample JES2 statements for a channel-attached printer (PRT1), an SNA-attached printer (PRT2), and a TCP/IP-attached printer (PRT3).

**Note:** These are samples only. For specific coding for JES parameters, refer to the JES publication for the level of JES you are using.

```
FSS(FSS1) PROC=SAMPPROC,HASPFSSM=HASPFSSM
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
```

*Figure 19. Sample JES2 Initialization Statements for Three Printers. These statements are commonly found in SYS1.PARMLIB (initialization deck member name).*

## Parameters for the JES2 Initialization Statement

The following sections describe the JES2 initialization statements shown in Figure 19, which are typical of the JES2 statements used with PSF. For more detailed descriptions of all JES2 initialization statements, refer to the JES2 initialization and tuning publication for your operating system.

The descriptions below tell which statements and parameters are required. In the programming syntax 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. The FSS initialization statement defines an FSS to JES2. The FSS statement is associated with one or more `PRTnnnn` statements that define each printer FSA. For a description of all FSS parameters, refer to the JES2 initialization and tuning publication for your operating system. The FSS statement contains the following parameters for PSF:

### FSS(*fssname*)

Specifies the unique 1- to 8-character name of a particular FSS. This name is referenced in the `PRTnnnn` 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 either `SYS1.PROCLIB` or a library concatenated to `SYS1.PROCLIB`. For more information, see “PSF Startup Procedures for AFP Printers” on page 67.

### HASPFSSM

Specifies the 1- to 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`).

## PRTnnnn Statement

A `PRTnnnn` statement, which is required to define each printer FSA, is associated with an FSS statement. Each printer FSA should have a unique `PRTnnnn` 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 Figure 22 on page 69.

**Note:** The label on the `PRINTDEV` statement must be 8 characters or fewer. PSF accepts the `PRTnnnn`, `PRINTnn`, and `PRINTERn` formats. `PRTnnnn` is the recommended format.

The `PRTnnnn` parameters create JES2 default values that are used unless other values are specified in the application program JCL. The `PRTnnnn` 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 or 3900 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. With a 3900 printer, however, 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 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 printing to microfilm output, specify CKPTMODE=SEC. For additional information about printing to microfilm, see Chapter 22, Supporting AFP Output on Microfilm.

#### **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 has to 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 22, Supporting AFP Output on Microfilm.

#### **CKPTSEC**

Specifies the seconds between checkpoints in the data set. When printing to microfilm output, specify a CKPTSEC keyword instead of CKPTPAGE. For additional information about printing to microfilm, see Chapter 22, Supporting AFP Output on Microfilm.

#### **CLASS**

Specifies the output classes processed by the printer. You may want to consider the handling of disabled mechanisms by defining print classes.

#### **COPYMARK=DATASET|JOB|CONSTANT**

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.

For more information on copy marking, see “Copy Marking and Offset Stacking” on page 84.

#### **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 $nnn$  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 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 81 and “Carrier-Strip Marking” on page 83.

**Note:** This parameter is new in JES2; it replaces the MARK|NOMARK parameter.

## MODE

Specifies that the printer is managed by an FSS (MODE=FSS). This parameter is required.

**Note:** For a 3800 printer 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 22, Supporting AFP Output on Microfilm.

If NPRO is non-zero when the printer is purged and the PSF FSA terminates, PSF causes the printer to do an NPRO immediately without waiting for the NPRO time to expire. This ensures that all data has reached 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 purged and the PSF FSA terminates. 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 terminates, data might be lost. Even if you do not normally want NPROS, you may want to set a large NPRO value to ensure that all data is complete when PSF terminates. This is recommended for duplex continuous-forms printers. The maximum NPRO value allowed for JES2 is 3600 seconds (1 hour).

## 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.

**SOSIn** Specifies a shift-out/shift-in processing mode of SOSI1, SOSI2, or SOSI3 for double-byte fonts.

PRMODE can be assigned to a print job on the JCL OUTPUT statement. 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, or SOSI3. If your installation uses double-byte fonts, specify the appropriate SOSIn 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 have defined in your installation to control job processing must be specified for printers. For more information about the PRMODE parameter, refer to *PSF for OS/390: User's Guide*.

#### **PRESELECT=YES|NO**

Specifies whether jobs are preselected for printers.

**YES** Specifies that two output elements or jobs are to be selected for each printer. This 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 the Distributed Print Function (DPF) of PSF/2 is installed, you may need to specify SEP=YES. For more information, see "Distributed Print Function (DPF) Considerations" on page 80.

For microfilm devices, you should 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, you should specify the SEPDS=NO parameter.

#### **START=YES|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|NO**

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. A table showing how fonts are selected in PSF is given in "How PSF Selects Special Processing Parameters" on page 111.

**Note:** To select a font in the startup procedure instead of the JES2 default, code UCS=0 on the PRTnnnn 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 has been offline during the initialization of JES. This parameter applies only to host-connected, channel-attached non-SNA printers.

---

## **Sample Initialization Statements for JES3**

Figure 20 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).

### **For a host-connected, channel-attached, non-SNA printer:**

```
FSSDEF,TYPE=WTR,FSSNAME=FSS1,PNAME=SAMPPR01
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,60DB),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=SAMPPR02,SYSTEM=SYS1,TERM=NO
DEVICE,DTYPE=PRTAFP1,JNAME=PRT2,JUNIT=(,SYS1,,OFF),FSSNAME=FSS2,
      MODE=FSS,PM=(LINE,PAGE,SOSI1),CHARS=(YES,60DB),
      CARRIAGE=(YES,A868),CKPNTPG=100,HEADER=YES,WC=(B)
```

### **For a TCP/IP-attached printer:**

```
FSSDEF,TYPE=WTR,FSSNAME=FSS3,PNAME=SAMPPR03,SYSTEM=SYS1,TERM=NO
DEVICE,DTYPE=PRTAFP1,JNAME=PRT3,JUNIT=(,SYS1,,OFF),FSSNAME=FSS3,
      MODE=FSS,PM=(LINE,PAGE,SOSI1),CHARS=(YES,60DB),
      CARRIAGE=(YES,A868),CKPNTPG=100,HEADER=YES,WC=(C)
```

Figure 20. Sample JES3 Initialization Statements for Three Printers. These statements are commonly found in SYS1.PARMLIB (initialization deck member name).

Figure 21 on page 62 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 21. Sample Initialization Statements for JES3. These statements are used for running multiple FSAs under a single FSS.

## Parameters for the JES3 Initialization Statement

The following sections describe the JES3 initialization statements shown in Figure 20 on page 61, which are typical of the JES3 statements used with PSF. For more detailed descriptions of all the JES3 initialization statements, refer to the JES3 initialization and tuning publication for your operating system.

The following descriptions show which statements and parameters are required. In the programming syntax 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 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 being invoked.

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 shown in Figure 20 on page 61, `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 Figure 23 on page 71.

## SYSTEM

Specifies the JES3 processor on which the FSS is to run. The name must be the same as that specified on the NAME parameter of the MAINPROC statement for the processor.

## TERM=YES|NO

YES specifies that the FSS is to be terminated if the JES3 global address space terminates 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 or not 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 the description of the STACKER parameter on page 66.
2. If DPF is installed, you must specify BURST=(YES). For more information, see "Distributed Print Function (DPF) Considerations" on page 80.

## 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 to 4 characters) of the page definition is to be used as a default. The name specified here, or the JES3 system default, is the PSF default page definition. For information on selecting PSF defaults, see "How PSF Selects Special Processing Parameters" on page 111.

## CHARS

Specifies a default JES3 font. For a chart showing how fonts are selected in PSF, see "How PSF Selects Special Processing Parameters" on page 111.

## CKPNTPG

Specifies the number of pages between data set checkpoints. Specify no more frequent checkpoints than you actually need in order 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 will need 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 22, Supporting AFP Output on Microfilm.

### **CKPNTSEC**

Specifies the seconds between data set checkpoints.

**Note:** For printing to microfilm output, specify the CKPNTSEC parameter. For additional information about printing to microfilm, see Chapter 22, Supporting AFP Output on Microfilm.

### **COPYMARK=C|N|J**

Specifies that you want vertical separator lines printed on the edges of the output. 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 on 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. 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 on a job basis. A change in the thickness of the copy mark indicates the beginning of a new job.

### **DTYPE**

Identifies the printer device type, as follows:

- 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 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, refer to the *JES3 Installation and Tuning* publication for your operating system, listed in the “Bibliography” on page 389.

### **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 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 if job and data set header pages are printed.

For microfilm devices, you should specify HEADER=NO.

**Note:** If DPF is installed, you may need to specify HEADER=YES. For more information, see “Distributed Print Function (DPF) Considerations” on page 80.

## JNAME

Specifies the name of the printer FSA. Each printer FSA should 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 Figure 23 on page 71. 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 22, Supporting AFP Output on Microfilm.

If NPRO is non-zero when the printer is purged and the PSF FSA terminates, PSF causes the printer to do an NPRO immediately without waiting for the NPRO time to expire. This ensures that all data has reached 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 purged and the PSF FSA terminates. 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 terminates, data might be lost. Even if you do not normally

want NPROS, you may want to set a large NPRO value to ensure that all data is complete when PSF terminates. This is recommended for duplex continuous-forms printers. The maximum NPRO value allowed for JES3 is 9999 seconds (2 hours 46 minutes).

**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 is 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.

**FCB**

Specifies that the JES3 default for carriage (FCB or CARR) is ignored and the PRINTDEV value for PAGEDEF is used.

**CHARS,FCB**

Specifies that the JES3 default for characters (UCS or train) and carriage (FCB or CARR) are ignored and the PRINTDEV value for CHARS and PAGEDEF are used.

**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, or PM=SOSI3 specifies a printer that can process double-byte fonts. For more information about SOSI processing, refer to *PSF for OS/390: 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 may want to consider handling disabled mechanisms by defining print classes. For more information, see "Disabled Mechanisms" on page 86.

**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 starting a PSF FSS, you must have a cataloged startup procedure in SYS1.PROCLIB or any other system procedure library. 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, chosen for the defined printer.

Several PSF startup procedures are supplied with PSF. You can modify the startup procedures supplied with PSF or write your own startup. For sample JES2 and JES3 PSF startup procedures, see “Sample PSF Startup Procedures.” Publications on the initialization and tuning of JES are listed in the “Bibliography” on page 389; these publications contain details about procedure libraries.

When you install PSF, you must create or update the required libraries referenced by the startup procedure, and you should 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 on JES defaults, see “Defining PSF FSSs and FSAs for Printers” on page 55 and “How PSF Selects Special Processing Parameters” on page 111.

## Sample PSF Startup Procedures

Figure 22 on page 69 shows a sample startup procedure for three FSSs, each with a single printer FSA. 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.

The JES2 initialization for this FSS is shown in Figure 19 on page 56. 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.

Figure 23 on page 71 shows a sample startup procedure for two FSSs, each with a single printer FSA. PRT1 FSA is a channel-attached printer. PRT2 is an SNA-attached printer.

The JES3 initialization statements for these FSSs and printer FSAs are shown in Figure 20 on page 61. 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 20 on page 61, the FSSDEF statement 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 following are examples showing how startup procedures could be modified, and do not necessarily use the recommended values. For recommended values, refer to SYS1.PROCLIB.

```

//SAMPPROC PROC
/* THE FOLLOWING STARTUP PROCEDURE DEFINES THE JCL AND PRINTDEV
/* STATEMENTS FOR THREE DIFFERENT PSF-SUPPORTED PRINTERS.
/*
//STEP01 EXEC PGM=APSPPIEP,REGION=4096K,TIME=1440,PARM=(,,,TCPIP)
//JOBHDR OUTPUT PAGEDEF=V06483, /* JOB HEADER PAGE */
// FORMDEF=A10120,CHARS=60DB /* FORMDEF: ALTERNATIVE BIN*/
//JOBTLR OUTPUT PAGEDEF=V06483, /* JOB TRAILER PAGE */
// FORMDEF=A10110,CHARS=60DB /* FORMDEF: MAIN BIN */
//DSHDR OUTPUT PAGEDEF=V06483, /* DATA SET SEPARATOR */
// FORMDEF=A10110,CHARS=60DB /* FORMDEF: MAIN BIN */
//MSGDS OUTPUT PAGEDEF=A08682, /* MESSAGE DATA SET */
// FORMDEF=A10110,CHARS=60DB /*
/*
//FONT01 DD DSN=SYS1.FONTLIB,DISP=SHR /* SYSTEM 240 PEL FONTS */
// DD DSN=SYS1.FONTOLN,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.FONTOLN,DISP=SHR /* SYSTEM OUTLINE FONTS */
// DD DSN=INST.FONTLIB,DISP=SHR /* INSTALLATION USER FONTS */
/*
//PSEG01 DD DSN=INST.PSEGLIB,DISP=SHR /* INSTALLATION PAGE SEGMENTS*/
/* FORMATTED FOR 240 PEL
/* PRINTERS
// DD DSN=SPEC.PSEGLIB,DISP=SHR /* SPECIAL PAGE SEGMENTS */
/*
//PSEG02 DD DSN=INST.PSEGLIB,DISP=SHR /* INSTALLATION PAGE SEGMENTS*/
/* FORMATTED FOR 300 PEL
/* PRINTERS
/*
//OLAY01 DD DSN=INST.OVERLIB,DISP=SHR /* INSTALLATION OVERLAYS */
/* FORMATTED FOR 240 PEL
/* PRINTERS
/*
//OLAY02 DD DSN=INST.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 */
/*
//PRT1 CNTL
//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 */
// 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 */
// PAGEDEF=A08682, /* DEVICE PAGEDEF DEFAULT */
// FORMDEF=A10110, /* DEVICE FORMDEF DEFAULT */
// CHARS=60DB, /* DEVICE DEFAULT FONT */
// PIMSG=(YES,16), /* ACCUMULATE DATA SET MESSAGES */

```

Figure 22 (Part 1 of 2). Sample PSF Startup Procedure for Three Printers, Showing Multiple FSAs

```

//      DATAACK=UNBLOCK,          /* UNBLOCK DATA CHECKS      */
//      TRACE=YES,                 /* BUILD INTERNAL TRACE     */
//      SETUP=FORMS                /* SETUP MESSAGE            */
//PRT1  ENDCNTL
//*
//PRT2  CNTL
//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     */
//      FORMDEF=A10110,          /* DEVICE FORMDEF DEFAULT     */
//      CHARS=60DB,              /* DEVICE DEFAULT FONT        */
//      PIMSG=(YES,16),           /* ACCUMULATE DATA SET MESSAGES */
//      DATAACK=BLOCK,          /* BLOCK DATA CHECKS         */
//      TRACE=YES,               /* BUILD INTERNAL TRACE       */
//      FAILURE=WCONNECT,        /* VTAM: ATTEMPT RECONNECT    */
//      DISCINTV=0,              /* VTAM: NO TIME OUT         */
//      MGMTMODE=IMMED,          /* VTAM: MAINTAIN SESSION     */
//      APPLID=name,             /* VTAM: APPLICATION PGM NODE */
//      LUNAME=name              /* VTAM: LOGICAL UNIT NODE    */
//PRT2  ENDCNTL
//*
//PRT3  CNTL
//PRT3  PRINTDEV FONTDD=*.FONT01, /* 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       */
//      JOBHDR=*.JOBHDR,         /* JOB HEADER SEPARATOR OUTPUT */
//      JOBTRLR=*.JOBTLR,        /* JOB TRAILER SEPARATOR OUTPUT */
//      MESSAGE=*.MSGDS,         /* MESSAGE DATA SET OUTPUT    */
//      PAGEDEF=A08682,          /* DEVICE PAGEDEF DEFAULT     */
//      FORMDEF=A10110,          /* DEVICE FORMDEF DEFAULT     */
//      CHARS=60DB,              /* DEVICE DEFAULT FONT        */
//      PIMSG=(YES,16),           /* ACCUMULATE DATA SET MESSAGES */
//      DATAACK=BLOCK,          /* BLOCK DATA CHECKS         */
//      TRACE=YES,               /* BUILD INTERNAL TRACE       */
//      FAILURE=WCONNECT,        /* ATTEMPT RECONNECT         */
//      DISCINTV=0,              /* NO TIME OUT                */
//      MGMTMODE=IMMED,          /* MAINTAIN SESSION           */
//      IPADDR='9.99.12.33'      /* IP ADDRESS OF PRINTER      */
//PRT3  ENDCNTL

```

Figure 22 (Part 2 of 2). Sample PSF Startup Procedure for Three Printers, Showing Multiple FSAs

**For a Channel-Attached Printer:**

```
//          MESSAGE= *.MSGDS,          /*MESSAGE DATA SET OUTPUT      */
//SAMP01 PROC
/** THE FOLLOWING STARTUP PROCEDURE DEFINES THE JCL AND PRINTDEV
/** STATEMENTS FOR PRT1, A CHANNEL-ATTACHED PRINTER.
/**
//STEP01 EXEC PGM=APSPPIEP,REGION=4096K,PARM=(,INTR)
//JOBHDR OUTPUT PAGEDEF=V06483,        /* JOB HEADER PAGE          */
//          FORMDEF=A10120,CHARS=60DB   /* FORMDEF: ALTERNATIVE BIN*/
//JOBTLR OUTPUT PAGEDEF=V06483,        /* JOB TRAILER PAGE        */
//          FORMDEF=A10110,CHARS=60DB   /* FORMDEF: MAIN BIN      */
//DSHDR  OUTPUT PAGEDEF=V06483,        /* DATA SET SEPARATOR     */
//          FORMDEF=A10110,CHARS=60DB   /* FORMDEF: MAIN BIN      */
//MSGDS  OUTPUT PAGEDEF=A08682,        /* MESSAGE DATA SET       */
//          FORMDEF=A10110,CHARS=60DB   /*                          */
/**
//FONT01 DD DSN=SYS1.FONTLIB,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  */
/**
//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      */
//          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 */
//          PAGEDEF=A08682,        /* DEVICE PAGEDEF DEFAULT */
//          FORMDEF=A10110,        /* DEVICE FORMDEF DEFAULT */
//          CHARS=60DB,           /* DEVICE DEFAULT FONT */
//          PIMSG=(YES,16),        /* ACCUMULATE DATA SET MESSAGES */
//          DATAK=UNBLOCK,        /* UNBLOCK DATA CHECKS */
//          TRACE=YES,            /* BUILD INTERNAL TRACE */
//          SETUP=FORMS           /* SETUP MESSAGE */
/** DEFAULT PAGEDEF AND CHARS ARE SPECIFIED BY JES3
/**          PAGEDEF=A868          /* COPY OF A08682 */
/**          CHARS=60DB          /* DEVICE DEFAULT FONT */
//PRT1     ENDCNTL
```

Figure 23 (Part 1 of 3). Sample PSF Startup Procedures for Three Separate Printers

**For an SNA-Attached Printer:**

```
//SAMP02 PROC
/* THE FOLLOWING STARTUP PROCEDURE DEFINES THE JCL AND PRINTDEV
/* STATEMENTS FOR PRT2, AN SNA-ATTACHED PRINTER.
/*
//STEP01 EXEC PGM=APSPPIEP,REGION=4096K,TIME=1440,PARM=(,INTR)
//JOBHDR OUTPUT PAGEDEF=V06483, /* JOB HEADER PAGE */
// FORMDEF=A10120,CHARS=60DB /* FORMDEF: ALTERNATIVE BIN*/
//JOBTLR OUTPUT PAGEDEF=V06483, /* JOB TRAILER PAGE */
// FORMDEF=A10110,CHARS=60DB /* FORMDEF: MAIN BIN */
//DSHDR OUTPUT PAGEDEF=V06483, /* DATA SET SEPARATOR */
// FORMDEF=A10110,CHARS=60DB /* FORMDEF: MAIN BIN */
//MSGDS OUTPUT PAGEDEF=A08682, /* MESSAGE DATA SET */
// FORMDEF=A10110,CHARS=60DB /*
/*
//FONT01 DD DSN=SYS1.FONTLIB,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 */
// FORMDEF=A10110, /* DEVICE FORMDEF DEFAULT */
// PIMSG=(YES,16), /* ACCUMULATE DATA SET MESSAGES */
// DATAACK=BLOCK, /* BLOCK DATA CHECKS */
// TRACE=YES, /* BUILD INTERNAL TRACE */
// FAILURE=WCONNECT, /* VTAM: ATTEMPT RECONNECT */
// DISCINTV=0, /* VTAM: NO TIME OUT */
// MGMTMODE=IMMED, /* VTAM: MAINTAIN SESSION */
// APPLID=name, /* VTAM: APPLICATION PGM NODE */
// LUNAME=name /* VTAM: LOGICAL UNIT NODE */
/* DEFAULT PAGEDEF AND CHARS ARE SPECIFIED BY JES3 */
/* PAGEDEF=A868 /* COPY OF A08682 */
/* CHARS=60DB /* JES3 DEFAULT FONT */
//PRT2 ENDCNTL
```

Figure 23 (Part 2 of 3). Sample PSF Startup Procedures for Three Separate Printers

**For a TCP/IP-attached printer:**

```
//SAMP3 PROC
/* THE FOLLOWING STARTUP PROCEDURE DEFINES THE JCL AND PRINTDEV
/* STATEMENTS FOR PRT3, A TCP/IP-ATTACHED PRINTER.
/*
//STEP01 EXEC PGM=APSPPIEP,REGION=4096K,TIME=1440,PARM=(,,,TCPIP)
//JOBHDR OUTPUT PAGEDEF=V06483, /* JOB HEADER PAGE */
// FORMDEF=A10120,CHARS=60DB /* FORMDEF: ALTERNATIVE BIN*/
//JOBTLR OUTPUT PAGEDEF=V06483, /* JOB TRAILER PAGE */
// FORMDEF=A10110,CHARS=60DB /* FORMDEF: MAIN BIN */
// FORMDEF=A10110,CHARS=60DB /* FORMDEF: MAIN BIN */
//MSGDS OUTPUT PAGEDEF=A08682 /* MESSAGE DATA SET */
// FORMDEF=A10110,CHARS=60DB /*
/*
//FONT01 DD DSN=SYS1.FONTLIB,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 */
/*
//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 */
// JOBHDR=*.JOBHDR, /* JOB HEADER SEPARATOR OUTPUT */
// JOBTLR=*.JOBTLR, /* DATA SET HEADER SEPARATOR */
// DSHDR=*.DSHDR, /* MESSAGE DATA SET OUTPUT */
// FORMDEF=A10110, /* DEVICE FORMDEF DEFAULT */
// PIMSG=(YES,16), /* ACCUMULATE DATA SET MESSAGES */
// DATAK=BLOCK, /* BLOCK DATA CHECKS */
// TRACE=YES, /* BUILD INTERNAL TRACE */
// FAILURE=WCONNECT, /* ATTEMPT RECONNECT */
// DISCINTV=0, /* NO TIME OUT */
// MGMTMODE=IMMED, /* MAINTAIN SESSION */
// IPADDR='9.99.12.33' /* IP ADDRESS OF PRINTER */
/* DEFAULT PAGEDEF AND CHARS ARE SPECIFIED BY JES3 */
/* PAGEDEF=A868 /* COPY OF A08682 */
/* CHARS=60DB /* JES3 DEFAULT FONT */
//PRT3 ENDCNTL
```

Figure 23 (Part 3 of 3). Sample PSF Startup Procedures for Three Separate Printers

## JCL for PSF Startup Procedures

This section describes the statements and the parameters shown in Figure 22 on page 69 and Figure 23 on page 71.

### 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. For more information on region size, see “Selecting Region Sizes” on page 112. To prevent TIMEOUT abends, you can include TIME=NOLIMIT.

Some optional PSF parameters can be specified with the PARM parameter in the EXEC statement:

```
// EXEC PGM=APSPPIEP,PARM=(NSTddname, tracetype, prompt, pagecount, tcPIP_name)
```

Several of the PARM field options are for tracing. For a description of the PARM field options and for more information about PSF traces, refer to *PSF for OS/390: Diagnosis*.

**Note:** You must specify the options in the correct order with the PARM parameter. If you do not want to specify a value for a given parameter, type a comma in its place. For example, if you want to specify PSF prompting, type:

```
// EXEC PGM=APSPPIEP,PARM=(, ,PROMPT)
```

## OUTPUT

Specifies the page definitions, form definitions, and fonts used to format and print different pages in a print job or data set. In the examples, these pages include job and data set header pages, job-trailer pages, and pages used for printing messages.

For information about how to specify OUTPUT so that the messages are not printed, see “Redirecting the Message Data Set” on page 113.

## DD

Identifies the system libraries that contain resources (fonts, page segments, overlays, page definitions, and form definitions) used for printing.

**Note:** If you are using Overlay Generation Language/370, 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 $nnn$  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 19 on page 56 and Figure 20 on page 61. For a description of the PRINTDEV parameters, see Chapter 6, PRINTDEV Parameters.



## 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 $nnn$  statement (under JES2), or the JNAME parameter (under JES3).

---

## 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 could be used when 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 the following:

- Job separator pages
- Forms marking
- Carrier-strip marking
- Copy-marking or offset stacking

Job separation is controlled by means of 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 56 and "Parameters for the JES3 Initialization Statement" on page 62.

The content of the header and trailer pages is controlled through PSF installation exits APSUX01 and APSUX02. PSF supplies default versions of these exits. A sample header page produced by the APSUX01 module supplied with PSF is shown in Figure 24 on page 76.

```

PPPPPPPPPP RRRRRRRRRR IIIIIIIIII NN NN TTTTTTTTTT IIIIIIIIII TTTTTTTTTT
PPPPPPPPPP RRRRRRRRRR IIIIIIIIII NNN NNN TTTTTTTTTT IIIIIIIIII TTTTTTTTTT
PP PP RR RR II NNN NNN TT TT
PP PP RR RR II NN NN NN TT TT
PPPPPPPPPP RRRRRRRRRR IIIIIIIIII NN NN NN TT TT
PPPPPPPPPP RRRRRRRRRR IIIIIIIIII NN NN NN TT TT
PP RR RR II NN NN NN TT TT
PP RR RR II NN NN NN TT TT
PP RR RR II NN NNN TT TT
PP RR RR IIIIIIIIII NN NNN TT IIIIIIIIII
PP RR RR IIIIIIIIII NN N TT IIIIIIIIII

JJJJJJJJJ 00000000000 BBBB BBBB bbbbbb 11
JJJJJJJJJ 00000000000 BBBB BBBB bbbbbb 1111
JJ 00 00 BB BB bb bb 11
JJ 00 00 BB BB bb bb bbbbbb 11
JJ 00 00 BBBB BBBB bbbbbb 11
JJ 00 00 BB BB bb bb bbbbbb 11
JJ 00 00 BB BB bb bb 11
JJ 00 00 BB BB bb bb 11
JJJJJJJJ 00000000000 BBBB BBBB bbbbbb 11111111
JJJJJJ 00000000000 BBBB BBBB bbbbbb 11111111

QQQQQQQQQ
QQQQQQQQQ
QQ QQ
QQ QQ
QQ QQ
QQ QQ
QQ QQ
QQ QQ
QQ QQ
QQ QQ
QQ QQ
QQ QQ
QQQQQQQQQ
QQQQQQQ QQ

*START JOB 61 PRINTIT 3 001 001 LOCAL USERNAME ROOM 3.57.04 PM 28 SEP 1987 PRT 9 LITS Q
*START JOB 61 PRINTIT 3 001 001 LOCAL USERNAME ROOM 3.57.04 PM 28 SEP 1987 PRT 9 LITS Q
*START JOB 61 PRINTIT 3 001 001 LOCAL USERNAME ROOM 3.57.04 PM 28 SEP 1987 PRT 9 LITS Q
*START JOB 61 PRINTIT 3 001 001 LOCAL USERNAME ROOM 3.57.04 PM 28 SEP 1987 PRT 9 LITS Q
*START JOB 61 PRINTIT 3 001 001 LOCAL USERNAME ROOM 3.57.04 PM 28 SEP 1987 PRT 9 LITS Q
*START JOB 61 PRINTIT 3 001 001 LOCAL USERNAME ROOM 3.57.04 PM 28 SEP 1987 PRT 9 LITS Q
*START JOB 61 PRINTIT 3 001 001 LOCAL USERNAME ROOM 3.57.04 PM 28 SEP 1987 PRT 9 LITS Q
*START JOB 61 PRINTIT 3 001 001 LOCAL USERNAME ROOM 3.57.04 PM 28 SEP 1987 PRT 9 LITS Q
*START JOB 61 PRINTIT 3 001 001 LOCAL USERNAME ROOM 3.57.04 PM 28 SEP 1987 PRT 9 LITS Q
*START JOB 61 PRINTIT 3 001 001 LOCAL USERNAME ROOM 3.57.04 PM 28 SEP 1987 PRT 9 LITS Q
*START JOB 61 PRINTIT 3 001 001 LOCAL USERNAME ROOM 3.57.04 PM 28 SEP 1987 PRT 9 LITS Q
*START JOB 61 PRINTIT 3 001 001 LOCAL USERNAME ROOM 3.57.04 PM 28 SEP 1987 PRT 9 LITS Q
*START JOB 61 PRINTIT 3 001 001 LOCAL USERNAME ROOM 3.57.04 PM 28 SEP 1987 PRT 9 LITS Q
*START JOB 61 PRINTIT 3 001 001 LOCAL USERNAME ROOM 3.57.04 PM 28 SEP 1987 PRT 9 LITS Q
*START JOB 61 PRINTIT 3 001 001 LOCAL USERNAME ROOM 3.57.04 PM 28 SEP 1987 PRT 9 LITS Q

```

Figure 24. Example of a Default PSF Job-Header Page (APSUX01)

PSF also provides optional replacements for APSUX01 and APSUX02 that create header and trailer pages containing fewer characters. A sample header page created by the optional separator page exit is shown in Figure 25 on page 77. This format should print 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, APSUX02), shown in Figure 24, rather than the optional separator pages (APSUX01P, APSUX02P).

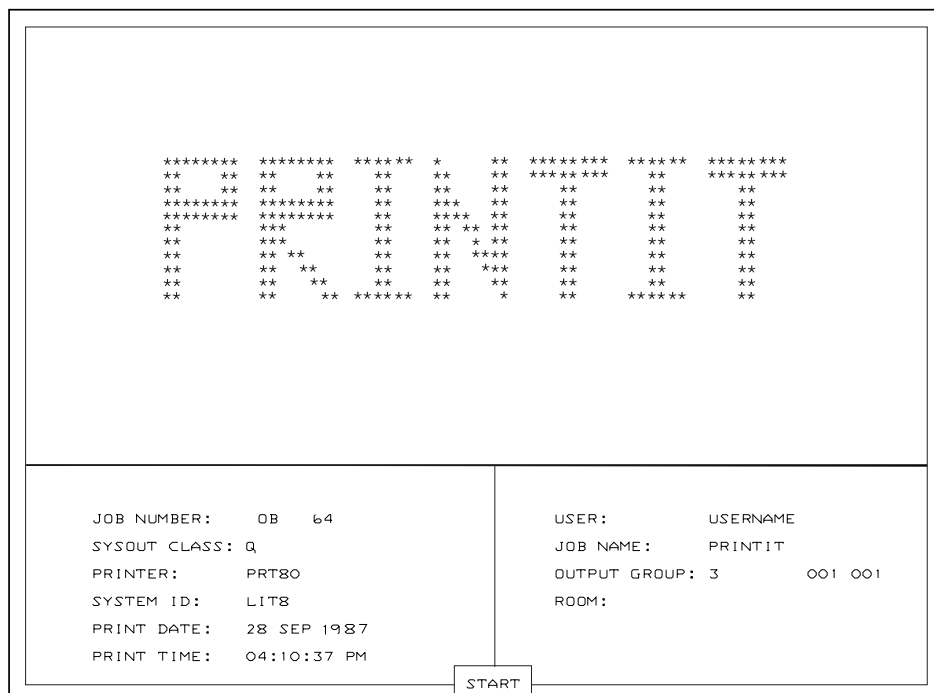


Figure 25. Example of a Fast-Printing Job-Header Page for PSF (APSUX01P). The appearance of your page may differ slightly from the example shown.

In addition, PSF supplies source code for a version of APSUX01 and APSUX02 to create job separators that use guaranteed print labeling (APSUX01S, APSUX02S). For more information about guaranteed print labeling, refer to *Print Services Facility: Security Guide*.

You can also create your own separator page exits. For more information, see “Changing Exit Routines” on page 169 and “Separator-Page Exits: APSUX01, APSUX02, and APSUX03” on page 179.

### Additional Distribution Information

The separator page produced by the IBM-supplied exits may 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 on these keywords and how to specify them, refer to *PSF for OS/390: User's Guide* and the JCL reference publication. The order numbers for these publications are listed in the “Bibliography” on page 389.

If the following requirement is met, additional distribution information appears on the separator pages:

- The IBM-supplied exits are assembled with SYSPARM set to any value except null.

The SYSPARM value causes the exits to use the format that places the additional distribution information on the separator page. If SYSPARM is omitted, the exits are assembled to produce the old format.

The default APSUX01 and APSUX02 exits are shipped assembled with SYSPARM specified. The default exits place the additional distribution information on the separator page. If the default APSUX01 and APSUX02 exits are used without modification, no assembly is required to obtain the additional distribution information.

Here is an example of how to specify SYSPARM:

```
//ASM EXEC asm_pgm,PARM=(SYSPARM(YES))
```

A sample header page containing the additional distribution information produced by the default APSUX01 exit supplied with PSF is shown in Figure 26.



Figure 26. Sample Header Page. Additional distribution information is produced by the default APSUX01 exit supplied with PSF.

A sample header page containing the additional distribution information produced by the optional “fast printing” APSUX01 exit (APSUX01P) is shown in Figure 27 on page 79.

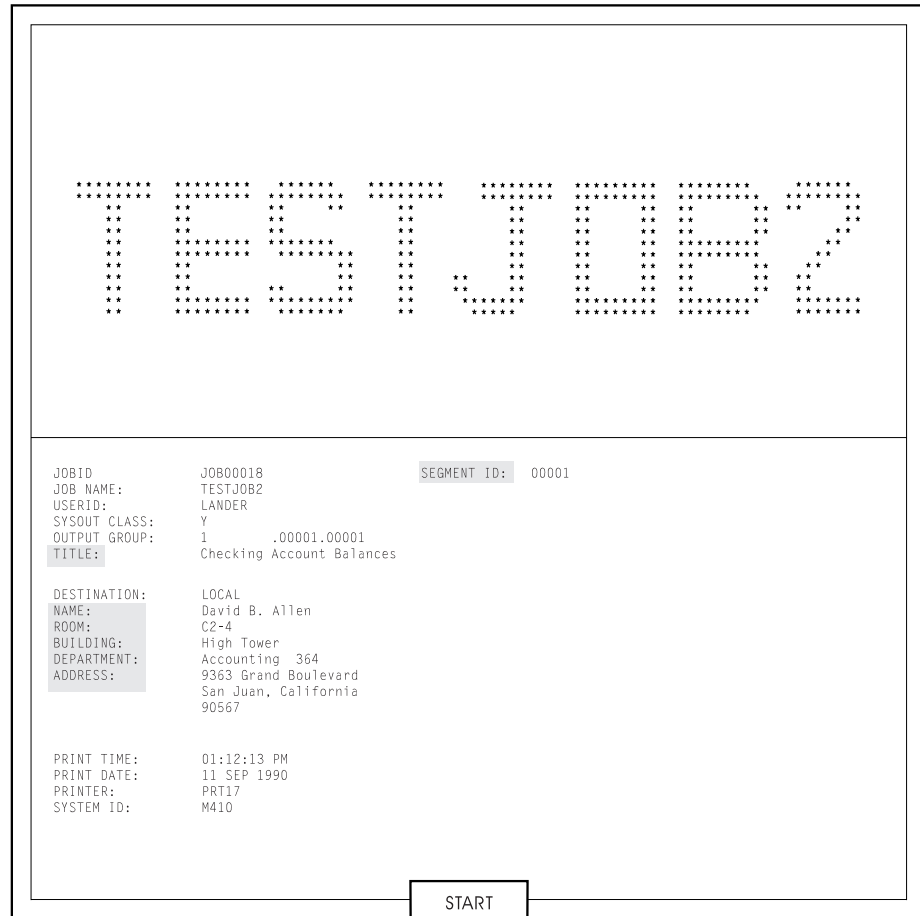


Figure 27. Sample Header Page. Additional distribution information is produced by the optional “fast printing” APSUX01 exit (APSUX01P).

## Separator Page Resources

The form definition, page definition, and font 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 67.

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 desired print direction. 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 more information on default media origins, refer to *Advanced Function Presentation: Printer Information*. For information on the coordinate system used to define page and media origins, refer to *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 3 copies of your trailer page, specify:

```
//JOBTRLR OUTPUT PAGEDEF=V06483,CHARS=60DB,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=60DB,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 67 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 COPIES parameter because this form does not apply to separator pages. Always use the COPIES=(,*n*) form as shown in the example.

For additional information on the COPIES parameter, refer to *PSF for OS/390: User's Guide*.

If forms marking is specified, the 3800 printer automatically prints multiple copies of trailer pages. See “Forms Marking” on page 81.

## 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 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. When you do this, however, a change of forms will usually cause an intervention message to be sent to the JES3 console. In this environment, the intervention message is neither necessary nor desirable. Because DPF is a spooling device, no forms actually 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 this is how your installation is set up, 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 28 and 29 show. If separator pages are turned off, the marks appear on the edges of a blank page that PSF has inserted at the end of the job. If there is a job header page but no job trailer page, the marks appear on the edges of the header page.

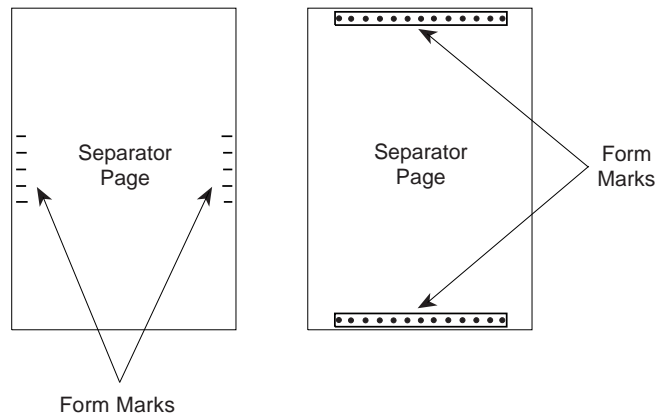


Figure 28. Example of Forms Marking for Cut-Sheet Paper. The actual mark depends on your printer.

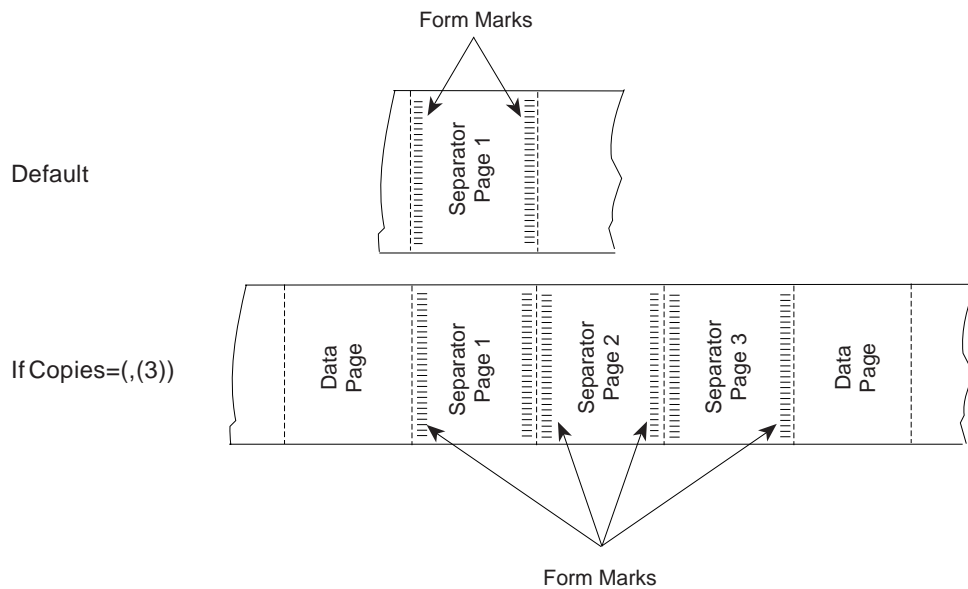


Figure 29. Forms Marking for Continuous Forms Not Printed on a 3800 Printer. Marks are not printed across perforations.

Forms marking is controlled by the following JES initialization options:

- For JES2, select the MARK option on the PRTnnnn statement.
- For JES3, select BURST=(YES,M) on the DEVICE statement.

For more information, see “Sample Initialization Statements for JES2” on page 56 and “Sample Initialization Statements for JES3” on page 61.

For a 3800 printer, forms marking extends across the perforation as shown in Figure 30.

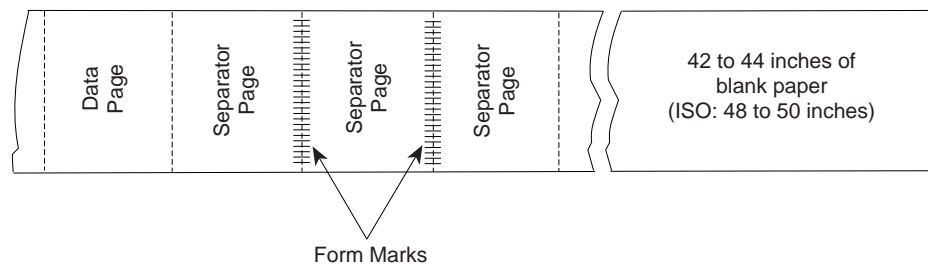


Figure 30. Example of Forms Marking. This example is for 7-inch, 8.5-inch, and 11-inch continuous forms.

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. Once 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).

When you are processing multiple-page job trailers in APSUX02, 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. APSUX02 should set ECAMFTRL to 1 before putting the first record of the first page of the multiple-page job trailer, and should set ECAMFTRL to 0 before putting 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 PSF installation exits. This support enables you to mark or offset-stack the interrupt message page(s) that PSF inserts into the printed output. For more information, see “Considerations on Forms Marking of an Interrupt Message Page” on page 205.

## Carrier-Strip Marking

The 3800 printer also supports carrier-strip marking, diagrammed in Figure 31.

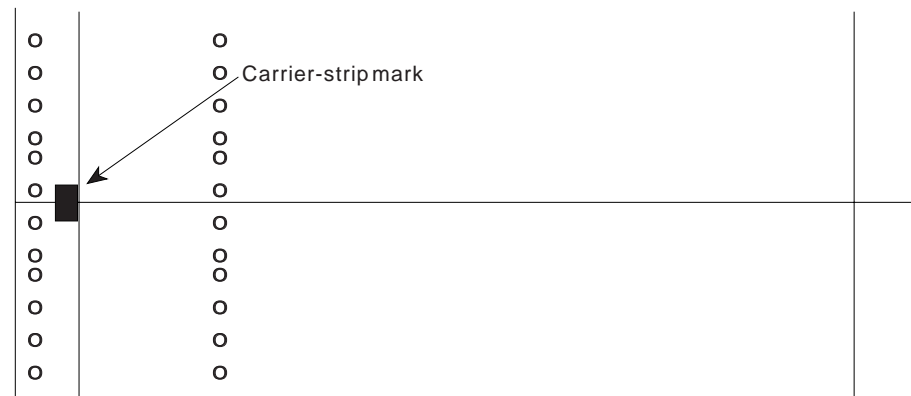


Figure 31. Example of Carrier-Strip Marking on the 3800 Printer

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 PRTnnnn 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 will be enabled until it is reset to X'00', or until the end of the job. ECAMFCS can be set in any exit except APSUX05. For separator page exits, set the field value before putting the first record. For basic information about exits, see Chapter 18, Using Installation Exits.

**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 will not be printed. For more information about carrier-strip marking, refer to *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.

Copy marking is illustrated in Figure 32.

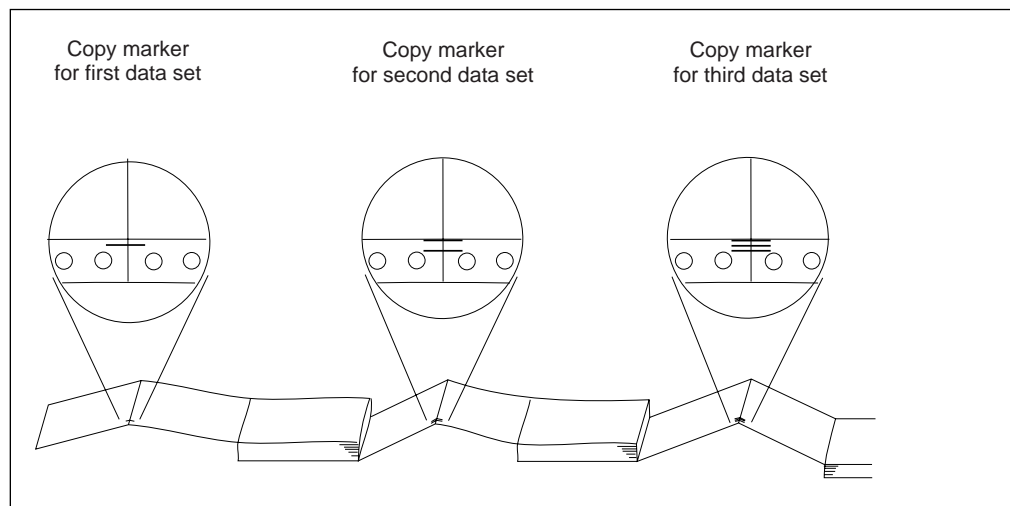


Figure 32. Example of Copy Marking to Simulate Offset Stacking

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 the following JES initialization parameters:

- For JES2, the COPYMARK parameter of the PRT $n$  statement specifies when the copy mark is to be incremented: at the end of each data set or copy of a data set, or at the end of each job.

For JES2, the COPYMARK parameter of the PRT $n$  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 JES2 Initialization Statement” on page 56.

- 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 62.

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 on the form definition, refer to *PSF for OS/390: User's Guide*.

---

## 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 $n$  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 56 and “Parameters for the JES3 Initialization Statement” on page 62.

The content of the data set header page is controlled by PSF installation exit APSUX03. 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 169 and “Separator-Page Exits: APSUX01, APSUX02, and APSUX03” on page 179.

The form definition, page definition, and font 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 on the OUTPUT statements used with separator pages, see “Job Separator Pages” on page 75.

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 84.

---

## 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, PSF abends, 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 as follows:

- 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.

For information about printer support for disabled mechanisms, refer to *Advanced Function Presentation: Printer Information*.

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 could be defined for print data sets that are tolerant of disabled mechanisms. This class would be defined for print-job submitters who, for example, will accept simplex instead of duplex output. Another class could 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.

---

## System Management Facility Type 6 Records

In deferred-printing mode, PSF creates a system management facility (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 terminated, interrupted, restarted by JES operator)
- Logical input-record count
- Logical page count
- Copy groups
- SYSOUT class
- Physical-page (forms) count
- Impression (side) count
- Form name
- CHARs fonts and UCS fonts
- Number of fonts, 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 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, APSUX01P, APSUX01S, APSUX02, APSUX02P, and APSUX025) contain an example of how to invoke the SWBTUREQ macro. For more information about the SWBTUREQ macro, refer to the *Authorized Assembler Services Reference* for your operating system, listed in the “Bibliography” on page 389. For more information about SWBs and how to use the SWB token to retrieve keywords, refer to the *Using the Functional Subsystem Interface* publication for your operating system, listed in the “Bibliography” on page 389.

- 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 may 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, refer to *Print Services Facility: Security Guide*.

PSF supplies an installation exit point that you can use to provide an exit routine (module APSUX05) to modify SMF type 6 records. If you have not provided this routine, PSF creates a standard type 6 record and writes it to the SMF data set.

For information about the APSUX05 module used to modify SMF type 6 records, see “SMF Type 6 Record Exit: APSUX05” on page 193. For information about the mapping of fields in SMF type 6 records, see “Structure of SMF Type 6 Records.”

The SMF type 6 accounting records 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

Figure 33 shows the map of SMF type 6 records. Values in parentheses are in hexadecimal. Offsets and constants in **bold** represent the latest updates to the map.

| Offsets   | Type | Length    | Name        | Description                                   |
|---|------|-----------|-------------|---|
| =====<br>RECORD TYPE 6 - SYSOUT WRITER<br>IN JES2, WRITTEN FOR EACH JOB OUTPUT ELEMENT, WHICH<br>REPRESENTS A GROUP OF DS DIFFERENTIATED BY PUNCH OR<br>PRINTER SETUP & TYPE OF OUTPUT(EG HELD VS NON-HELD).<br>IN JES3, WRITTEN FOR EACH COPY OF A DATA SET<br>===== |      |           |             |   |
| 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<br>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...        | *   |
|   |      | ....      | .1..        | SMF6DIE                                       |
|   |      | ....      | ..1.        | *   |
|   |      | ....      | ...1        | SMF6CBIE                                      |
|   |      |           |             | CONTROL BLOCK INPUT<br>ERROR@ZA05171          |
| 56  | (38) | UNSIGNED  | 1 SMF6NDS   | # DS PROCESSED BY WRITER                      |
| 57  | (39) | CHARACTER | 4 SMF6FMN   | FORM NUMBER                                   |

Figure 33 (Part 1 of 6). Map of SMF Type 6 Records

|    |                |    |          |   |
|----|----------------|----|----------|---|
| 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<br>EXTWTR=0, JES2-2, JES3-5,<br>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. ....      |    | SMF60CN  | TERMINATED BY OPERATOR  |
|    | ...1 ....      |    | SMF6ORD  | INTERRUPTED BY OPERATOR (JES2)<br>OPERATOR RESTARTED DATA SET<br>WITH DESTINATION(JES3)   |
|    | .... 1...      |    | SMF6OR   | RESTARTED BY OPERATOR   |
|    | .... .1..      |    | SMF6ROR  | CONTINUATION OF INTERRUPTED<br>GROUP(JES2) RECEIVED OP<br>RESTARTED DS(JES3)  |
|    | .... ..1.      |    | SMF60SS  | CARRIAGE OVERRIDEN BY<br>OPERATOR(JES2) OPERATOR<br>STARTED WITH SINGLE<br>SPACE(JES3)  |
|    | .... ...1      |    | SMF6INT  | PUNCH WAS INTERRUPTED   |
| 67 | (43) BITSTRING | 1  | SMF6INDC | BIT INDICATORS. BITS 0-3 ARE<br>RESERVED FOR FUTURE EXPANSION<br>OF DATA SET CONTROL<br>INDICATORS. BITS 4-7 ARE<br>RECORD LEVEL INDICATORS TO BE<br>SET IN BIT VALUE FORMAT.<br>EXAMPLE: LEVEL 1=X'01', LEVEL<br>12=X'0C'. THIS NUMBER WILL BE<br>INCREMENTED BY 1 EACH TIME A<br>NEW RELEASE CHANGES THE RECORD.  |
| 68 | (44) CHARACTER | 4  | SMF6JNM  | WHEN SMF6INDC CONTAINS A X'1',<br>THIS FIELD CONTAINS A<br>FOUR-DIGIT EBCDIC JOB NUMBER<br>WHEN SMF6INDC CONTAINS A X'3'<br>OR GREATER, AND THE JOB NUMBER<br>HAS MORE THAN 4 DIGITS, THIS<br>FIELD CONTAINS ZEROS. IF THE<br>JOB NUMBER IS < OR = TO 9999,<br>THIS FIELD CONTAINS THE JOB<br>NUMBER. FOR AN APPC<br>TRANSACTION, THIS FIELD<br>CONTAINS ZEROES. THE CORRECT<br>JOB NUMBER OR APPC TRANSACTION<br>ID IS FOUND IN SMF6JBID |
| 72 | (48) CHARACTER | 8  | SMF6OUT  | LOGICAL OUTPUT DEVICE NAME<br>(FOR THE 3820, ACF/VTAM<br>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<br>COUNT  |
| 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  |

Figure 33 (Part 2 of 6). Map of SMF Type 6 Records

|   |      |           |     |            |                              |
|---|------|-----------|-----|------------|------------------------------|
| 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   | SMF60PR    | 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 |
| 0   | (0)  | SIGNED    | 2   | SMF6LN2    | PRINTING SUBSYSTEM SECT      |
| 2   | (2)  | CHARACTER | 1   | SMF6CPS(8) | LENGTH OF FIRST EXTENSION    |
| 10  | (A)  | CHARACTER | 16  | SMF6CHR    | INCLUDING THIS FIELD         |
|   |      |           |     |            | COPIES DISTRIBUTION          |
| 26  | (1A) | CHARACTER | 4   | SMF6MID    | TRANSLATE TABLE NAMES        |
|   |      |           |     |            | FROM CHARS PARAMETER         |
|   |      |           |     |            | 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.. .... |     | SMF60PJ    | 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       |
| 39  | (8B) | CHARACTER | 3   | *          | RESERVED                     |
| 142   | (8E) | CHARACTER | 20  | SMF60TOK   | OUTPUT GROUP TOKEN           |

Figure 33 (Part 3 of 6). Map of SMF Type 6 Records



```

=====
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   |
| 12 | (C)  | SIGNED    | 4   | SMF6OVLY | NUMBER OF OVERLAYS USED  |
| 16 | (10) | SIGNED    | 4   | SMF6LQLY | 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 PROCESSED                                     |
| 32 | (20) | SIGNED    | 4   | SMF6FEET | NUMBER OF FEET OF DOCUMENT PRINTED (ZERO FOR THE 3820)                     |
| 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. .... |     | SMF6JTTP | JOB TRAILER PAGE WAS PRINTED   |
|    |      | ...1 .... |     | SMF6DPLS | DATA PAGE LABELING WAS SUPPRESSED  |
|    |      | .... 1... |     | SMF6UPAS | USER PRINTABLE AREA WAS SUPPRESSED   |

Figure 33 (Part 4 of 6). Map of SMF Type 6 Records

|  |             |               |          |                 |   |
|--|-------------|---------------|----------|-----------------|---|
| 47   | (2F)        | BITSTRING     | 1        | *               | RESERVED  |
| 48   | (30)        | SIGNED        | 4        | SMF6NSOL        | NUMBER OF SECURITY OVERLAYS<br>USED                                     |
| 52   | (34)        | SIGNED        | 4        | SMF6NSFO        | NUMBER OF SECURITY FONTS USED   |
| 56   | (38)        | SIGNED        | 4        | SMF6NPS         | NUMBER OF SECURITY PAGE<br>SEGMENTS USED                                |
| 60   | (3C)        | CHARACTER     | 8        | SMF6FDM         | FORMDEF NAME  |
| 68   | (44)        | CHARACTER     | 8        | SMF6PDM         | 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  |
| <b>116</b>   | <b>(74)</b> | <b>SIGNED</b> | <b>4</b> | <b>SMF6LPGE</b> | <b>COUNT OF LOGICAL PAGES</b>   |
| =====  |             |               |          |                 |   |
| MULTI-BINS HEADER SECTION (OFFSET DEFINED BY SMF6BN0F) |             |               |          |                 |   |
| =====  |             |               |          |                 |   |
| 0  | (0)         | STRUCTURE     | 4        | SMF6BNMI        | MULTI BIN INPUT SECTION   |
| 0  | (0)         | SIGNED        | 2        | SMF6BNLN        | LENGTH OF SECTION INCLUDING<br>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 LNGLTH 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<br>INCLUDING THIS FIELD                           |
| 2  | (2)         | SIGNED        | 4        | SMF6SGID        | SEGMENT IDENTIFIER  |
| 6  | (6)         | BITSTRING     | 1        | SMF6IND         | SECTION INDICATOR   |
|  |             | 1... ..       |          | SMF6SJF         | ERROR OBTAINING SWBTU - SWBTU<br>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<br>CAN BE PROCESSED USING<br>SWBTUREQ MACRO |

Figure 33 (Part 5 of 6). Map of SMF Type 6 Records

| =====                 |                |           |                 |   |
|-----------------------|----------------|-----------|-----------------|---|
| 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 BYTES SENT  |
| 6                     | (6)            | UNSIGNED  | 1 SMF6IP1       | 1ST SEGMENT OF TARGET ADDRESS   |
| 7                     | (7)            | UNSIGNED  | 1 SMF6IP2       | 2ND SEGMENT OF TARGET ADDRESS   |
| 8                     | (8)            | UNSIGNED  | 1 SMF6IP3       | 3RD SEGMENT OF TARGET ADDRESS   |
| 9                     | (9)            | UNSIGNED  | 1 SMF6IP4       | 4TH SEGMENT OF TARGET ADDRESS   |
| 10                    | (A)            | CHARACTER | 12 *            | RESERVED  |
| 22                    | (16)           | UNSIGNED  | 2 SMF6PQLN      | LENGTH OF PRINT QUEUE NAME  |
| 24                    | (18)           | CHARACTER | * SMF6PRTQ      | PRINT QUEUE NAME  |
| =====                 |                |           |                 |   |
| 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                     | <b>DECIMAL</b> | <b>6</b>  | <b>SMF6LEV6</b> | <b>PSF for OS/390 RELEASE 3.1.0</b>                                   |

Figure 33 (Part 6 of 6). Map of SMF Type 6 Records



---

## Chapter 8. Using Direct-Printing Mode

This chapter provides information on how to attach an AFP printer in direct-printing mode, describes what must be done to define PSF as an MVS 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 MVS.

### Notes:

1. See Chapter 2, "Defining Channel-Attached Printers to OS/390" on page 5 if your printer is not yet defined to OS/390 or MVS.
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 library
  - 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 <sup>2</sup>
  - System management facilities (SMF) type-6 processing
  - Notification of job completion
  - Restartable abends
  - PSF installation exits 1–7

---

<sup>2</sup> For more information about page labeling, refer to *Print Services Facility: Security Guide*

- Message data set redirection
5. If PSF is attached to a printer that does not support the selective deletion of page segments and overlays, DPSS will stop processing data sets if jobs require more than 127 page segments or more than 127 overlays. If you have IBM printer microcode that supports extended overlays and page segments, the maximum number of overlays is extended to 32511.
  6. Information about the installation exit for direct-printing mode is given in “Direct-Print Exit: APSUX16” on page 223.
  7. For forms marking, see the APSUX16 begin-data-set call (BDSC), described in “Direct-Print Exit: APSUX16” on page 223.
  8. For additional information, refer to the *Initialization and Tuning* publication for your operating system, listed in the “Bibliography” on page 389.

---

## How to Use Direct-Printing Mode

The procedure for using PSF in direct-printing mode is as follows:

1. Define DPSS to MVS, using the procedures described in “Defining the DPSS” on page 97.
2. Code JCL to invoke direct-printing mode, using “Application Program JCL for Direct-Printing Mode” on page 97 as a sample.
3. Allocate the printer by either:
  - Coding a DD statement in the JCL that is used to invoke 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 100.
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 through the use of exclusive ENQ/DEQ may result in system deadlock. You can prevent system deadlock by coding application level ENQ/DEQ around all system macros that may require ENQ on TIOT. Some examples of system macros that may require ENQ/DEQ are Open, Close, Dynalloc, and EOVS. To avoid unexpected results or abend conditions, review the *Authorized Assembler Services Reference* for your operating system before coding 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 an MVS subsystem by modifying the PARMLIB member IEFSSN00 or by defining a new IEFSSNxx 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 64 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 34 shows a job stream that can create a new IEFSSNxx member of SYS1.PARMLIB. Note that if you create a new IEFSSNxx member, you must update the IEASYSxx member to point to IEFSSNxx, 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 34. Job Stream for Creating an IEFSSNBB Member of SYS1.PARMLIB

Consider the following:

- 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 35 on page 98 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 you have allocated the space 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
//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,
//         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),
//         DATAACK=BLOCK
//DA1      ENDCNTL
//OUT1     OUTPUT FORMDEF=fdefname
//name    DD     UNIT=180,CNTL=*.DA1,OUTPUT=*.OUT1,...
:

```

Figure 35. Sample JCL to Enable Direct Printing

Following are descriptions of the statements and parameters shown in Figure 35:

#### 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 on the PSF PARM parameters and tracing, refer to *PSF for OS/390: Diagnosis*. For more information on the REGION parameter, see “Selecting Region Sizes” on page 112.

#### 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, refer to Chapter 4, “Working with Libraries” on page 37.



## 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 Chapter 6, PRINTDEV Parameters.

## ENDCNTL

Specifies the end of a PRINTDEV statement.

## OUTPUT

This optional statement specifies AFP parameters for PSF. Refer to *PSF for OS/390: 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 containing the DD statement. This DD statement can also be allocated by using dynamic allocation instead of JCL. Refer to the *Authorized Assembler Services Guide* for your operating system 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 3 shows.

Table 3. JCL DD Allocation vs. 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.          |
| 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 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 required. 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 parameter under JES, refer to *PSF for OS/390: 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, refer to *DFSMS/MVS Macro Instructions for Data Sets*.

---

## Chapter 9. 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. (Cut-sheet emulation mode is not supported when connected through Distributed Print Function (DPF) or RPM3.)

Some continuous-forms printers<sup>3</sup> 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 measuring 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.<sup>4</sup>

The output, once it has been slit and collated by post-processing equipment, is identical with that 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. By using installation exit APSUX07, however, you can request that PSF check page sizes to determine whether CSE should be used. APSUX07 gives you 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 APSUX07 options are used only for non-N\_UP copy groups.

---

<sup>3</sup> For the printers and microcode level to support CSE, consult the Customer Engineer for your hardware.

<sup>4</sup> Some printers may offer options enabling 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.

---

## Will Cut-Sheet Emulation Work with 3800 Line-Mode Conversion?

If you are using the 3800 Line-Mode Conversion function and you have 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 APSUX04 has an option you can use to let PSF know which paper to emulate. For more information, see “Logical-Record Processing Exit: APSUX04” on page 187.

---

## 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 invoked, and therefore would ordinarily would cause a sheet eject, now causes only a half-sheet eject. There are numerous such operations:

- 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 could contain the job trailer for job 1 and the job header for job 2. If you need to prevent this, you can force the header page to eject to a new sheet by assigning it a form definition that specifies N\_UP (N\_UP 1 will work).

- Form definition parameters **PRESENT** and **DIRECTION**, which PSF ordinarily ignores for cut-sheet printers, are honored in CSE mode. This enables CSE to accommodate a wide range of jobs in addition to those formatted specifically for cut-sheet paper, but it has an unfortunate side effect. Customers have sometimes inadvertently coded 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, PSF tries to aid you by forcing the number of interrupt message pages to be an even number. The separator pages, however, still require your attention. To handle them, consider one of the following options:
  - You can request an even number of copies to force the user data to a new sheet. If the print job data contains an odd number of pages, however, this option may still cause the first reprinted page of the data set to be printed on a different half-sheet from the one the original page was printed on.
  - You can turn off separator pages and mark forms instead. However, if the print job contains an odd number of pages, this option can also cause trouble.
  - You can use N\_UP 1 to force the separator pages to be on separate sheets of paper from the user’s data.

- If your printer has an attached post-processor that can do left/right offset stacking, the offset stack (jog) command 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 simply 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.



---

## Chapter 10. Enabling PSF 3800 Line-Mode Conversion

This chapter explains how you can use the PSF 3800 Line-Mode Conversion function to 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. In line mode, however, 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 could 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 enable you to use a limited set of standard AFP functions that are not available on the 3800 line-mode printer.

This chapter explains what 3800 Line-Mode Conversion is, how it works, how to enable it, and some limitations or restrictions that you may encounter in your particular environment.

---

### What is IBM 3800 Line-Mode Conversion?

The PSF 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, it enables you to 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 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 executing 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 specified by **LINECT** or the number 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 on the dimensions of the paper 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 *O1*) in the PSF resource library, PSF issues an error message. Uncollated copies specified by the **COPIES** parameter in the JCL are used in conjunction with **FLASH** to determine how many copy subgroups should be 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 (APSUX04) in deferred-printing mode or through PSF Exit 14 (APSUX14) in direct-printing mode. Exit 4 is the PSF record processing exit, which normally 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 calling APSUX04/APSUX14 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 on PSF exit processing, see Chapter 18, Using Installation Exits.

You can set XTP4NCAL flag ON in Exit 4/14 to indicate that this exit should not be 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 will 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 should 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 should identify the 3800 jobs you want converted. Because the exit has access to all the JCL parameters for the job, you may 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 also set up a table in Exit 4/14 that enables 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 in order 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.

PSF also lets you set the XTP4LMRG flag in Exit 4/14 to indicate to PSF that line merge should be activated. This line merge duplicates the line merge function of a 3800 while in “line mode.” To enable line merge, you should 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.

PSF also lets you set the XTP4DPLX flag in Exit 4/14 to indicate to PSF if duplexing should be enabled and if so, what type is enabled. To enable duplexing, you should 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 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 has been 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 should 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 APSUX04 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 have been 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 had not been identified in Exit 4/14, it would print as a normal AFP job; the MODIFY parameter would simply be 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 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 should also examine jobs that use a special post-processing routine and character set to create the 3800 version of rotated print. Although you may 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 will use 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 will fit on the paper that these cut-sheet printers handle. Although 3800 jobs that print on narrow paper should easily fit on cut-sheet paper, 3800 jobs designed for wide paper will probably 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 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 have the capability to 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 comparing 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 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, JCL, and so on, or to install higher 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 used by the jobs being processed.
- 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 those that cannot be processed by PSF—that is, jobs using 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.



---

## Chapter 11. Managing Resources and Selecting Special Processing Parameters

Resources are collections of data and control information that PSF can use in printing a data set. Form definitions, page definitions, fonts, page segments, overlays, and object containers are all types of PSF resources.

PSF resources can be specified in:

- Job entry subsystem (JES) initialization parameters
- PSF startup procedures
- User job control language (JCL)
- Print data set
- Form definitions and page definitions
- PSF resource exit, APSUX07
- Printer defaults (fonts only)

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 processing a data set, PSF deletes all the page segments and overlays that were downloaded to printer storage, except for overlays specified in the 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 219. 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 normally contain many page segments and use a variety of fonts.

Default resources can be defined when PSF is started. For instructions, see “PSF Startup Procedures for AFP Printers” on page 67, and Chapter 8, “Using Direct-Printing Mode” on page 95. 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.

---

### How PSF Selects Special Processing Parameters

This section describes the following special processing parameters:

- Selecting a region size
- Printing PSF error messages
- Specifying checkpoint intervals

Some processing parameters can be specified in:

- JES initialization parameters
- PSF startup procedures
- User JCL
- Active page definitions and form definitions

PSF selects these processing parameter values in the following order:

1. In a form definition named on a JCL OUTPUT statement, including any inline form definitions named on the OUTPUT statement.
2. In the user JCL.
3. In JES initialization statements.
4. In the PRINTDEV statement.
5. In a form definition named in the PRINTDEV statement, including any inline form definitions named on the PRINTDEV statement.

## 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 have to be stored. Each additional FSA requires storage for control blocks and buffers. Storage acquired during initialization remains static and is not released until the FSA is drained.

Additional storage for certain FSA control blocks is required 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 should be considered when you calculate the region-size requirements.

Table 4 shows the minimum storage required for PSF.

| <i>Table 4. Minimum Storage Required for FSS and FSAs. MB = megabyte.</i> |  |   |
|---|--|---|
| <b>FSA</b>  | <b>Channel-attached</b>                    | <b>SNA- or TCP/IP-attached</b>            |
| The <b>first</b> FSA in an FSS  | 1 MB below 16 MB<br>11 MB above 16 MB      | 0.5 MB below 16 MB<br>11 MB above 16 MB   |
| Each <b>additional</b> FSA in the FSS                                     | 0.65 MB below 16 MB<br>0.25 MB above 16 MB | 0.1 MB below 16 MB<br>0.25 MB above 16 MB |

## Installation Options

The following installation options can affect storage requirements during the initialization of PSF:

- Size of the trace table
  - PSF default is 128 KB per FSA.
  - For more information on tracking, refer to *PSF for OS/390: 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 the following 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 will cause the allocation of an additional amount of virtual storage, calculated as follows:  
(BUFNO parameter) \* resource library block size

- The printer output buffer is calculated with the following formula:  
(BUFNO parameter) \* 32 KB per FSA

The BUFNO value is taken from the PRINTDEV statement. It specifies a factor for allocating real, fixed storage used in printer I/O operations. (The BUFNO value is multiplied by 32 KB (32768 bytes) to determine the total amount of reserved storage.) The value of BUFNO can range from 1 to 10000.

Below-the-line storage for all channel-attached printers, except the 3800 printer, can be from 6 KB to 200 KB per FSA with BUFNO values from 1 to 10000 respectively.

Below-the-line storage for the 3800 printer can be from 38 KB to 486 KB per FSA with BUFNO values from 1 to 15 respectively. For 3800 printers, PSF uses the maximum BUFNO value of 15 if a BUFNO value greater than 15 is specified.

Above-the-line storage for all channel-attached printers, except the 3800 printer, can be from 32 KB to 320 MB per FSA with BUFNO values from 1 to 10000 respectively.

There is no above-the-line storage for the 3800 printer.

**Note:** The BUFNO parameter is ignored for TCP/IP-attached printers and for SNA printers using VTAM.

For more information, see “Buffers” on page 239.

### For Each FSA

Following are some characteristics of print jobs that place additional storage requirements on each FSA:

- Number of resources 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

## Redirecting the Message Data Set

The errors found during the printing of a user's data set are printed as a message data set at the end of the user's print data set. You can redirect this message data set as a SYSOUT data set to another CLASS or DEST for viewing or printing. This function does not include redirecting messages printed within a user's data set, caused by paper jams, operator commands, and so on. If you wish, you can use installation Exit 7 to suppress messages printed within a user's data set. For details, see “Considerations on Suppressing an Interrupt Message Page” on page 202.

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.

To activate redirection of message data sets, 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 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=60DB,DATAACK=BLOCK
```

In this example, each message data set will be 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 specified in the 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.

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.

An alternative way to activate redirection of message data sets is to set XTP7MDSD with APSUX07. All the keywords specified on OUTPUT statement for the message data set referenced by the MESSAGE= keyword in the PRINTDEV statement will be attributed to the generated SYSOUT data set. For details, see “Considerations on Redirecting a Message Data Set” on page 202.

For the following example we 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 generated in the middle of a user’s print data set (unless such messages have been suppressed by instructions in installation 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 may be degraded, because fewer pages are printed in the same number of CPU cycles. To minimize such degradation, reduce the number of FSAs for each FSS.

## Printing PSF Error Messages

The PIMSG parameter, which can be coded in the user JCL or in the PRINTDEV statement, controls the printing of PSF error messages at the end of the data set. The PIMSG parameter specified in the user JCL takes precedence over that specified in the PRINTDEV statement. For more information on the PIMSG parameter in the user JCL, refer to *PSF for OS/390: 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 invokes 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 67.

## Specifying Checkpoint Intervals

Checkpoint intervals can be specified in the following places, and PSF uses the values 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. Refer to the *JCL Reference* for your operating system, in the “Bibliography” on page 389, for information.

2. Applicable JES initialization parameters:

- JES2 initialization CKPTPAGE, CKPTSEC, and CKPTMODE parameters. See “Parameters for the JES2 Initialization Statement” on page 56.
- JES3 initialization CKPNTPG and CKPNTSEC parameters. See “Parameters for the JES3 Initialization Statement” on page 62.

---

## How PSF Selects Resources

PSF resources can be specified in:

- JES initialization statements
- PRINTDEV statement
- JCL for the print job
- Print data set
- Form definitions and page definitions
- PSF resource exit, APSUX07
- Printer defaults (fonts only)

Tables 7 through 12 on pages “Form Definitions in Deferred-Printing Mode” on page 117 through “Fonts in Direct-Printing Mode” on page 121 show how the page definition, the form definition, and the 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<sup>5</sup>. 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=60DB
//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=60DB
//STEP01 EXEC PGM=ENDMONTH
//DD1 DD SYSOUT=A
```

For more information about the OUTPUT statement, refer to the JCL reference publication for your operating system.

## Selecting a Forms Flash for a 3800 Printer

To select a forms flash for the IBM 3800 printer, you use control statements, in the order shown in Table 5 and Table 6.

| <i>Table 5. Hierarchy of Flash Selection in Deferred-Printing Mode</i> |   |                     |
|--|---|---------------------|
| User JCL   | FLASH specified by FORMDEF= on OUTPUT statement (see Notes 1 and 2) |                     |
|  | DD statement  | FLASH= (see Note 3) |
|  | OUTPUT statement  | FLASH= (see Note 3) |
| JES default in the current printer setup                               |   |                     |
| FLASH specified by FORMDEF= in PRINTDEV statement (see Notes 2 and 3)  |   |                     |

| <i>Table 6. Hierarchy of Flash Selection in Direct-Printing Mode</i> |   |                     |
|--|---|---------------------|
| User JCL   | FLASH specified by FORMDEF= on OUTPUT statement (see Notes 1 and 2)   |                     |
|  | DD statement  | FLASH= (see Note 3) |
|  | OUTPUT statement  | FLASH= (see Note 3) |
|  | FLASH specified by FORMDEF= in PRINTDEV statement (see Notes 2 and 3) |                     |

<sup>5</sup> You cannot use an "implicitly referred to" OUTPUT statement on a printer in direct-printing mode.

**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 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 will the forms flash currently loaded in the printer be 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 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 used to print the data set.
- If an IMM is not present, PSF uses the first copy group in the form definition used to print the data set.

See “Form Definitions in Deferred-Printing Mode” and “Form Definitions in Direct-Printing Mode” on page 118 for the hierarchies that PSF follows to select the form definition used to print the data set.

## Form Definitions in Deferred-Printing Mode

| <i>Table 7. Hierarchy of Form Definition Selection in Deferred-Printing Mode</i>                       |                  |  |           |
|--|------------------|--|-----------|
| Internal copy group immediately followed by an IMM structured field in the print data set (see Note 1) |                  |  |           |
| User JCL   | OUTPUT statement | Explicitly referred to (see Notes 2 and 3) | FORMDEF = |
|  |                  | Implicitly referred to step-level          | FORMDEF = |
|  |                  | Implicitly referred to job-level           | FORMDEF = |
| First inline form definition (see Note 4)  |                  |  |           |
| PRINTDEV statement from the PSF startup procedure  |                  |  | FORMDEF = |

**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.
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 there are inline form definitions, the first inline form definition in the print data set is selected.

## Form Definitions in Direct-Printing Mode

| <i>Table 8. Hierarchy of Form Definition Selection in Direct-Printing Mode</i>              |  |           |
|---|--|-----------|
| Internal copy group immediately preceding the IMM record in the print data set (see Note 1) |  |           |
| User JCL  | Explicitly referred to OUTPUT statement (see Note 2) | FORMDEF = |
| First inline form definition (see Note 3)   |  |           |
| User JCL  | PRINTDEV statement                                   | FORMDEF = |

**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.
3. If the user JCL does not specify the form definition parameter and there are inline form definitions, 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

|  |                               |           |
|--|-------------------------------|-----------|
| User JCL   | OUTPUT statement (see Note 1) | PAGEDEF = |
|  | DD statement                  | FCB =     |
|  | OUTPUT statement (see Note 1) | FCB =     |
| JES default in the current printer setup (see Notes 2 and 3)   |                               |           |
| First inline page definition (see Notes 4 and 5)               |                               |           |
| PRINTDEV statement from the PSF startup procedure (see Note 3) |                               | PAGEDEF = |

### Notes:

1. The OUTPUT statement 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 $nnn$  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 PRINTDEV statement is only selected if FCB is specified for the PDEFAULT parameter 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 is used, which invokes page definition P16. To use inline page definitions in JES3, the name of the page definition should be specified 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 has set a default.
5. If an inline page definition is included in the print data set, and either its name matches the name selected or the name from the JCL is DUMMY, the inline page definition is used.
6. A page definition from a user library is used if its name matches the name selected.

## Page Definitions in Direct-Printing Mode

|   |  |           |
|---|--|-----------|
| User JCL                                  | Explicitly referred to OUTPUT statement (see Note 1) | PAGEDEF = |
|   | DD statement (see Note 1)                            | FCB =     |
|   | Explicitly referred to OUTPUT statement (see Note 1) | FCB =     |
| First inline page definition (see Note 2) |  |           |
| User JCL                                  | PRINTDEV statement                                   | PAGEDEF = |

**Notes:**

1. This includes inline page definitions, which are specified by name or with the name DUMMY, on the OUTPUT or DD statement.
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.

**Fonts in Deferred-Printing Mode**

When PSF is printing line-format data, Table 11 applies.

| <i>Table 11. Hierarchy of Font Selection in Deferred-Printing Mode</i>  |                               |        |
|---|-------------------------------|--------|
| Fonts specified by the current page definition. This includes a JES default page definition, but not a page definition specified only by the PRINTDEV statement of the PSF startup procedure. |                               |        |
| User JCL  | DD statement                  | CHARS= |
|   | OUTPUT statement (see Note 1) | CHARS= |
|   | DD statement                  | UCS=   |
|   | OUTPUT statement (see Note 1) | UCS=   |
| JES default font in the current printer setup (see Notes 2 and 3)   |                               |        |
| PRINTDEV statement of the PSF startup procedure   |                               | CHARS= |
| If the PRINTDEV statement of the PSF startup procedure names the active page definition, any fonts specified in that page definition are loaded.  |                               |        |
| Hardware default  |                               |        |

**Notes:**

1. The OUTPUT statement used is the first one found from explicit, step-level implicit, and job-level implicit.
2. Under JES2, you can select fonts with a SET command, the UCS parameter of the PRTnnnn 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 PRTnnnn statement. For more information on selecting fonts, refer to the JES2 publication for your operating system, or to *PSF for OS/390: User's Guide*.
3. 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 is specified in the JES3 DEVICE statement, hardware default fonts or fonts specified in the PRINTDEV statement are selected; otherwise, hardware default fonts or fonts specified in the PRINTDEV statement are never selected. For more information on selecting fonts, refer to the JES3 publication for your operating system, or to *PSF for OS/390: User's Guide*.

## Fonts in Direct-Printing Mode

When PSF is printing line-format data, Table 12 applies.

| <i>Table 12. Hierarchy of Font Selection in Direct-Printing Mode</i>  |   |         |
|---|---|---------|
| Fonts specified by the current page definition (unless the current page definition is specified only by the PRINTDEV statement) |   |         |
| User JCL  | DD statement                            | CHARS = |
|   | Explicitly referred to OUTPUT statement | CHARS=  |
|   | DD statement                            | USC=    |
|   | Explicitly referred to OUTPUT statement | USC=    |
|   | PRINTDEV statement                      | CHARS=  |
| If the PRINTDEV statement names the active page definition, any fonts specified in the page definition are loaded.              |   |         |
| Hardware default  |   |         |

## Searching for a Resource Specified for a Print Job

After PSF determines the name of the resource to be used, it must find that resource. It conducts a search, in the following order:

1. The internal copy group immediately preceding the IMM record in the print data set is searched.
2. If inline resources were sent in the print data set, they are searched second.
3. If user libraries were specified, they are searched in the order in which they were specified.
4. Finally, the system libraries that were specified in the PRINTDEV statement are searched in the order in which they were concatenated. If multiple system libraries have been defined for different resolutions, refer to “Format Resolution Order” on page 130 for more information about how PSF chooses which system library to use.

On the first occurrence of a resource bearing the name specified, PSF stops the search.

**Note:** 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, refer to *Print Services Facility: Security Guide*. An overlay called out in the PRINTDEV is always loaded from the system libraries.

## Using Form Definitions for Basic N\_UP Printing

PSF supports basic N\_UP printing, which enables the job submitter to place up to 4 pages in partitions on a sheet in simplex mode, and up to 8 pages on a sheet in duplex mode. The term N\_UP comes from the Page Printer Formatting Aid/370 (PPFA/370) N\_UP subcommand, which the job submitter specifies when creating a form definition.

N\_UP printing is supported by some, but not all, AFP printers; for a list of the printers supported, refer to *Advanced Function Presentation: Printer Information*. To find the microcode level necessary for N\_UP printing, check the documentation for your printer, or consult your IBM marketing representative.

When processing a form definition containing the basic N\_UP subcommand, the PSF-supported printer divides a side of a sheet into 1 to 4 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 the following in mind when using N\_UP printing:

- To create form definitions containing the N\_UP subcommand, you must have PPFA/370 1.1.0.
- 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 should not adversely affect the performance of PSF, the printer uses more buffer space when accumulating data for multiple pages than when accumulating 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 additional page eject actions when invoking a medium map.

## Using Form Definitions for Enhanced N\_UP Printing

N\_UP printing can enable you 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. Be aware, however, that if the user requests a MICR font for a device that either has 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 38.

For additional information about MICR printing, refer to *Advanced Function Presentation: Printer Information*.

---

## Maintaining Resource Integrity

To maintain the integrity of resources, consider the following:

- 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. A PPTNDSI bit in the MVS program-properties table specifies when resources can be updated. When this bit is set to 1, AFP programs can update resources in the PSF library while PSF is running. When this bit is set to 0, and DISP=SHR, AFP programs cannot update resources in the PSF library until either the FSA or the printer has stopped running.
- If a resource in a library is modified after it is loaded into virtual storage or into printer storage,<sup>6</sup> PSF might not have deleted the unmodified resource from storage.

To make sure that the modified resource is used, do one of the following:

- 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.
- Use the resource exit.

---

<sup>6</sup> Page definitions and form definitions are loaded into virtual storage; fonts, page segments, and overlays are loaded into printer storage.

- 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.
- Do not compress PSF library data sets while PSF is active.
- Exclude PSF library data sets from HSM migration.

---

## Chapter 12. 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 containing resources at the same resolution as the printer.

Today, an enterprise may 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-justified 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. Using routing options and print queues 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 indicating the resolution 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 indicating how to handle any mismatches between the resolution specified when formatting and the resolution of the resource found when printing.
- Exploit printers that accept print data formatted at multiple resolutions.
- Exploit printers that allow the manual switching of print resolution, for example the InfoPrint 4000.

The following sections introduce the key concepts related to printing at multiple resolutions and point to specific sections with more details. The topics discussed are:

- 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 could be switched from one resolution to another, the following procedure was used:

1. Purge 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. There can be a 240-pel JCL DD statement, a default JCL DD statement, or both for each of these resource types.

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 selecting resource libraries for this printer.

For more information about how to define these JCL DD statements, see Chapter 6, “PRINTDEV Parameters” on page 41.

For an example startup procedure set up for multiple resolution libraries, see Figure 22 on page 69.

## Printers with Automatic or Auto-Resolution Mode

PSF now supports printers operating in automatic or auto-resolution mode, unless they are connected through Distributed Print Function (DPF) 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.

In order to exploit 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 Chapter 6, “PRINTDEV Parameters” on page 41.

For an example startup procedure set up for multiple resolution libraries, see Figure 22 on page 69.

For more information about format resolution, see “Format Resolution Specified from the Application” on page 129.

---

## 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 specified to the print data formatting application match the resource libraries specified to PSF when the data is being printed. Failure to do this may 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 generally 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 formatted with fonts and images 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 may already have printers with different print resolutions attached to PSF. Multiple JCL DD statements may be present in the startup procedure defining 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.FONTOLN /* Outline fonts */
//*
//FONT03 DD DSN=SYS1.FONT300 /* 300 pel raster fonts */
// DD DSN=SYS1.FONTOLN /* Outline fonts */
```

In this case the FONT02 font library should be used when formatting data for and printing data on a 240-pel resolution printer. Likewise, the FONT03 font library should be 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 there are overlays and page segments 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.

Separating the resource libraries by resolution lets you define them to PSF through the PRINTDEV JCL statement indicating the resolution of the resources in each particular library.

For more information about how to define these JCL DD statements, see Chapter 6, "PRINTDEV Parameters" on page 41.

For an example startup procedure set up for multiple resolution libraries, see Figure 22 on page 69.

---

## Format Resolution Specified from the Application

In earlier discussions in this chapter, it was pointed out that multiple resolution printers can scale print data 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, there is a companion parameter (Font Fidelity) that PSF uses if there is a mismatch 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, refer to the *PSF for OS/390: 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/370 or OGL/370) 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, refer to the *Mixed Object Document Content Architecture Reference*. For more information about the support of the triplet for your application, refer to that application's documentation.

## Specifying Font Fidelity in the Print Data Stream

MO:DCA has 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 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 referenced in the data stream is found but is not available to PSF at the specified resolution.

The triplet has two settings:

- Stop presentation at the point of the first font resolution exception and report the exception.
- Do not stop presentation due to font resolution exceptions.

PSF uses Font Fidelity 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 terminates 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.

---

## Multiple Resolution and Installation Exits

The PSF installation exit for Resource Management (APSUX07) can be used to specify the format resolution in several ways:

- The installation exit can establish a PSF default format resolution with the initialization call (INIT).
- The installation exit can set the format resolution value for any or all print data sets with the begin data set call (BDSC).

For more information about using the installation exit to set the format resolution, see to “Considerations for Setting a Format Resolution Value” on page 207.

---

## Format Resolution Order

As you can see from the discussions about multiple resolution earlier in this chapter, the 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 and therefore 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 and is therefore capable of accepting 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 (APSUX07) begin-data-set call
  - b. The PSF installation exit for Resource Management (APSUX07) 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 JCL OUTPUT statement RESFMT keyword

PSF uses the determined format resolution to choose the system libraries to use when looking 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 13. 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 means of the available printer commands.
- Characters stored in the print buffer cannot be easily interpreted, because they have been translated from EBCDIC form.
- All raster character patterns stored in the printer (excluding the permanently resident fonts) can be erased. PSF can erase data 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 responding to a STOP WRITER command. When a printer 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 perform 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 perform an NPRO on the last page of one form before JES issues the message telling 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, additional 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) of PSF/2. Sensitive material stored in the PSF resource library should be marked PRIVATE. For more information, see “Using APSRMARK to Mark Resources” on page 157.
- Resources stored in libraries defined in the PSF startup procedure are available to all print jobs. These libraries should not be used for sensitive resources, nor should they 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 38.
- 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.

For specific printers, refer to 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, refer to *Print Services Facility: 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 be sent, use the NOTIFY JCL keyword in the OUTPUT statement. To read more about specifying the NOTIFY keyword, refer to *PSF for OS/390: 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 provided by these profiles, activate both the SMESSAGE class and the SETROPTS RACLIST processing for the SMESSAGE class. For more detailed information, refer to the *RACF Security Administrator's Guide*.

## Chapter 14. 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. To learn which AFP printers support resident fonts, and how many each of them supports, refer to *AFP: Printer Information*.

Resident fonts for different printers are shipped on different media. For most LAN-attached printers, resident fonts are shipped internal to the printer, installed and ready to use. For information about resident fonts for a particular printer, refer to the publications for that printer.

Figure 36 is a diagram of a printer that supports resident fonts.

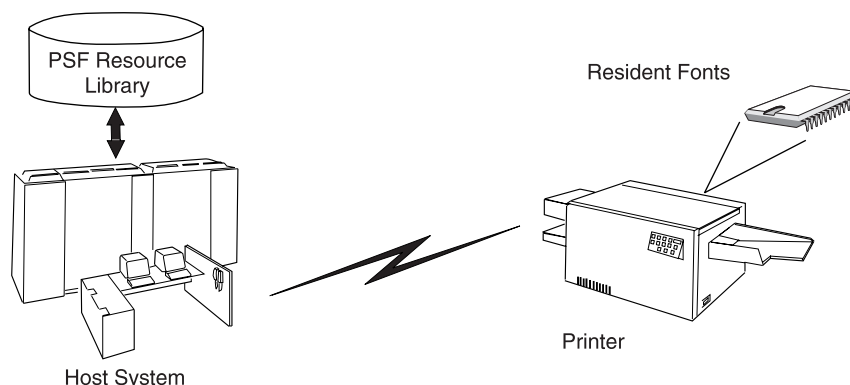


Figure 36. A Printer That Supports Resident Fonts

### Notes:

1. The host version and the printer-resident version of a font may not be identical, especially if your installation has edited either version of the font. Also, printer-resident fonts and fonts stored in other libraries may differ, even though the fonts have the same name.
2. For printers attached through Workstation PrintManager (WPM, available in Japan) or the Distributed Print Function (DPF) of PSF/2, PSF does not support printer-resident fonts, even though the printer itself may support them. PSF Direct, however, 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 15, "Marking Fonts for Font Capture" on page 145.

PSF provides two methods to access resident fonts:

- APSRFTBL (hard-coded table), used to identify resident fonts in printers that support only symbol sets—for example, the 4224, 4230, 4234, 4247, 6408, and 6412 printers. For more information, refer to *AFP: Printer Information*.
- APSRMARK (utility), used for all other printers that support resident fonts.

---

## PSF Resident Font Table (APSRFTBL)

PSF uses the APSRFTBL module, supplied with PSF in the SYS1.SAMPLIB data set, to identify the symbol sets resident in 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 executable code. 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 C, Font Resource Tables Supplied by PSF.

### 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. For the contents shipped with PSF, see “APSRFTBL Font Resource Table” on page 283.

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 a user has specified PRESTIGE 10-point font (character set C0S0PR10) for a print job being 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, duplicate the entry line for the Gothic 10-point font and 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'
```

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.

After changing the table, you must assemble the APSRFTBL module and install (link-edit) APSRFTBL into the applicable LINKLIB.

Similarly, you can modify the relevant table in the APSRFTBL module as needed for your printer: the APSCS24V for the 4230 printer; the APSCS34V for the 4234; the APSCS47V for the 4247 printer; or the APSCS64X for the 64XX printer.

Similarly, you can modify the APSCS34V table in the APSRFTBL module for a 4230, 4234, 6408, or 6412 printer.

---

## 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 invoked with control statements. You can use it to do the following:

- Mark all or individual host code pages and character sets of partitioned data sets, so that PSF will know that there is a printer-resident version of the resource.
- Report the marking status of all or individual members of partitioned data sets.
- Report the marking status of a font resource found in a sequential file that can be subsequently moved into a font library.

Fonts that have been 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 stored in the PSF resource libraries is marked PRIVATE, PSF downloads the font to the printer. A font 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, and the 4028 Font Metrics, are shipped *already* marked PUBLIC. For more information about marking the fonts, see “Syntax of APSRMARK Control Statements” on page 137 and “APSRMARK Control Statement Parameters for Resident Fonts” on page 138.

If someone updates a code page or character set on the host that was marked PUBLIC, you should run APSRMARK to change the marking of the host resources—for example, by entering new date and time values. Otherwise, the change will not be 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 have 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 134.

3. To read about the sample jobs that PSF supplies for marking resources, see Appendix A, APSRMARK Sample Jobs.
4. To use APSRMARK to identify resources for use by DPF, see Chapter 17, Marking Resources for Distributed Print Function.

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 165. For printers that support resident outline fonts, you can use APSRMARK to mark host outline fonts. If the printer is supported through WPM or DPF, however, you cannot use outline fonts.

## APSRMARK Considerations

Before marking fonts for resident use, consider the following:

- Font character sets containing only font metrics are shipped 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 shipped have been marked by use of the APSW4028 sample APSRMARK job. Code pages shipped with PSF have also been 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 on the print directions and character rotations supported, refer to *AFP: Printer Information*.
- 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.
- When using resident fonts, you may 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 above differences occurs, consider a couple of alternatives. One is to mark any altered fonts 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 has not been altered). The other is to provide a host version of the font that is identical with the printer-resident font being 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 will query the printer to determine whether the resident font is available. To determine which fonts each printer supports, refer to *AFP: Printer Information*.

**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 in the

240-pel library. Instead, create and mark a 300-pel version. For more information, see “Using APSRMARK” on page 137. 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 you want to use. For a list of the fonts resident on a particular printer, refer to *AFP: Printer Information*.
2. Determine the member names for both the code page and the font character set of the corresponding font in the system library. A list of code-page names and font character set names is given in *AFP: Printer Information*.

**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* using APSRMARK. APSRCF30 is a font-scaling utility; it is described in Appendix G, Font Conversion Program, 240-pel to 300-pel (APSRCF30).

3. Using APSRMARK, mark the chosen code page and font character set. For more information on marking the fonts, see “Syntax of APSRMARK Control Statements” and “APSRMARK Control Statement Parameters for Resident Fonts” on page 138.
4. Make sure that the application generating the print job references the font correctly, using the member names marked in Step 3.
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 a Resource Specified for a Print Job” on page 121.

## 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 should 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 shipped with PSF; see Appendix A, APSRMARK Sample Jobs. For specific values of RRID, use the following formulas:

- For the font character set:

$$\text{RRID} = \text{FGID} + \text{FW}$$

where

FGID = a font identifier of 4 hexadecimal digits

+ = concatenation

FW = a font width of 4 hexadecimal digits

Both FGID and FW must be specified as above, even if one or both fields contain only zeros or leading zeros.

- For the code page:

$$\text{RRID} = \text{GCSGID} + \text{CPGID}$$

where

GCSGID = a graphic character set identifier  
of 4 hexadecimal digits

+ = concatenation

CPGID = a code page identifier  
of 4 hexadecimal digits

Both GCSGID and CPGID must be specified as above, even if one or both fields contain only zeros or leading zeros.

Refer to *AFP: Printer Information* for information such as:

- FGID and FW values for fonts on the 3270 IDS diskette <sup>7</sup>
- GCSGID and CPGID values for fonts on the 3270 IDS diskette
- FGID and FW values for fonts on the DCF/GML diskette <sup>8</sup>
- GCSGID and CPGID values for fonts on the DCF/GML diskette
- FGID and FW values for fonts on the 4028 font cards
- GCSGID and CPGID values for fonts on the 4028 font cards

**RRDATE** Consists of any value of 5, 6, or 7 decimal digits.

$$\text{RRDATE} = \text{yyyyddd}$$

where

<sup>7</sup> 3270 Information Display Systems diskette

<sup>8</sup> Document Composition Facility/General Markup Language diskette



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. Figure 37 shows an example of marking resources.

Do not mark metric-only font character sets PRIVATE.

## Using Control Statements to Mark a Font

Figure 37 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 will be 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
/*
```

Figure 37. Sample JCL for Marking Resident Fonts. Your syntax may differ; for syntax rules, see “Syntax of APSRMARK Control Statements” on page 137.

## 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 has been updated or deleted since the alias was generated; in that case, the alias is not updated. An alias whose primary has been 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 have not been updated or deleted are marked. Aliases whose primaries have been updated or deleted are not marked. When resources are reported, only primary members are listed. When marking multiple fonts for printer-resident use, do not specify MEMBER=ALL; rather, specify each member name in multiple control statements. When you specify the REPORT parameter, however, you can specify MEMBER=ALL.

### **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 will be marked and will become 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 in order 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 40 on page 142 for an example of a short report.
- LONG specifies that a long report is generated. This is the default report format. See Appendix B, “APSRMARK Long Report Format” on page 273 for an example of a long report.

## Sample Control Statements for APSRMARK

In the following examples, lowercase letters represent information you specify to customize the JCL.

### Marking a Member with Resident Font Information

Figure 38 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=c1ass
//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
/*
```

Figure 38. Example of the Use of JCL Statements to Mark a Member of a Data Set. Your syntax may differ; for syntax rules, see “Syntax of APSRMARK Control Statements” on page 137.

Figure 38 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 will contain the member marked by APSRMARK.

**Note:** The data set attributes for OUT1 must match those 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 39 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
/*
```

Figure 39. Example of the Use of JCL Statements in Generating a Short Report. Your syntax may differ; for syntax rules, see "Syntax of APSRMARK Control Statements" on page 137.

Figure 39 contains the following JCL statements:

### THISDD DD

Defines the partitioned data set that contains the member that will be reported.

### SYSIN DD

Precedes the control statements that specify the parameters for the report to be printed.

Figure 40 is an example of a printed short report.

```
AP5580I          INDD=OUT2,MEMBER=ALL,REPORT
AP5580I CZB50C PUBLIC SAVE 2004-02-29 14:32:56 9763 661200354381 PSFDVT L1111111.L2222222.L3333333.L4444444.L5555555
AP5580I COMYNEW1 PUBLIC SAVE 2004-02-29 14:32:56 9763 661200354381 PSFDVT L1111111.L2222222.L3333333.L4444444.L5555555
AP5580I +00280090 77060 12121212
AP5580I COFACTRY PUBLIC SAVE PSFDVT L1111111.L2222222.L3333333.L4444444.L5555555
AP5580I CONOMARK ?????? ?????? ???-??-?? ??:??:?? ??? ????????????? ?????? ?
AP5580I FINOBOX ?????? ?????? ???-??-?? ??:??:?? ??? ????????????? ?????? ?
AP5580I MIUBLPK ?????? ?????? ???-??-?? ??:??:?? ??? ????????????? ?????? ?
AP5580I O1ROI000 PUBLIC SAVE 2004-02-29 14:32:56 CBE4 661200354381 PSFDVT LISZ.MKDRCS
AP5580I P1SEC5 ?????? ?????? ???-??-?? ??:??:?? ??? ????????????? ?????? ?
AP5580I S1SCHLAF PUBLIC SAVE 2004-02-29 14:32:56 47A5 661200354381 PSFDVT DOSCHE.PRIVATE.PSEGMENT.RESOURCE
AP5580I T1L000XN PUBLIC SAVE 2004-02-29 14:32:56 A335 661200354381 PSFDVT LISZ.MKDRCS
AP5580I TIMYNEW1 PUBLIC SAVE 2004-02-29 14:32:56 A335 661200354381 PSFDVT LISZ.MKDRCS
AP5580I +01330129 77060 12121212
AP5580I T1BASIC1 PUBLIC SAVE PSFDVT LISZ.MKDRCS
AP5580I XZH0FN ?????? ?????? ???-??-?? ??:??:?? ??? ????????????? ?????? ?
```

Figure 40. 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 41 on page 143 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
/*
```

Figure 41. Example of the Use of JCL Statements in Generating a Long Report. Your syntax may differ; for syntax rules, see “Syntax of APSRMARK Control Statements” on page 137.

Figure 41 contains the following JCL statements:

**THISDD DD**

Defines the partitioned data set that contains the member that will be reported.

**SYSIN DD**

Precedes the control statements that specify the parameters for the report to be printed.

See Appendix B, “APSRMARK Long Report Format” on page 273 for a description of a long report format.

## Diagnosing Problems with Printer-Resident Fonts

Table 13 summarizes some of the problems resulting from incorrect use of APSRMARK, and provides possible solutions.

| <i>Table 13 (Page 1 of 2). Resident Font Diagnosis</i>  |  |
|---|--|
| <b>Problem</b>  | <b>Possible solution</b>   |
| PSF loads the host library version of the font into the printer and does not use the printer-resident font. | <p>The code page, the font character set, or both are not marked correctly. See “Using APSRMARK” on page 137.</p> <p>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.</p> <p>Make sure that the marked resource is found by PSF before any unmarked version of the resource. For more information, refer to “Searching for a Resource Specified for a Print Job” on page 121.</p> |
| A printer-resident font is used, but not the one specified.   | <p>Some printer-resident fonts are simulated by the printer and may appear different from the version in the system library. For more information on font simulation done by your printer, refer to the publications for your printer.</p> <p>If a simulation is not the problem, make sure that the correct RRID values are specified. For more information on RRID values, see “APSRMARK Control Statement Parameters for Resident Fonts” on page 138.</p>   |

Table 13 (Page 2 of 2). Resident Font Diagnosis

| Problem  | Possible solution   |
|--|---|
| <p>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.</p> | <p>You may have specified a character rotation or print direction that your printer does not support. For information about the character rotations and print directions your printer supports, refer to the publications for your printer or to <i>AFP: Printer Information</i>.</p>   |
| <p>Printing results in message APS279I.</p>  | <p>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 identified in the message is marked correctly.</p> <p><b>Note:</b> 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.</p> <p>If the "4028 Font Metrics" character set has been marked correctly, make sure that the printer contains the desired resident font.</p> |
| <p>Printing results in message APS280I.</p>  | <p>If your installation is using 240-pel fonts to access 300-pel resident fonts, make sure that the font cartridge 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.</p> <p>Make sure that the host font library is a 300-pel version (contains 300-pel character sets). For more information on creating and marking a 300-pel font library, see "Using APSRMARK" on page 137.</p>  |

---

## Chapter 15. 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 on DPF dynamic caching, see Chapter 17, “Marking Resources for Distributed Print Function” on page 157.

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 only if it is marked PUBLIC and is loaded from a system library. Fonts marked PRIVATE are always downloaded, never captured. You can use the APSRMARK utility to mark a resource.

The following sections explain what needs to be done for different kinds of fonts.

---

### Single-Byte Raster Font Capture

For raster fonts that were previously shipped as a part of the PSF product, run the sample APSRMARK jobs that were shipped with PSF. For more information, see Appendix A, “APSRMARK Sample Jobs” on page 271.

The single-byte raster fonts shipped in the IBM AFP Font Collection 1.1 (Program Number 5648-113) or IBM AFP Font Collection 2.1 (Program Number 5648-B33) are already marked correctly both for activating resident fonts and for capturing fonts.

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. For information on using APSRMARK, see “PSF Resident-Font Utility—APSRMARK” on page 135.

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 drive, mark the font PRIVATE.

---

### Single-Byte Outline Font Capture

All single-byte outline fonts shipped in the IBM AFP Font Collection 1.1 (Program Number 5648-113) or IBM AFP Font Collection 2.1 (Program Number 5648-B33) are already marked correctly both for activating resident fonts and for capturing fonts.

Any single-byte outline fonts created or changed by the IBM Type Transformer program may 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.

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 drive, mark the font PRIVATE.

---

## Double-Byte Raster Font Capture

If you are migrating from a printer with resident fonts to a printer that captures fonts, the host libraries are already marked correctly.

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 on using APSRMARK, see “PSF Resident-Font Utility—APSRMARK” on page 135.

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 drive, mark the font PRIVATE.

---

## Double-Byte Outline Font Capture

The double-byte fonts in “IBM AFP CJK Metric-Only Fonts,” RPK 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.

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.



---

## Chapter 16. Using Outline Fonts

This chapter explains why IBM supports printers that use 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 to enable you 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<sup>9</sup> because outline font technology is resolution-independent.

The IBM AFP Font Collection (Program Number 5648-B33) contains a large selection of fonts in both outline and raster formats. (For descriptions of IBM's fonts, see the "Bibliography" on page 389.) 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 PSF Exit 7 (APSUX07) to map to an equivalent outline font, see "Considerations Related to Mapping to Outline Fonts" on page 209.

You can create fonts and add them to your font libraries by using the Type Transformer program, which is an optional feature of AFP Font Collection (Program Number 5648-B45). For information about creating new fonts, refer to *IBM AFP Fonts: Type Transformer User's Guide*, which is available online with the Type Transformer program.

---

### 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.

---

<sup>9</sup> For example, the 3900-0W1, 3900 Duplex, 3900 wide/duplex, 3935, and 3130 printers.

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 using 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 42. A raster font character, by contrast, is stored and used in its full, rasterized size.

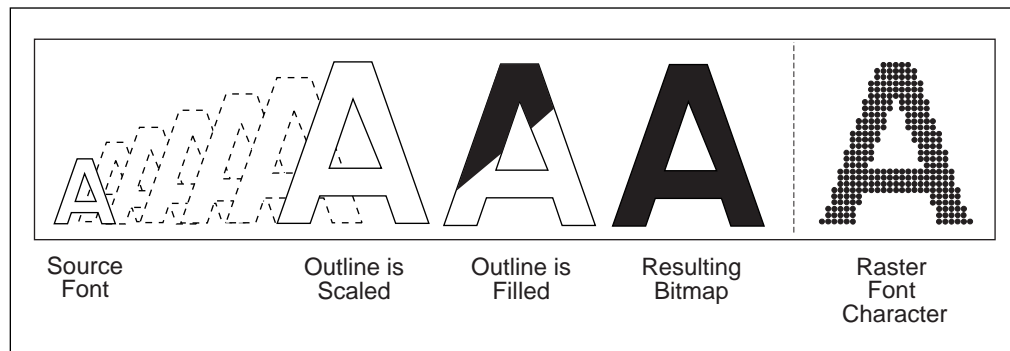


Figure 42. Comparison of an Outline Font Character with a Raster Font Character

## 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 37.

## Creating New Fonts

You can create new 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 new single-byte outline font:

1. Using Type Transformer, 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 will be routed to printers that do not support outline fonts, generate the FOCA raster equivalent of this font (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 MVS 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 below.
  6. Assemble the user mapping tables, and link-edit them with PSF.
  7. Make sure that the MVS 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 15, “Marking Fonts for Font Capture” on page 145.

---

## 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 should not be confused with font substitution (which is done by use of APSUX07). 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:

- When 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.
- When 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 XTP7MTOF (see next bullet). This process is called a *GRID map*.
  - When APSUX07 requests that PSF map fonts to outline fonts (XTP7MTOF = ON), and the attached printer supports outline fonts. This process is called an *outline map*.

The user does not select technology-compatibility mapping or GRID mapping; PSF performs these maps automatically.

The outline map function is an option that can be selected by using APSUX07. See “Considerations Related to Mapping to Outline Fonts” on page 209 for more information.

The different mapping functions are performed at different times during the processing of the font. The choice of mapping functions to be performed 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.

## Timing of Mapping Operations

The technology-compatibility map and the GRID map functions are performed before the resource-access call to APSUX07, if that call is enabled. This means that APSUX07 can be called with font names different from those specified in the job.

The outline map function happens after the resource-access call to APSUX07 and before the resource-load call to APSUX07, if those calls are enabled.

## Mapping Algorithms

PSF uses different processes to map one technology to another; the process 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 desired technology.

---

## 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 have added new 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 XTP7MTOF = ON in APSUX07, you can add an entry to the user mapping tables to disable the mapping of this particular font, so that it has to be printed as a raster font.

Appendix J, Mapping Tables for Outline Fonts, contains two PSF font mapping tables:

- APSRGTM, 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 265.

When performing the mapping functions, PSF searches the user tables first. A mapping in the user tables takes precedence over a similar mapping in the internal system tables.

Each time you modify the mapping tables, you must compile them and link-edit them with PSF.

## Adding a New Mapping of a Raster Font to an Outline Font

To add a new mapping of a raster font name to an outline font name:

1. Set XTP7MTOF=ON in APSUX07 to make sure that outline mapping is requested,.
2. Add a new 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*. For non-IBM supplied fonts or raster fonts created from OEM outlines using the IBM Type Transformer product, either determine the values that were used when the raster fonts were created or assign values yourself 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'.

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 have 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 will use the internal system tables. If you have created a specific code page for use with your character set, add a new 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 a new entry to the GRID-to-Font-Name Mapping table (APSRGTM), 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 item 1 above can be coded as follows:

```
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'CZMYOURL'  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 should use the same code page as the corresponding raster font character set entry in the Font-Name-to-GRID Mapping table (APSRNTGM). If you have 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 will use the internal system tables. If you have created a specific code page for use with your character set, add a new 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'T1MYCPG'   CODE PAGE NAME
```

## Adding a New Mapping of an Outline Font to a Raster Font

To add a new mapping of an outline font to a raster font:

1. Add a new 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'CZMYOURL'  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*. For non-IBM supplied fonts or outline fonts created from OEM outlines using the IBM Type Transformer product, either determine the values that were used when the raster fonts were created or assign values yourself 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 have 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 will use the internal system tables. If you have created a specific code page for use with your character set, add a new 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 a new 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 non-zero font width, this entry is also used for GRID-to-raster font mapping.

This raster font should use the same code page as the corresponding raster font character set entry in the Font-Name-to-GRID Mapping table (APSRNTGM). If you have 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 will use the internal system tables. If you have created a specific code page for use with your character set, add a new 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, do the following:

1. Add an entry to the Font-Name-to-GRID Mapping table (APSRNTGM), to associate 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 New Mapping of a GRID to a Raster Font

If you have added a new mapping for an outline font to this raster font character set, you already have 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 the following:

1. Add a new 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*. For non-IBM supplied fonts or outline fonts created from OEM outlines using the IBM Type Transformer product, either determine the values that were used when the raster fonts were created or assign values yourself 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 may also specify a vertical font size.

If you have 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 will use the internal system tables. If you have created a specific code page for use with your character set, add a new 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 New Mapping of a GRID to an Outline Font

If you have added a new mapping for a raster font to this outline font character set, you already have 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 the following:

1. Add a new 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*. For non-IBM supplied fonts or raster fonts created from OEM outlines using the IBM Type Transformer product, either determine the values that were used when the raster fonts were created or assign values yourself 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 have 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 will use the internal system tables. If you have created a specific code page for use with your character set, add a new 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 17. Marking Resources for Distributed Print Function

Distributed Print Function (DPF), a part of the PSF/2 program, runs on an IBM PS/2 personal computer. DPF makes it possible to store PSF resources—fonts, overlays, and page segments—in the fixed-disk storage of the PS/2 personal computer. As Figure 43 shows, these stored resources can then be used by a printer that is attached to the PS/2 personal computer.

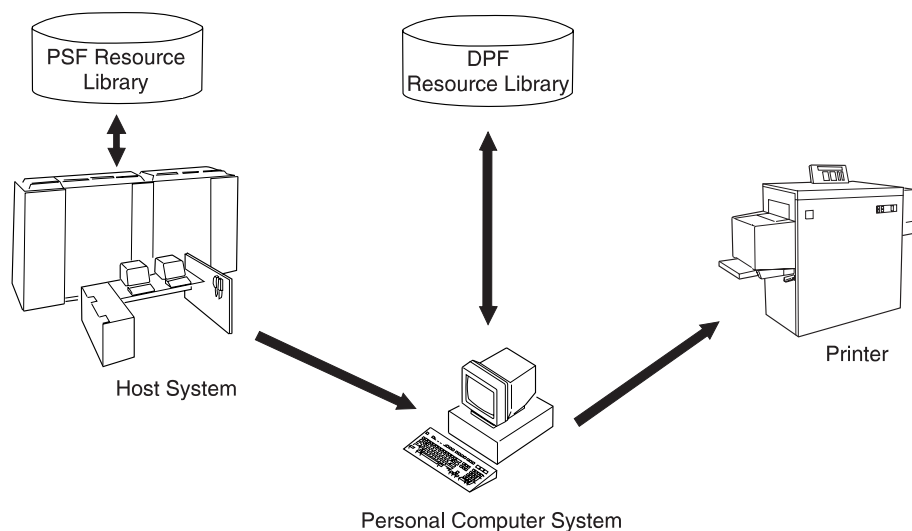


Figure 43. Access to Resources by DPF

**Note:** The PSF Direct function of PSF/2 and PSF for AIX does *not* store resources in the intermediate device.

If requested, DPF accesses these resources directly from the fixed disk and loads them into the printer at System/390 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 stored in the PSF resource library. This chapter describes the JCL statements that are used to invoke the APSRMARK utility.

For more information about DPF, refer to the publications for PSF/2 in the “Bibliography” on page 389.

**Note:** You can mark a resource just 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, using the APSRMARK utility.

APSRMARK is a batch program invoked 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 14, Supporting Resident Fonts.
3. You can use APSRMARK to identify the same resources for use by DPF and printer-resident fonts. With DPF, however, 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 165
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 having 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 through 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 should 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 has been updated or deleted after the alias was generated, in which case the alias is not updated. An alias whose primary has been 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 have not been updated or deleted are marked. Aliases whose primaries have been updated or deleted are not marked. When resources are being reported, only primary members are listed. When 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 will be marked and will become 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 just 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 48 on page 164 for an example of a short report.
- LONG specifies that a long report is generated. This is the default report format. See Appendix B, “APSRMARK Long Report Format” on page 273 for an example of a long report.

## **Examples of Control Statements for APSRMARK**

This section 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 A, APSRMARK Sample Jobs.

### **Creating a PSF Resource Library with All Members Marked PUBLIC**

Figure 44 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 44. Sample JCL for Creating a PSF Resource Library and Marking all Members PUBLIC

The sample JCL in Figure 44 uses the following statements:

**IN1 DD**

Defines the input partitioned data set that contains the members to be marked by APSRMARK.

**OUT1 DD**

Creates a new output partitioned data set that contains the PUBLIC members.

**Note:** The data set attributes for OUT1 must match those for IN1.

**SYSIN DD \***

Precedes the control statements specifying 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 45 shows the APSRMARK control statements for creating a partitioned data set that contains only PUBLIC members and for then 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 45. Sample JCL for Creating a PSF Resource Library and Marking Some Members PRIVATE

The sample JCL in Figure 45 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 a new output partitioned data set that contains the PUBLIC members.

**Note:** The data set attributes for INOUT1 must match those for IN1.

**SYSIN DD \***

Precedes the control statements specifying 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 specifying that members *name1* and *name2* are marked PRIVATE and replaced in INOUT1.

**Modifying an Existing PSF Resource Library**

Figure 46 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 46. Sample JCL for Modifying an Existing PSF Resource Library

The JCL in Figure 46 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 47 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=c1ass
//IN1 DD UNIT=dasd,DISP=SHR,DSN=dsname,VOL=SER=volser
//SYSIN DD *
        INDD=IN1,MEMBER=ALL,REPORT,SHORT
/*
```

Figure 47. Sample JCL for Reporting Resource Markings in a Partitioned Data Set

The JCL in Figure 47 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.

Figure 48 on page 164 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 142.

|         |                           |        |      |          |          |      |                |        |                  |
|---------|---------------------------|--------|------|----------|----------|------|----------------|--------|------------------|
| 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 | C0D0G112                  | 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 | EAE3 | 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 | C0S0C110                  | 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 | C0S0E1TR                  | 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 | C0S0OB10                  | PUBLIC | SAVE | 89-07-21 | 12:50:56 | 9453 | 660241123081   | DSNAME | VOLSER1.INFONTS  |
| APS580I | C0S0OR10                  | 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 48. 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 Figure 48.

## 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 those 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 invoked, as part of the page data. You cannot store these page segments in the DPF library.

In a line-format print data set, a page segment included by an 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/370, 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.<sup>10</sup>

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 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 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.<sup>10</sup>

---

## 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. You cannot, however, 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:

- DPF only
- DPF and printer-resident fonts
- Neither of the above (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

---

<sup>10</sup> APSRMARK adds an appropriate MPS structured field for each soft page segment in an overlay that is marked for use with DPF.

## Changing How Fonts Are Marked

You may want to use the APSRMARK utility to change marks for:

- Fonts already marked for use by DPF only
- Fonts 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 14, Supporting Resident Fonts. If you want to mark the font PRIVATE, follow the procedures described in this chapter 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 described in Chapter 14, Supporting Resident Fonts, and use a value of 00000000 in the RRID keyword instead of the values given in *AFP: Printer Information*.

If you want to mark the font PRIVATE, follow the procedures described in Chapter 14, Supporting Resident Fonts, 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 14, Supporting Resident Fonts. If you want to mark the font for only DPF, follow the procedures described earlier in this chapter.

---

## Chapter 18. Using Installation Exits

This chapter describes how to use the installation exits supplied with PSF. The information in this chapter documents intended Programming Interfaces that let the customer write programs to obtain the services of PSF for OS/390.

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.

### Notes:

1. PSF supports two exits (APSUX14 and APSUX16) for direct-printing mode and one exit (APSUX15) for Download for OS/390. The remaining exits are for use when PSF is running in deferred-printing mode.
2. For OS/390 Softcopy Print (full-function PSF is not installed), three exits are supported: APSUX01, APSUX02, and APSUX03.

---

## Do's and Don'ts

Code processing in exit routines is an integral part of PSF and can affect performance. Keep the following points in mind when using exit routines:

- Exit routines should be coded by experienced programmers who are aware of overall system implications.
- Exit control blocks are provided in assembler language.
- Use only the common control blocks passed to this exit. For a description of the common control blocks, see "Common Control Blocks" on page 170.
- 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, which are passed to the PSF exit below the 16 MB line.
- The exit code should reference a field in a common control block only by the name associated with that field in the DSECT 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 APSUX01 and APSUX02 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 182.) 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 let this 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 invoke selected exits only in those address spaces in which they are needed. The data sets on the STEPLIB statement must be APF-authorized. (The STEPLIB statement is highlighted in the example in Figure 49.)

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=2048K
//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=(60DB),
//  PIMSG=YES,
//  TRACE=YES,
//  DATAACK=BLOCK
//PRINTR1 ENDCNTL
```

Figure 49. Use of STEPLIB Statement for Exit Routine in PSF Startup Procedure

## Changing Exit Routines

During PSF installation, the source code (in assembler language) for all the exit routines is installed into SYS1.SAMPLIB. 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 14 lists these routines and summarizes their functions.

| <i>Table 14 (Page 1 of 2). PSF Exit Routines in SYS1.SAMPLIB</i> |  |
|--|--|
| <b>Sample Routine</b>  | <b>Description</b>   |
| APSUX01  | Source routine for default job headers. See Figure 24 on page 76 for an example.   |
| APSUX01P   | Source routine for job headers that print faster on communication-attached printers, except the 4224 and 4234 printers. See Figure 25 on page 77 for an example.   |
| APSUX01S   | Source routine for job headers that use guaranteed print labeling (described in <i>Print Services Facility: Security Guide</i> ).  |
| APSUX02  | Source routine for default job trailers.   |
| APSUX02P   | Source routine for job trailers that print faster on communication-attached printers, with the exception of the 4224 and 4234 printers.  |
| APSUX02S   | Source routine for job trailers that use guaranteed print labeling (described in <i>Print Services Facility: Security Guide</i> ).   |
| APSUX03  | Source routine for default data set separators.  |
| APSUX04X   | Sample (skeleton) source routine for processing line-data records.   |
| APSUX04Y   | Sample (skeleton) source routine for processing structured fields.   |
| APSUX05X   | Sample (skeleton) source routine for modifying SMF type 6 record processing.   |
| APSUX06X   | Sample (skeleton) source routine for modifying PSF message processing.   |
| APSUX07S   | Sample source routine for loading all page definitions and form definitions from DASD with each use.   |
| APSUX07T   | Sample source routine for mapping a FORMS= value to a PAGEDEF and FORMDEF pair.  |
| APSUX07U   | Sample source routine for altering the member-name prefix of resource objects (permitting use of objects with “nonstandard prefixes” for production or secure work).   |
| APSUX07V   | Sample source routine for indicating how PSF can perform many of the functions that are requested on the initialization and begin-data-set calls.  |
| APSUX07W   | Sample source routine for indicating how PSF can perform many of the functions that are requested on the resource access and end-data-set calls.   |
| APSUX07X   | Sample (skeleton) source routine explaining exit structure.  |
| APSUX07Y   | Sample (skeleton) source routine demonstrating: <ul style="list-style-type: none"> <li>• Exit involvement, based on printer name and type</li> <li>• Substitution of a page definition or a page segment</li> <li>• Issuance of a message for page segment substitutions</li> <li>• Retention of an overlay over data set boundaries.</li> </ul> |

Table 14 (Page 2 of 2). PSF Exit Routines in SYS1.SAMPLIB

| Sample Routine | Description   |
|----------------|---|
| APSUX07Z       | Sample (skeleton) source routine demonstrating: <ul style="list-style-type: none"> <li>• Exit involvement based on printer name and type</li> <li>• Substitution of a job-header form definition.</li> </ul>                  |
| APSUX14X       | Sample source routine for invoking 3800 Line-Mode Conversion in direct-printing mode.   |
| APSUX15X       | Sample source routine to provide the printer device name and the output group name for Download for OS/390 to transmit to its server on AIX. This member is only installed in SYS1.SAMPLIB if Download for OS/390 is present. |
| APSUX16X       | Sample (skeleton) source routine for the direct printing exit.  |

**Note:** PSF provides object code for the default separator-page exits (modules APSUX01, APSUX02, and APSUX03). PSF does not provide object code for the other exits—logical-record processing, SMF type 6 record processing, PSF-message processing, resource processing, direct printing 3800 Line-Mode Conversion, Download for OS/390, and direct printing.

During PSF initialization, the exit subcomponent issues a BLDL request to locate any user-provided exit routines prefixed with APSUX $nn$  (where  $nn$  is replaced with the exit numbers: 01, 02, 03, 04, 05, 06, 07, 14, 15, 16). The exit routines may be in a data set pointed to in a STEPLIB DD statement in the PSF startup procedure, or they may 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 re-enterable and must be link-edited with the RENT,LET attributes.
2. Exit routines must reside in an authorized library.
3. Code for installation exits can be written in 24-bit or 31-bit addressing mode.
4. PSF-program maintenance may require the reassembly of installation exits.
5. All exit routines 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) should be placed in the object code, starting at the sixth byte. If a dump is run, the module name is printed in the dump. (ESTAE is in effect during the execution of exits.)

## Common Control Blocks

When PSF calls an exit routine, register 1 points to a 1-word parameter list that contains the address of the APSGEXTP area. This area contains the address of the exit-communication area (APSUECA), pointers to JSPA, and pointers to the input records associated with exits APSUX04, APSUX06, APSUX07, and APSUX16. When the exit routine returns to PSF, PSF does not check register 15 for exits.



## APSGEXTP Area

The format of APSGEXTP is shown in Figure 50. Values in parentheses are in hexadecimal.

| 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      | XTPCAP   | 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.   |
| 16   | (10) Character | 4      | *        | Reserved.  |
| =====  |                |        |          |  |
| 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. |                |        |          |  |
| =====  |                |        |          |  |
| 20   | (14) Address   | 4      | XTPRIXP  | Pointer to record table index.   |
| 24   | (18) Unsigned  | 1      | XTPPIND  | Processing indicator.  |
| 25   | (19) Bitstring | 1      | XTP4FLAG | APSUX04 or APSUX14 flags.  |
|  | 1... ....      |        | XTP4NTY  | PSF will send end-of-transmission (EOT) indicator to APSUX04.  |
|  | .1.. ....      |        | XTP4NEOT | Normal EOT has occurred.   |
|  | ..1. ....      |        | XTP4ABNE | Abnormal EOT has occurred.   |
|  | ...1 ....      |        | XTP4NCAL | Call APSUX04 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.   |

Figure 50 (Part 1 of 3). Format of the APSGEXTP Area

|  |                |   |          |  |
|--|----------------|---|----------|--|
|  | .... .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). |
|  | .... ..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."<br>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 | APSUX05 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).<br>0 = Not specified (defaults to formdef value)<br>1 = Simplex<br>2 = Normal duplex<br>3 = Tumble duplex<br>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) Character  | 8 | XTP6PROC | FSA identifier:<br>channel-attached printers:<br>DDnnn, where nnn is the device-address<br>non channel-attached printers:<br>label of PRINTDEV statement   |
| 8  | (8) Character  | 8 | XTP6MSG  | Message ID (APSXXXXI).   |

Figure 50 (Part 2 of 3). Format of the APSGEXTP Area

|    |                |   |          |   |
|----|----------------|---|----------|---|
| 16 | (10) Character | 1 | XTP6TYPE | Message TYPE.<br>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) Character | 1 | XTP6FLG1 | APSUX06 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.   |
|    | ..11 1111      |   | *        | Reserved.   |
| 18 | (12) 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.   |
| 19 | (13) Character | 5 | *        | Reserved.   |

Figure 50 (Part 3 of 3). Format of the APSGEXTP Area

For information on the contents of XTP7, see Figure 67 on page 296.

### Exit-Communication Areas: APSUECA and APSUECE

APSGEXTP provides addressability to the exit-communication area (APSUECA, or simply ECA) and to its extension (APSUECE). These areas provide the interface among the exit routines.

The format of APSUECA is shown in Figure 51. Values 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.                               |

Figure 51 (Part 1 of 2). Format of Exit-Communication Area, APSUECA

|     |           |           |    |          |  |
|-----|-----------|-----------|----|----------|--|
| 36  | (24)      | Address   | 4  | ECAECP   | 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.<br>0 = job trailers may be marked.<br>1 = do not mark job trailers.   |
|     | .1.. ..   |           |    | ECALSTDS | Last or only data set.<br>0 = Not the last or only data set of the job.<br>1 = Last or only data set of job.   |
|     | ..11 1111 |           | *  |          | Reserved.  |
| 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)      | Bitstring | 1  | *        | Reserved.  |
| 66  | (42)      | Signed    | 2  | ECATRNC  | Transmission count.  |
| 68  | (44)      | Bitstring | 1  | ECAFLAGS | Exit-control flags.  |
|     | 1... ..   |           |    | ECASLANT | Slant letters requested.   |
|     | .1.. .... |           |    | ECALJUST | Left justify 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 | APSUX06 (message) exit save area.  |
| 368 | (170)     | Character | 72 | ECARSCSV | APSUX07 (resource) exit save area.   |
| 440 | (1B8)     | 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 minimum length of the work buffer is 1599 bytes. |

Figure 51 (Part 2 of 2). Format of Exit-Communication Area, APSUECA

If you do not want PSF to generate separator pages, delete or modify the separator-page exits. For information on how to do this, see "Programming with Separator-Page Exits" on page 180.

The format of APSUECE is shown in Figure 52 on page 175. Values 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, 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.  |
|         | .... 1111      |        | *        | 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 pages in job.   |
| 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 (does not include header or trailer page records).                                    |

Figure 52 (Part 1 of 2). Format of Exit-Communication Area Extension, APSUECE

|     |      |           |    |          |  |
|-----|------|-----------|----|----------|--|
| 136 | (88) | Signed    | 4  | ECELPCNT | Logical page count (includes header and trailer pages).  |
| 140 | (8C) | Signed    | 4  | ECEPPCNT | Physical page count (includes header and trailer pages). |
| 144 | (90) | Character | 2  | *        | Reserved.  |
| 146 | (92) | Signed    | 2  | ECETXMS  | Total transmission required.                             |
| 152 | (98) | Address   | 4  | ECEX02P  | Address of exit APSUX02.                                 |
| 156 | (9C) | Address   | 4  | ECEX03P  | Address of exit APSUX03.                                 |
| 160 | (A0) | Address   | 4  | ECEX04P  | Address of exit APSUX04.                                 |
| 164 | (A4) | Address   | 4  | ECEX05P  | Address of exit APSUX05.                                 |
| 168 | (A8) | Address   | 4  | ECEX06P  | Address of exit APSUX06.                                 |
| 172 | (AC) | Address   | 4  | ECEX07P  | Address of exit APSUX07.                                 |
| 176 | (B0) | Address   | 4  | ECEX08P  | Reserved.  |
| 180 | (B4) | Address   | 4  | ECEX09P  | Reserved.  |
| 184 | (B8) | Address   | 4  | ECEX10P  | Reserved.  |
| 188 | (BC) | Address   | 4  | ECEX11P  | Reserved.  |
| 192 | (C0) | Address   | 4  | ECEX12P  | Reserved.  |
| 196 | (C4) | Address   | 4  | ECEX13P  | Reserved.  |
| 200 | (C8) | Address   | 4  | ECEX14P  | Address of exit APSUX14 (in direct print mode only).     |
| 204 | (CC) | Address   | 4  | ECEX15P  | Address of exit APSUX15 (in Download for OS/390 only).   |
| 208 | (D0) | Address   | 4  | ECEX16P  | Address of exit APSUX16 (in direct print mode only).     |
| 212 | (D4) | Signed    | 4  | ECEIMPS  | Impressions (sides) printed for the data set.            |
| 216 | (D8) | Character | 12 | ECEDSID  | FSI GETDS data set ID (in Download for OS/390 only).     |
| 228 | (E4) | Address   | 4  | ECELMO   | Reserved for LM0.  |
| 232 | (E8) | Character | 32 | *        | Reserved.  |

Figure 52 (Part 2 of 2). Format of Exit-Communication Area Extension, APSUECE

If repositioning is required, PSF can enter the exit routines for header pages (APSUX01 and APSUX03) 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 APSUX01 or exit APSUX03, PSF sets ECATRNC to the current transmission count, which can be used to determine whether the exit has been reentered. An installation exit should 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.

**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 terminated.

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.

**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, ECEPRDT matches ECEPRTT, and ECEPRTMD matches ECEPRTM.   |
| <b>ECEPRTMD</b> | 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, ECEPRDT matches ECEPRTT, and ECEPRTMD matches ECEPRTM.   |
| <b>ECEPRTT</b>  | 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 supported are identical.  |
| <b>ECEPRTM</b>  | 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 supported are identical.  |
| <b>ECEOUTTK</b> | <p>The scheduler work block (SWB) token for the OUTPUT JCL statement of the print data set.</p> <p>PSF uses ECEOUTTK to retrieve the keywords specified in the OUTPUT JCL statement of the print data set.</p> <p>JES passes this token to PSF in the functional subsystem interface (FSA) GETDS parameter list (IAZFSIP).</p> <p>For more information on SWBs and how to use the SWB token to retrieve keywords, refer to the <i>Using the Functional Subsystem Interface</i> for your operating system, listed in the “Bibliography” on page 389.</p> <p><b>Note:</b> The SWB contains JCL values merged from the DD statement, the OUTPUT statement, and the JES defaults according to the resource hierarchy. JES merges the values before allocating the print data set to PSF. When the JCL values are returned, therefore, there is no way to determine whether a particular value was specified on the print data set or defaulted to by JES.</p> |
| <b>ECETUPTR</b> | <p>The address of the SWBTU for the print data set’s OUTPUT JCL statement.</p> <p>You can use the SWBTUREQ macro to obtain the JCL parameters specified for the print data set. The SWBTU address is input to the SWBTUREQ macro. ECETULEN contains the SWBTU length.</p> <p>The separator exits supplied by PSF contain an example of how to invoke the SWBTUREQ macro.</p> <p>For more information about the SWBTUREQ macro, refer to the <i>Authorized Assembler Services Reference</i> for your operating system, listed in the “Bibliography” on page 389.</p> <p><b>Note:</b> The SWBTU contains JCL values merged from the DD statement, the OUTPUT statement, and the JES defaults according to the resource hierarchy. JES merges these values before allocating the print data set to PSF. When the JCL values are returned, therefore, there is no way to</p>  |

determine whether a particular value was specified on the print data set or defaulted to by JES.

**ECEWKPTR** The address of a local work area for the PSF 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. If those exits are used, therefore, this area should not be used to retain data across exit invocations.

---

## Sequence of Exit Invocation

Following is a typical flow of exit control in the sequence of exit interpretation:

1. APSUX01: Job-header page
2. APSUX03: Data-set separator page
3. APSUX04: Examine first spool record
4. APSUX04: Examine next spool record (until end of data set)
5. APSUX04: End-of-data-set call
6. APSUX02: Job-trailer page
7. APSUX05: Write SMF type 6 record

**Note:** After the trailer page prints, calls to APSUX05 are asynchronous. Calls to APSUX06 may be asynchronous also.

## Control Flow with APSUX07

Following is a typical flow of exit control that includes the use of APSUX07 for resource management.

1. APSUX07: Begin-data-set call
2. APSUX01: Job-header page
3. APSUX07: Access and load calls for resources used on the job-header page
4. APSUX03: Data-set separator page
5. APSUX07: Access and load calls for resources used on the data set separator page
6. APSUX04: Examine first spool record
7. APSUX07: Resource-access call for page definition
8. APSUX07: Resource-access call for form definition
9. APSUX07: Access and load calls for resources identified in the page definition.
10. APSUX07: Access and load calls for resources identified in the form definition.
11. APSUX04: Examine next spool record (until end of data set)
12. APSUX07: Access and load calls for resources identified in the input data stream
13. APSUX04: End-of-data-set call
14. APSUX02: Job-trailer page



15. APSUX07: Access and load calls for resources used on the job-trailer page
16. APSUX07: Resource-delete calls
17. APSUX05: Write SMF type 6 record

**Note:** After the trailer page prints, calls to APSUX05 are asynchronous. Calls to APSUX06 may be asynchronous, also.

---

## Separator-Page Exits: APSUX01, APSUX02, and APSUX03

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 24 on page 76, Figure 26 on page 78, and Figure 27 on page 79, 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:

|                |   |
|----------------|---|
| <b>APSUX01</b> | Job-header pages (including job-continuation pages) |
| <b>APSUX02</b> | Job-trailer pages                                   |
| <b>APSUX03</b> | Data-set-header pages                               |

PSF also provides assembler-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, listed in Table 14 on page 169, run in the same address space as PSF.

## Input for Separator-Page Exits

When a separator-page exit is entered, it contains the following:

|                    |   |
|--------------------|---|
| <b>Register 1</b>  | Pointer to the address of parameter area APSGEXTP |
| <b>Register 13</b> | Pointer to caller’s save area                     |
| <b>Register 14</b> | Return address                                    |
| <b>Register 15</b> | Entry-point address                               |

## Output for Separator-Page Exits

When control returns from the exit routine to PSF, the contents of the registers should be the same as they were at entry to the exit routine. As is explained in “Service Routines” on page 185, 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.

## 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 83.

## Programming with Separator-Page Exits

The following 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 53 on page 182 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 185.
- Before calling 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 185.

If you use an optional separator page, the job-separator-page OUTPUT statement in the PSF startup procedure should specify a page definition having a page size no larger than  $7\frac{1}{8} \times 7$  inches and a font no larger than 12 pitch.<sup>11</sup>

**Note:** A 12-pitch font produces text that is centered on the separator page.

- To suppress job headers and job trailers, you can delete or modify the separator exits. If some of the PSF-controlled printers on the system are not attached through DPF, you can add code to the separator exits. Place the following in the PROLOG section of the code, after the "SAVE CALLERS SAVE AREA ADDR":

|       |                |                       |
|-------|----------------|-----------------------|
| USING | APSUECE,@02    | Get addressability    |
| L     | @02,ECAECP     | to ECE                |
| TM    | ECEFLAG,ECEDPF | DPF-attached printer? |
| BO    | OUT            | Yes--exit immediately |
| DROP  | @02            | Drop address to ECE   |

---

<sup>11</sup> 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.

- As is explained in “Exit Table, APSUCOM” on page 182, the APSXUCOM macro provides a table for converting the Julian date to a year/month/day format (for example, 87 Jun 03).
- As is explained in “Block-Letter Tables, APSUBTBL” on page 184, 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
- JSPJSDDD for data set DD name
- JSPJSOCL for SYSOUT class
- JSPJPRI0 for data set priority
- JSPJDSPN for procedure name

## JES Job-Separator-Page Areas—IAZJSPA and IAZJSEP

IAZJSPA 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 may be available in a JES exit.

A pointer to the modified IAZJSPA is passed to the PSF separator-page exit. To modify IAZJSP in JES2, use exit 23; in JES3, use exit IATUX45.

Figure 53 on page 182 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      | JSPJPRI0 | Data set priority                    |
| 112 (70) | 4      | JSPAUSR1 | Reserved for user                    |
| 116 (74) | 4      | JSPAUSR2 | Reserved for user                    |

Figure 53. 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 macro. For more information, refer to your JES publications.

Figure 54 shows the format of a job-separator-page area extension, IAZJSEP. Values inside parentheses are in hexadecimal.

| Offsets  | Length | Name     | Description                 |
|----------|--------|----------|-----------------------------|
| 0 (0)    | 2      | JSPXNUM  | Number of extensions        |
| 2 (2)    | 2      | JSPXLEN  | 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 54. Format of JES Job-Separator-Page Area Extension, IAZJSEP

## 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 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 185, 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 three characters. With default values underscored, the syntax for the APSXUCOM macro is as follows:

```
[SYMBOL] APSXUCOM [BUFF=nnnnnn|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= {nnnnnn|**1599**}**

The size of the work buffer, ECAWKBUF, in the exit-communications area, APSUECA. *nnnnnn* is a decimal number between 1599 and 1048576. (For more information on APSUECA, see Figure 51 on page 173.)

**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 will 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 assembler 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 APSUX01, APSUX02, or APSUX03 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 two service routines: a block-letter routine, APSUBLK and an exit-put routine, APSUPUT. The exit-communication area, APSUECA, provides input to these service routines.

### 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-justified. 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 24 on page 76 and Figure 25 on page 77.)

The print lines 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 the following:

##### General Register 5

The address of the exit-communication area, APSUECA

##### ECABLKIN

The name (ranging from one to eight characters) to be used by the block letters

##### ECALJUST

If set on, used to left-justify 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 leaving the service routine, APSUBLK passes the block letters to the exit in the work buffer, ECAWKBUF. The format of the output in this buffer is shown in Figure 55 on page 186.

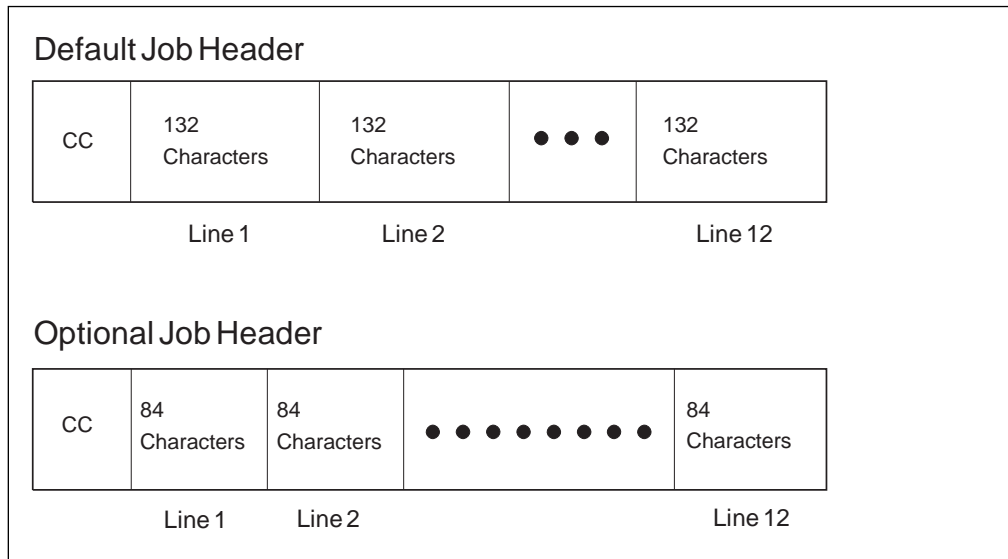


Figure 55. Format of the Output in the Work Buffer, ECAWKBUF

## Programming Considerations for APSUBLK

The service routine produces corresponding block characters from the tables in APSUBTBL.

The separator routines supplied by PSF do not generate slanted block letters. You can, however, set the ECASLANT field in APSUECA to produce block letters that are slanted to the right. For the location of this field, see Figure 51 on page 173.

The separator routines 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.

APSUBLK saves the caller's registers in the area addressed by register 13. When register 5 is entered, it must provide APSUBLK with the address of APSUECA. APSUBLK uses ECAGSAVE as its own save area. When control is returned from APSUBLK to PSF, register 15 is reset to 0.

## 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 should not modify the ECARCODE field.



When APSUPUT is entered, register 15 is reset. If an error occurs during a record PUT, 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 the following:

|                           |   |
|---------------------------|---|
| <b>General Register 5</b> | The address of the exit-communications area (APSUECA).  |
| <b>APSUECA</b>            | Before running APSUPUT, the following fields must be set: <ul style="list-style-type: none"><li>• The address of the record in ECARECAD.</li><li>• The record length in ECARECLN.</li><li>• 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, shown in Figure 57 on page 193.</li></ul> |

### Output for APSUPUT

Upon leaving the service routine, APSUPUT passes the ECARCODE and the contents of register 15.

---

## Logical-Record Processing Exit: APSUX04

PSF provides an exit you can use to create an APSUX04 routine for modifying records in the printed output, adding records, and suppressing records. This routine runs in the same address space as does PSF.

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 (for processing line data)
- APSUX04Y (for processing data already composed into pages)

For more information about line-data records, refer to *PSF for OS/390: User's Guide*.

For more information about structured fields composing page data, refer to *Mixed Object Document Content Architecture Reference*.

### Input for APSUX04

When the logical-record processing exit is entered, the contents of the registers are as follows:

|                    |   |
|--------------------|---|
| <b>Register 1</b>  | Pointer to the address of parameter area APSGEXTP |
| <b>Register 13</b> | Pointer to caller's save area                     |
| <b>Register 14</b> | Return address                                    |
| <b>Register 15</b> | Entry-point address                               |

For a logical record call:

|                |   |
|----------------|---|
| <b>XTPRIXP</b> | 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 04 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 APSUX04

**XTPPIND** The exit may 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 may or may not include the original print record.

The APSGEXTP macro, shown in Figure 50 on page 171, provides the values used to set XTPPIND.

**XTP4NCAL** Call Exit 04 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 APSUX04

The following considerations apply to APSUX04:

- When APSUX04 receives control, no processing has occurred. When APSUX04 sees the first record of a new data set, no resource-access processing has taken place. For the first record, therefore, APSUX04 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 182 for information on expanding ECAWKBUF. If you expand ECAWKBUF, PSF does not use the additional storage, which exceeds the APSUECA default size, between successive exits. The PSF exits can therefore 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 the following:

**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 additional 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 57 on page 193.

**Note:** When split records are processed, and the record is not changed, the default X'00' should be 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 appropriately.
  - 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 appropriately. If the data set does not use line merge, this flag is ignored.
- If duplexing is desired, 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 be 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 APSUX04, 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 APSUX04 is entered. XTPRECP points to the beginning of the entire record.

- An index table must be built, 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 57 on page 193. Values inside parentheses are in hexadecimal.

**Note:** IAZIDX is an MVS control block, not a PSF control block.

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 being 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 containing 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 182 for information on expanding ECAWKBUF. If you expand ECAWKBUF, PSF does not use the additional storage (that exceeds the APSUECA default size) between successive calls. The PSF exits can therefore 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 has been 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 would have to return 'DEF' to PSF. If this was not done, data could be lost.

- For APSUX04, PSF provides the end-of-transmission (EOT) indicators (first three bits in Figure 56 on page 192) and the 3800 Line-Mode Conversion indicators in field XTP4FLAG in APSGEXTP, as is shown in Figure 56.

| Bit       | Field    | Meaning  |
|-----------|----------|--|
| 1... ..   | XTP4NTY  | APSUX04 sets this bit to request an EOT notification from PSF. For notification to be received, this bit must be on every time APSUX04 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). APSUX04 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). APSUX04 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 APSUX04 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. APSUX04 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 APSUX04/APSUX14 is called for the first record of the transmission. APSUX04/APSUX14 may set this bit ON for the first record of the transmission to request that conversion for 3800 line-mode printing be performed. If APSUX04 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 APSUX04/APSUX14 is called for the first record of the transmission; this causes portrait pages to be generated. APSUX04/APSUX14 can set this bit ON for the first record of the transmission to request that landscape pages be generated. If APSUX04 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 APSUX04/APSUX14 is called for the first transmission. APSUX04/APSUX14 can set this bit ON for the first record of the transmission to request that line merge be activated. If APSUX04 sets this bit ON for any other records of the transmission, the bit is ignored.   |

Figure 56. APSUX04/APSUX14 Indicators in XTP4FLAG Field

In Figure 57 on page 193, the first 12 bytes represent a fixed-length header. The information starting at field ADXENTRL at offset 12 (X'0C') is repeated for each

index entry. This control block is mapped by macro IAZIDX. For more information, refer to 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 57. Format of JES Index Table, IAZIDX

## SMF Type 6 Record Exit: APSUX05

You can use this exit to modify system management facilities (SMF) type 6 records. SMF is an optional control program that provides a way for you to collect information to aid in evaluating system use. If you have not provided an APSUX05 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 Table 14 on page 169.) This routine runs in the same address space as PSF.

For a description of the SMF type 6 records that PSF creates, see “System Management Facility Type 6 Records” on page 86. For detailed information about SMF type 6 records, refer to *OS/390 MVS System Management Facilities (SMF)*.

## Input for APSUX05

When the SMF type 6 record-processing exit is entered, the contents of the registers are as follows:

|                    |   |
|--------------------|---|
| <b>Register 1</b>  | Pointer to the address of parameter area APSGEXTP |
| <b>Register 13</b> | Pointer to caller's save area                     |
| <b>Register 14</b> | Return address                                    |
| <b>Register 15</b> | Entry-point address                               |
| <b>XTP5NEWS</b>    | JESNEWS data set indicator                        |
| <b>XTPRECP</b>     | Pointer to SMF type 6 record                      |
| <b>XTPRECL</b>     | Length of record                                  |

## Output for APSUX05

|                |  |
|----------------|--|
| <b>XTPPIND</b> | The exit may set the XTPPIND indicator to: |
| <b>X'00'</b>   | Write the SMF type 6 record.               |
| <b>X'01'</b>   | Skip the SMF type 6 record.                |

## Programming Considerations for APSUX05

The following considerations apply to APSUX05:

- When APSUX05 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 APSUX05 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.

APSUX05 can use the APSUECA work buffer, ECAWKBUF. If this work buffer is too small, see "Exit Table, APSUCOM" on page 182 for information on expanding ECAWKBUF. If you expand ECAWKBUF, PSF does not use the additional storage (that exceeds the APSUECA default size) between successive calls. The PSF exits can therefore use this storage to maintain information between successive calls.

- You should 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.
- Macro IFASMFR maps the fields of the SMF type 6 record. For more information, refer to "Structure of SMF Type 6 Records" on page 88.

**Note:** Fields in IFASMFR provided by SVC 83 are not available to APSUX05.

- PSF can be processing a new data set during the same time that APSUX05 is processing the SMF type 6 record for a preceding data set. This may result in APSUX05 being called for the preceding data set at the same time one of the other exits is being called for the current data set.
- APSUX05 is called from the NOTIFY subtask; the following exits are called from the FSA subtask:

- APSUX01
- APSUX02



- APSUX03
- APSUX04
- APSUX06
- APSUX07

APSUX05 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. ECFEFCAP points to the ECA that is passed to the following exits:
  - APSUX01
  - APSUX02
  - APSUX03
  - APSUX04
  - APSUX06
  - APSUX07

ECENECAP points to the ECA that is passed to APSUX05.

- 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.
- Each time a different rotation of an outline font is activated, the SMF record is updated; that is, the number of fonts is incremented by 1. With raster fonts, however, a coded font is counted only once, regardless of the number of rotations of the font that are activated.

---

## PSF Message Exit: APSUX06

APSUX06 defines a message exit that is given control before most of the messages generated by PSF FSA APSxxxI and APSxxxA, 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 be 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 also use this exit to terminate a printer FSA. To do this, set the XTP6TFSA flag. After the data set runs to completion, PSF terminates the FSA.

If you set both flags, PSF terminates the FSA and the system does *not* hold the data set.

PSF provides a sample (skeleton) 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 14 on page 169.) This routine runs in the same address space as PSF.

## Input for APSUX06

When the PSF message-inspection exit is entered, the contents of the registers are as follows:

|                    |   |
|--------------------|---|
| <b>Register 1</b>  | Pointer to the address of parameter area APSGEXTP   |
| <b>Register 13</b> | Pointer to caller's save area   |
| <b>Register 14</b> | Return address  |
| <b>Register 15</b> | Entry-point address   |
| <b>XTP6PROC</b>    | FSA identifier:<br><br>Channel-attached printers: DD <i>nnn</i> , where <i>nnn</i> is the device-address<br>Non channel-attached printers: label of PRINTDEV statement                              |
| <b>XTP6MSG</b>     | Message ID (APSxxxI or APSxxxA)   |
| <b>XTP6ACT</b>     | Message action to be taken by PSF:<br><b>XTP6WTO</b> Put message to the operator console.<br><b>XTP6SEC</b> Put message to the security console.<br><b>XTP6PAP</b> Print message with the data set. |

## Output for APSUX06

|                 |  |
|-----------------|--|
| <b>XTP6HOLD</b> | Request that the system hold the data set. PSF marks the data set as one that cannot be selected.  |
| <b>XTP6TFSA</b> | Terminate FSA.   |
| <b>XTP6ACT</b>  | 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.<br><br><b>XTP6WTO</b> Put the message to the operator console.<br><b>XTP6SEC</b> Put the message to the security console.<br><b>XTP6PAP</b> Print the message with the data set.<br><b>XTP6SUP</b> Suppress the message. |

## Programming Considerations for APSUX06

The following considerations apply to APSUX06:

- 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. APSUX06 might receive control while one of the other exits is active. If APSUX06 needs to make an external call, the code should 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.

APSUX06 can use the APSUECA work buffer, ECAWKBUF. If this work buffer is too small, see "Exit Table, APSUCOM" on page 182 for information on expanding ECAWKBUF. If you expand ECAWKBUF, PSF does not use the additional storage (that which exceeds the APSUECA default size) between successive exits. The PSF exits can therefore use this storage to maintain information between successive exits.

**Notes:**

1. The transmission-count field, ECATRNC, is not updated before this exit is invoked.
  2. ECEOUTTK and ECEJDVT are not updated before this exit is invoked.
- 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 APSUX06 needs to have in order to determine its action. The pointer XTPRECP within APSGEXTP provides the address of XTP6. XTP6 is mapped within macro APSGEXTP. For more information on APSGEXTP, see Figure 50 on page 171.
  - The message action, XTP6ACT, can be changed by the exit as follows:
    - 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 APSxxxA, cannot be suppressed. If the exit requests that an action message be suppressed, PSF ignores the request and sends the message to the operator.
    - The exit cannot request that a message be 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 the following messages is issued, processing is not transferred to the exit:

```
APS000I APS006I APS043I APS049I APS055I APS066I APS551I APS955I
APS001I APS022I APS044I APS050I APS060I APS067I APS605I APS957I
APS002A APS023I APS045I APS051I APS062I APS068I APS607I
APS003A APS025I APS046I APS052I APS063I APS226I APS608I
APS004I APS038I APS047I APS053I APS064I APS516I APS610I
APS005I APS042I APS048I APS054I APS065I APS517I APS922I
```

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 by the operator interface—messages APS620I through APS699I, with the exception of 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: APSUX07

PSF can call APSUX07 at various times while processing a print data set during an FSA session. APSUX07 controls these calls by setting the appropriate flags in the XTP7NOTY field of APSGEXTP. (For a listing of the XTP7 section in APSGEXTP, see Figure 67 on page 296.) This set of calls controls many functions, including resource-management functions. See “Programming Considerations for APSUX07” on page 200 for information about all the functions that APSUX07 controls. Some of the resource-management functions let you:

- Specify whether fonts should be mapped to outline fonts, if the target printer supports outline fonts.
- Specify that PSF should verify the outline font exits in the host library when raster fonts are mapped to outline fonts.
- Specify whether a resource should be loaded from the host library, even if it already resides in virtual storage or in the printer.
- Specify which resources of a given type should be saved in the printer across data set boundaries.
- Substitute one resource for another.
- Enforce the security standards of your installation by terminating 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.

## Invoking APSUX07

The call to APSUX07 can be initiated from different places in PSF. A call to APSUX07 is defined by the following fields in the XTP7 section of APSGEXTP:

**XTP7ETYP** Exit type

**XTP7RTYP** Resource type

**Note:** For IOCA, GOCA, or BCOCA resources, this field indicates a page segment resource.

**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 50 on page 171.

## Sequence of Events

1. If APSUX07 is present at PSF initialization,<sup>12</sup> 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. At this time, APSUX07 may 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.

## APSUX07 Subroutine Calls

The input and output for the APSUX07 subroutine calls (summarized in Table 15) are described in terms of their associated processing indicators within each control block. Table 22 on page 319 references the input and output for each call, and Appendix D, Field Descriptions for APSUX07, provides a listing of the control blocks.

| Subroutine call and page number             | Description  |
|---|--|
| "Initialization Call (INIT)" on page 211    | Defines the control parameters of the resource exits.  |
| "Begin-Data-Set Call (BDSC)" on page 212    | Controls the processing of the data set.   |
| "Resource-Access Call (RAC)" on page 213    | Controls processing for individual resources—for example: <ul style="list-style-type: none"> <li>• Specifying whether a resource should be loaded from the host library, even if it already resides in virtual storage or in the printer.</li> <li>• Substituting one resource for another.</li> </ul> |
| "Resource-Load Call (RLC)" on page 218      | Requests or releases control of serially reusable resources, using ENQ or DEQ. (ENQ and DEQ are MVS system macros. For more information on system macros, refer to the macro publication for your version of MVS.)<br><br>You can also use RLC to collect statistics.                                  |
| "Resource-Deletion Call (RDC)" on page 219. | Controls resource deletion at data set end (DSE).  |
| "Termination Call (TERM)" on page 221.      | If needed, restores processing states at FSA-termination time.   |

<sup>12</sup> For a description of how PSF determines that an exit is present, see "Changing Exit Routines" on page 169.

## Programming Considerations for APSUX07

The following general considerations apply to APSUX07:

- 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 on ECAWKBUF, see Figure 51 on page 173.)
- APSUX07 can receive control at the same time that one of the other exits is active. APSUX07 should always use ECARSCSV, a unique save area provided for this exit in the ECA.

Additional pointers are provided for data areas 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

PSF can convert a page segment before loading it into the printer, 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 actually different versions of the page segment, loaded into the printer with different L-units.

### Special Performance Considerations

- XTP7NOTY flags are located in the APSGEXTP parameter area. See "XTP7 Parameter Area (DSECT)" on page 295 for more information. XTP7NOTY flags should 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. Do not, however, retain resources arbitrarily. Page segments can take up substantial amounts of storage, and retaining overlays can force PSF to retain unneeded resources.

## IOCA, GOCA, and BCOCA Considerations

IOCA, GOCA, and BCOCA resources can be placed in the page segment library. On entry to APSUX07, the XTP7RTYP field indicates a page segment resource type for IOCA, GOCA, and BCOCA resources. It is recommended that you establish a naming convention to differentiate these resources. IBM does not enforce a prefix for the eight-character resource name, but you should define a convention that identifies each type of resource in the page segment library.

IBM recommends a two-character prefix:

|           |                        |
|-----------|------------------------|
| <b>B1</b> | BCOCA resources        |
| <b>G1</b> | GOCA resources         |
| <b>I1</b> | IOCA resources         |
| <b>S1</b> | Page segment resources |

## 3800-Compatible Media Origin Considerations

APSUX07 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 an explanation of these differences, refer to the appendix describing “Compatibility, Conversion, and Performance” in *AFP: Printer Information*.

When 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 APSUX07 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 newer printer. As new applications are developed or old ones are modified to take advantage of new functions, the work should be done to remove the need for the 3800 origin override. Jobs designated by this exit bit should 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 designated as 3800-compatible, the 3800-compatible bit is ignored. PSF processes the job, using the N\_UP capabilities of the 3900 or 3835 printer to which the job is sent, but issues a warning message noting the conflict between the exit bit and the form definition.

If the exit bit 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 exit bit is ignored, but no message is issued.

You can use APSUX07 to control whether to set the media origin to the 3800 media origin.

|                 |  |
|-----------------|--|
| <b>XTP738MO</b> | Controls setting the media origin for the user's data set on continuous-forms printers to the 3800 media origin. |
| <b>XTP7JHMO</b> | Controls setting the media origin for the job header on continuous-forms printers to the 3800 media origin.      |
| <b>XTP7JTMO</b> | Controls setting the media origin for the job trailer on continuous-forms printers to the 3800 media origin.     |

**XTP7DHMO** Controls setting the media origin for the data set header on continuous-forms printers to the 3800 media origin.

**XTP7MDMO** Controls setting the media origin for the message data set continuous-forms printers to the 3800 media origin.

These flags default to 0. The default specifies that PSF is not to set the media origin to the 3800 media origin. If any of the flags is set to 1, PSF sets the media origin to the 3800 media origin for that particular page or pages. Any or all of the flags can be set to 1 during a call to APSUX07. These flags can be set during the INIT call or the BDSC call, and remain set between calls; only APSUX07 can change the state of these flags.

If you set one or more of the APSUX07 3800 media origin flags, use wide paper on the 3800, and use cut-sheet emulation, the media origin will not be in the correct position. PSF assumes narrow paper (portrait pages are generated for printing in cut-sheet emulation mode).

You can use APSUX07 to control whether PSF sets the media origin for wide or narrow paper:

**XTP7WIDE** Controls whether landscape or portrait pages are generated for printing in cut-sheet emulation mode.

This flag defaults to 0. The default specifies that PSF is to print portrait pages. If the flag is set to 1, PSF assumes wide paper and prints landscape pages. This flag is only valid when using cut-sheet emulation with one or more of the 3800 media origin flags set in APSUX07. This flag can be set during the INIT call or the BDSC call and remains set between calls; only APSUX07 can change the state of this flag.

### **Considerations on Suppressing an Interrupt Message Page**

You can use APSUX07 to control whether the interrupt message page that PSF inserts in your printed output is to be printed (such as the message page for a paper jam). Suppressing interrupt message pages is suggested when PSF is attached to a printer with Finisher capability; otherwise, interrupt message pages will separate finished documents into two finished groups with the inserted page in the middle.

**XTP7IPSU** Controls the suppression of the interrupt message page.

This flag defaults to 0. The default specifies that PSF is to print the interrupt message page. If the flag is set to 1, PSF suppresses the interrupt message page. The flag can be set during the INIT call, and it remains set between calls; only APSUX07 can change the state of this flag.

**Note:** If printing to microfilm, PSF automatically suppresses the printing of interrupt messages, and this flag has no effect.

### **Considerations on Redirecting a Message Data Set**

With APSUX07 you can control the capability of redirecting a message data set as a SYSOUT data set to another CLASS or DEST for viewing or printing.

**XTP7MDSD** Controls the redirection of a message data set.

This flag defaults to the PIMSG keyword setting on the OUTPUT statement for the message data set. If the flag is set to 0, PSF will not send the message data set to a SYSOUT data set. If the flag is set to 1, PSF will send the message data set to a



SYSOUT data set. The keywords specified on the OUTPUT statement for the message data set referenced by the MESSAGE= keyword in the PRINTDEV statement will be attributed to the generated SYSOUT data set. The flag can be set during the INIT call or during the BDSC call. This flag does *not* remain set between calls; it is set to the value based on the PIMSG keyword.

## Overlay Considerations

Before an overlay containing fonts or page segments can be printed, the overlay and its associated resources must be loaded in the printer. Fonts and page segments 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 using the resources begins to print.
- 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, refer to *Mixed Object Document Content Architecture Reference*.

If APSUX07 is not set up to be called at Resource-Deletion for overlays, PSF deletes all overlays at data set end. There is one exception to this rule: if there is an overlay specified in the PRINTDEV, it is kept along with any resources used by that overlay. If APSUX07 is called at Resource-Deletion for overlays, the exit is in full control and the only overlays that are kept are those with the RLSTKEEP flag set ON. If you are specifying overlays in the PRINTDEV and using the RDC for overlays, you should keep the PRINTDEV overlays. (For more information on deleting resources, see “Full Control of Resource Deletion” on page 220.)

## Considerations on Converting a Multiple-Celled IM1 Image

With APSUX07, you can control whether a multiple-celled IM1 is converted to a single IOCA image or to multiple IOCA images:

**XTP7C2SI** Controls the conversion of a multiple-celled IM1 image.

This flag defaults to 0. The default specifies that PSF is to convert a multiple-celled IM1 image to multiple IOCA images. If the flag is set to 1, PSF converts a multiple-celled IM1 image to a single IOCA image. The flag can be set during the INIT call or the BDSC call, and it remains set between calls; only APSUX07 can change the state of this flag.

This request is honored 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 may degrade performance.

## Considerations on Eject-to-Front-Facing Function

If your continuous-forms printer supports the eject-to-front-facing (ETFF) function, PSF issues an ETFF command before the job-header page and before the start of a new document. With APSUX07, you can control the ETFF by use of two flags:

**XTP7ETFF** Controls the ETFF issued between documents in a data set.

**XTP7EFFH** Controls the ETFF issued before the job-header page.

Both of these flags default to 0. The default specifies that PSF is to issue the ETFF. If the flags are set to 1, PSF does not issue the ETFF. The flags can be set during the INIT call or the BDSC call and remain set between calls; only APSUX07 can change the state of these flags.

### **Considerations on Restartable Abends**

PSF automatically restarts when certain abend reason codes occur. See “Understanding Restartable Abends” on page 255 for more information on restartable abends.

Because restarting after an abend may cause duplicate pages to print without notification to the operator or the job submitter, you may want to prevent PSF from automatically restarting after abends. With APSUX07, 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 APSUX07 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 on Requesting an Acknowledgement**

With APSUX07, you can indicate to PSF to only 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 due to buffer shortages. The amount of PSF buffer storage is controlled by the BUFNO parameter in the PRINTDEV. See Chapter 6, “PRINTDEV Parameters” on page 41 for more information about the BUFNO parameter.

Use the following flag to request an acknowledgement:

**XTP7APSF** Controls requesting of acknowledgements.

This flag 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. This field can be set during the INIT call and it remains set between calls, only APSUX07 can change the state of this flag.

**Attention:** If you are using channel extenders, check with the vendor of your channel extender before setting XTP7APSF. Many channel extenders will not work correctly when XTP7APSF is set.

## Considerations on Cut-Sheet Emulation Mode

With APSUX07, you can control how PSF checks the pages for cut-sheet emulation mode. If a page is checked and found not to fit 2 up on the sheet, the page is printed in normal 1\_UP. For more information, see Chapter 9, “Using Cut-Sheet Emulation Mode” on page 101.

**XTP7ACSE** Controls the page checking for cut-sheet emulation mode

This field defaults to 0(XTP7NCK). The default specifies that PSF is not to check to see whether the page fits 2 up on the sheet. If this field is set to 1(XTP7K1P), PSF checks only the first page printed for a new code page to see if it fits 2 up on the sheet. If the field is set to 2(XTP7CKAP), PSF checks all the front side pages to see if they fit 2 up on the sheet. The field can be set during the INIT call or the BDSC call, and it remains set between calls; only APSUX07 can change the state of this field.

## Considerations on Forms Marking of an Interrupt Message Page

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.

The following fields are used to control this function:

|                 |   |
|-----------------|---|
| <b>XTP7IPMF</b> | Controls forms marking for the interrupt message page.                |
| <b>XTP7IPOS</b> | Controls offset stacking for the interrupt message page.              |
| <b>XTP7IPCC</b> | Controls the number of copies printed for the interrupt message page. |

Both of the flags, XTP7IPMF and XTP7IPOS, default to 0. The default specifies that PSF is not to issue any mark forms or offset stacking requests for the interrupt message page. The flags and the number of copies can be set during the INIT call, or the BDSC call and remain set between calls. Only APSUX07 can change these fields. If the copies field, XTP7IPCC, is not specified, or if an incorrect value is specified, it defaults to 1. XTP7IPCC is used only when XTP7IPMF is set to 1.

## Considerations Related to Page Definitions and Form Definitions

The following page definitions and form definitions are useful when you are working with APSUX07:

|                       |   |
|-----------------------|---|
| <b>User-specified</b> | 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.                                  |
| <b>Auxiliary</b>      | A page definition or a form definition specified in the PSF startup procedure for use in an auxiliary data set. Auxiliary data sets are: <ul style="list-style-type: none"><li>• Job headers</li><li>• Job trailers</li></ul> |

- 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 JCL OUTPUT 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 could 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 38.
- 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 specified in the PRINTDEV statement and modifies it with the respective parameters from the JCL OUTPUT 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 on the selection hierarchy of page definitions and form definitions, see "How PSF Selects Resources" on page 115.

### Data-Set Termination Considerations

By setting the XTP7TDS flag to 1, you can use APSUX07 to request termination of the current data set. You can set this flag during the BDSC, the RAC, or the RLC call.

A terminated data set is usually purged. To request that a terminated data set be returned to JES, set the XTP7HQUE flag to 1. PSF then marks the data set as one that cannot be selected.

When a data set is terminated, 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 APSEXTP. For more information on XTP7, see Figure 67 on page 296.

### 3800 Considerations

You can use APSUX07 to select the 3800 hardware default font. Initially, the XTP7HDF flag is set to 0, indicating that the 3800 should use the hardware default font. APSUX07 can set XTP7HDF to 1, to specify that the printer is to take the first font in the current font list as the hardware default.

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 a data set is terminated with PSF message APS921I.

**Note:** The hardware default font may 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.

### 3800 Forms Marking

The 3800 prints three to five 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 the 3800 printer, PSF ignores the copies field, XTP7IPCC; the 3800 device controls the number of copies printed of the interrupt message page. For more information, see “Forms Marking” on page 81.

### Considerations for Setting a Format Resolution Value

By setting the XTP7FMT field, Exit 7 can override the format resolution specified in the data stream and on the user’s JCL.

XTP7FMT can be set during the BDSC call to 240-pels or 300-pels per inch resolution. This lets you 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 and another that is 300, 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, the exit could set the XTP7FMT value based on the characteristic for each data set.

XTP7FMT can also be set during the INIT call to 240-pels or 300-pels per inch resolution. This specifies the format resolution for all jobs and can only be overridden by the BDSC call.

XTP7FMT defaults to 0, which indicates that no resolution is specified. This field can be set during the INIT call or the BDSC call and remains set between calls. Only APSUX07 can change this field. If this field is not specified, or if an incorrect value is specified, it defaults to 0.

## Considerations Related to User Libraries

The following processing considerations apply to resources stored in a user library:

- APSUX07 cannot substitute user library resources. PSF ignores a request to do so.
- APSUX07 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 219.
- APSUX07 is passed a concatenation retrieval list, APSLCRL, for user library resources.

## Considerations Related to Inline Resources

PSF supports inline resources—form definitions, page definitions, overlays, page segments, character sets, code pages, and coded fonts—as well as inline object containers. In addition, inline IOCA, GOCA, and BCOCA resources are supported (these resources are contained in the page segment library). The following processing considerations apply to inline resources:

- APSUX07 cannot substitute inline resources. PSF ignores a substitution request for an inline resource.
- APSUX07 cannot keep an inline resource across data set boundaries. PSF deletes inline resources before the Resource-Deletion Call (RDC).
- APSUX07 is not passed a concatenation retrieval list (APSLCRL) for inline resources.
- APSUX07 cannot force a reload of an inline resource. When PSF is unable to reload a resource, RLSTRRI is set ON. If PSF requests that the resource be reloaded, the request is ignored and no error occurs. If APSUX07 requests substitution, and PSF determines that the substitution request should be honored, PSF attempts to retrieve the substituted resource only from the library from which the original resource would have been retrieved.

## Considerations Related to Outline Fonts

PSF supports outline fonts as well as raster fonts. When the printer does not support outline fonts, PSF maps an outline font to an equivalent raster font, as is explained in Chapter 16, Using Outline Fonts. This mapping function is always active; it occurs before the resource-access call. For more information, see “Font Mapping” on page 149.

For performance considerations, see “Using Outline Fonts” on page 233. For information on changing the mapping, see “Using the Font Mapping Tables” on page 150.

The following processing considerations apply to outline fonts:

- If Exit 7 is invoked and a font substitution request is made, but the printer does not support the type of font substituted, PSF terminates 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 be reloaded, you may experience some performance degradation if several coded fonts are using the same code page or character set. All the fonts 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 may use the same code pages and character sets, other instances of these font members are deleted (and made unusable) at the same time.

### **Considerations Related to 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.

XTP7MTOF controls whether PSF maps fonts to outline fonts. The XTP7MTOF flag defaults to zero. The default specifies that PSF is *not* to map fonts to outline fonts. If the flag is set to 1, PSF attempts to map the font to a corresponding outline font, using user and system mapping tables. For more information, see “Font Mapping” on page 149 and “Using the Font Mapping Tables” on page 150. The flag can be set during the INIT call, and it remains set between calls. Only APSUX07 can change the state of this flag.

**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.

**Note:** 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 233.

### **Considerations Related to Recover-from-Font-Not-Found Flag**

If your printer supports outline fonts and you have requested that PSF map to outline fonts (set XTP7MTOF to 1), but many pages in your jobs are terminated because the outline font identified through the mapping algorithms did not exist on the host, you may want to turn on the Recover-from-Font-Not-Found function.

The XTP7RFNF flag can be set during the INIT call, and remains set between calls. Only APSUX07 can be used to change the state of this flag. The XTP7RFNF flag defaults to 0, 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 terminates 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 XTP7RFNF flag to 1. If XTP7RFNF is set to 1, PSF performs 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 XTP7RFNF is set to 1 and 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 there are additional library queries for every mapped font resource, setting XTP7RFNF to 1 degrades the performance of PSF.

### **Considerations Related to Inhibiting Recovery**

The normal PSF error recovery actions include resetting to the point in the data path at which an exception 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 should be discarded. By setting the XTP7IHIB flag on the INIT call or on the BDSC call for the first data set of a job, you prevent PSF from resending any data that has already printed. This gives you the ability to prevent reprinting of sensitive data, such as checks.

When an exception (error) is reported to PSF, the point at which recovery is done depends on the type of exception that is reported. Each type of exception 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 XTP7IHIB flag is set, PSF does not perform any recovery beyond the point at which the printer begins to put 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 terminated.

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 should only be used on cut-sheet printers.

When recovery is inhibited, the only operator interruption or repositioning command that is allowed is the cancel command.

The XTP7IHIB flag defaults to 0, which specifies that PSF perform a total recovery. If the flag is set to 1, PSF does not perform any recovery for pages in this job that are already on paper. The flag can be set during the INIT call or the BDSC call and remains set between calls; only APSUX07 can change the state of this flag. If the setting of the XTP7IHIB flag is changed within a job, PSF ignores the new setting and issues a warning message.

### **Considerations Related to Ending the SNA LU1 Conversation**

If PSF ends the SNA LU1 conversation with the printer while maintaining its SNA session with the printer, the printer can honor a request for a print screen without deleting the resources that PSF has loaded into the printer. When APSUX07 sets XTP7ENDC to B'1' on the INIT call, PSF ends the SNA LU1 conversation between print jobs when the NPRO timer expires or after no job is available for one minute and the last page printed has been stacked.

The XTP7ENDC flag defaults to B'0'. The default specifies that PSF should not end the SNA LU1 conversation without ending the SNA session as well. If the flag is set to B'1'. and the printer is SNA LU1 attached, PSF ends the conversation with the printer while maintaining its SNA session with the printer. The flag can be set during



the INIT call, and it remains set between calls; only APSUX07 can change the state of this flag.

## Initialization Call (INIT)

The INIT call is made once, after the FSA has been initialized for each device. By modifying XTP7NOTY flags (listed in Figure 67 on page 296), you can use the INIT call to set up for future calls to APSUX07. As defaults, XTP7NOTY flags are set inactive; however, you can modify these flags to:

- Control resource access by type<sup>13</sup>; set any of the XTP7NACC flags.
- Load resources by type<sup>13</sup>; 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 an FSA terminates; set XTP7ETRM=1.

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.

This call has several other uses:

- To control ETFF. For more information on ETFF, see “Considerations on Eject-to-Front-Facing Function” on page 203.
- To control whether to set the media origin to the 3800 media origin. For more information, see “3800-Compatible Media Origin Considerations” on page 201.
- To control whether to print portrait or landscape pages when using cut-sheet emulation and 3800 media origin. For more information, see “3800-Compatible Media Origin Considerations” on page 201.
- To control the printing of portrait or landscape pages. For more information, see “3800-Compatible Media Origin Considerations” on page 201.
- To control the conversion of a multiple-celled IM1 image. For more information, see “Considerations on Converting a Multiple-Celled IM1 Image” on page 203.
- To control the marking and offset stacking of the interrupt message page. For more information, see “Considerations on Forms Marking of an Interrupt Message Page” on page 205.
- To modify the “reasonable resource loading values” (RRLV). PSF uses the RRLV to determine the number of resources 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 219.
- To control the suppression of the interrupt message page. For more information, see “Considerations on Suppressing an Interrupt Message Page” on page 202.
- To control the message data set redirection. For more information, see “Considerations on Redirecting a Message Data Set” on page 202.

---

<sup>13</sup> 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.

- To control how PSF checks the pages for cut-sheet emulation mode. For more information, see “Considerations on Cut-Sheet Emulation Mode” on page 205.
- To set a format resolution for all jobs. For more information, see “Considerations for Setting a Format Resolution Value” on page 207.
- To control the ending of the PSF SNA LU1 conversation. For more information, see “Considerations Related to Ending the SNA LU1 Conversation” on page 210.
- To control when PSF requests an acknowledgement. For more information, see “Considerations on Requesting an Acknowledgement” on page 204.
- To control whether PSF recovers from an error. For more information, see “Considerations Related to Inhibiting Recovery” on page 210.

### Input and Output Indicators for INIT

For a complete cross-reference of the XTP7 and RLST fields used as input and output, see Table 22 on page 319.

### Programming Considerations for INIT

With APSUX07 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 Figure 67 on page 296.

**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. You can use this call to do the following:

- Control EFFF. See “Considerations on Eject-to-Front-Facing Function” on page 203.
- Request termination of the data set (XTP7TDS). See “Data-Set Termination Considerations” on page 206.
- Modify XTP7RRLV and XTP7NOTY. For XTP7RRLV, see “Partial Control of Resource Deletion” on page 219. For XTP7NOTY, see “Initialization Call (INIT)” on page 211.

- Control the marking and offset stacking of the interrupt message page. See “Considerations on Forms Marking of an Interrupt Message Page” on page 205.
- Control the setting of the media origin to the 3800 media origin. See “3800-Compatible Media Origin Considerations” on page 201.
- Control converting a multiple-celled IM1 image. See “Considerations on Converting a Multiple-Celled IM1 Image” on page 203.
- Control the page checking for Cut-Sheet Emulation mode. See “Considerations on Cut-Sheet Emulation Mode” on page 205.
- Set a format resolution on a data set boundary. See “Considerations for Setting a Format Resolution Value” on page 207.
- Limit the recovery performed by PSF. See “Considerations Related to Inhibiting Recovery” on page 210.

### Input and Output Indicators for BDSC

For a complete cross-reference of the XTP7 and RLST fields used as input and output, see Table 22 on page 319.

### 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 on the basis of resource type, occurs when PSF determines that it needs a resource in order 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 the following:

- Request termination of the data set (XTP7TDS). For information on messages issued when data sets are terminated, see “Data-Set Termination Considerations” on page 206.

**Note:** Terminating 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 RLSTMSG flag in the RLSTAFGL 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 RLSTMSGGL. 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 APSUX07 requests substitution, and PSF determines that the request

should be honored, PSF attempts to retrieve the substituted resource only from the library from which the original resource would have been retrieved.

- Force a load or reload<sup>14</sup> of the resource by setting the RLSTLOAD flag in the RLSTAFLG field of the APSURLST control block.
  - If the resource has not been loaded, the action is ignored.
  - If the resource has been loaded, the resource is deleted and loaded from the library.
  - When processing certain map structured fields, PSF 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 calling the exit. PSF ignores any request to reload the resource and no error occurs. PSF continues processing the resource as though the reload had not been 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 on PLRs, see "Partial Control of Resource Deletion" on page 219.
- Modify the XTP7NOTY and XTP7RRLV fields. For information on XTP7NOTY, see "Initialization Call (INIT)" on page 211. For information on XTP7RRLV, see "Partial Control of Resource Deletion" on page 219.

## Input and Output Indicators for RAC

For a complete cross-reference of the XTP7 and RLST fields used as input and output, see Table 22 on page 319.

## Programming Considerations for RAC

- APSUX07 is called to process a mapped resource each time PSF encounters a structured field for the resource:
  - Map medium overlay (MMO) structured field
  - Map page segment (MPS) structured field
  - Map coded font (MCF) structured field
  - Include page segment (IPS) structured field
  - Include page overlay (IPO) structured field
  - Map page overlay (MPO) structured field

Even if the resource containing the structured field remains loaded, subsequent references to that structured field result in calls to APSUX07.

---

<sup>14</sup> To have the exit request that a resource be "reloaded," set RLSTLOAD to 1. PSF does not use an existing version of the resource, but loads the resource from the host library.

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 the resource is mapped, APSUX07 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 may 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 be 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 be 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 mapped in an MPO, the page must contain an include page overlay (IPO) structured field. Each time PSF encounters an MMO, an MPO, or a IPO structured field, it checks to see if APSUX07 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, refer to *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 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 should be aware that:

- APSUX07 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 on reloading resources, see the bullet on page 214.
- PSF calls the exit each time PSF encounters an MPS or Include Page Segment (IPS) structured field.

For more information about page segment structure, refer to *Mixed Object Document Content Architecture Reference*.

## **RAC and Fonts**

Fonts are resources 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 APSUX07 is active for fonts at access time. If APSUX07 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 (those with code-page–character-set pairing), PSF builds an RLST font-extension area. PSF stores the names of the code-page–character-set pairs 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 may 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 will be 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, neither the code page nor the character set name can 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 on coded fonts, code-page–character-set pairs, single-byte fonts, double-byte fonts, and the MCF structured field, refer to *Advanced Function Printing: 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 205.

### Default and Auxiliary Processing

Default and auxiliary page definitions and form definitions are processed in two ways:

- During the initialization of PSF (APSUX07 is called for the system defaults, job header, job trailer, data set header, and message data set; this is the only time APSUX07 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 (terminate 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 on reloading resources, see the bullet on page 214.

### User-specified processing

A user can specify page definitions and form definitions in the OUTPUT JCL statement. All the functions listed under “Resource-Access Call (RAC)” on page 213 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, refer to *PSF for OS/390: User's Guide*. When inline resources are used, all the functions listed under “Resource-Access Call (RAC)” on page 213 are supported; however, no substitution for inline resources can be made.

## Resource-Load Call (RLC)

The resource-load call (RLC) occurs just before and just 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 on the basis of 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.<sup>15</sup>

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.”
- APSUX07 can modify the RLSTPLR flag for the resource being loaded, and it can modify XTP7NOTY and XTP7RRLV. For more information on XTP7NOTY, see “Initialization Call (INIT)” on page 211; for more information on XTP7RRLV, see “Partial Control of Resource Deletion” on page 219.
- Request termination of the data set (XTP7TDS). For information on issuing messages when terminating data sets, see “Data-Set Termination Considerations” on page 206.

### Input and Output Indicators for RLC

For a complete cross-reference of the XTP7 and RLST fields used as input and output, see Table 22 on page 319.

### Programming Considerations for RLC

- PSF cannot ensure that the ENQ and DEQ macros will be used by multiple users of a resource library. Each installation is responsible for the use of these macros.

**Note:** You should not run the system ENQ macro for a member and then request the termination of the data set.

---

<sup>15</sup> ENQ and DEQ are MVS system macros. For more information on system macros, refer to the macro publication for your version of MVS.



- 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 loading single-byte fonts, PSF can access the library several times. When loading double-byte fonts, PSF 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 on the basis of resource type.

PSF deletes resources by applying a least-recently-used algorithm. PSF uses this method to maintain the number of resources loaded, but you can use APSUX07 to take full or partial control of resource deletions. The following sections, “Partial Control of Resource Deletion” and “Full Control of Resource Deletion” on page 220, describe the PSF deletion algorithm.

### Input and Output Indicators for RDC

For a complete cross-reference of the XTP7 and RLST fields used as input and output, see Table 22 on page 319.

### Partial Control of Resource Deletion

You can modify the least-recently-used PSF algorithm in two ways:

- Modify the reasonable resource loading values (RRLV). See Table 23 on page 325 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 APSUX07.

RRLVs have no upper or lower limit, except that for overlays and object containers RRLV is always 0. (For retention of overlays at DSE, see “Full Control of Resource Deletion” on page 220.) However, setting the RRLVs too high can cause a resource-constrained condition, in which the printer does not have enough storage for the resources required to print the current page. Setting the RRLVs too low can lead to additional downloading of resources.

**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 on auxiliary resources, see “Considerations Related to Page Definitions and Form Definitions” on page 205.

The one exception is that with overlays, you must have full control of resource deletion. For more information, see “Full Control of Resource Deletion” on page 220.

- 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 219.

You can take full control of resources kept or deleted by modifying RLSTKEEP. On return from APSUX07, PSF checks the RLST chain and deletes resources whose RLSTKEEP flags are turned off. The number of resources kept is not limited to the RRLV.

If APSUX07 does not modify the RLSTKEEP flags, PSF deletes resources on the basis of the least-recently-used algorithm.

**Note:** PSF tries to save all fonts and page segments 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.

Some restrictions apply, however. For more information, see “Using RDC to Control Resources.”

## Using RDC to Control Resources

Keep the following considerations in mind when 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 zero (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 containing 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.

- APSUX07 can modify the XTP7NOTY fields.
- APSUX07 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 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 calling the exit.
- PSF does not delete object container resources. The printer manages these resources. PSF ignores requests to delete or retain object container resources.

## 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 during the termination of an FSA, TERM occurs. For example, if a resource exit started the GETMAIN system macro during the INIT call, the TERM call could 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 204 for more information.

### Input and Output Indicators for TERM

For a complete cross-reference of the XTP7 and RLST fields used as input and output, see Table 22 on page 319.

---

## Direct-Printing Line Mode Exit: APSUX14

In direct-printing mode, PSF gives the APSUX14 exit control for the first record of the transmission so the user can invoke 3800 Line-Mode Conversion.

### Input for APSUX14

When the direct-printing line mode exit is entered, the contents of the registers are as follows:

|                    |   |
|--------------------|---|
| <b>Register 1</b>  | Pointer to the address of parameter area APSGEXTP |
| <b>Register 13</b> | Pointer to the caller's save area                 |
| <b>Register 14</b> | Return address                                    |
| <b>Register 15</b> | Entry-point address                               |
| <b>XTP4CONV</b>    | Do conversion flag                                |
| <b>XTP4XOFF</b>    | Horizontal offset value                           |
| <b>XTP4YOFF</b>    | Vertical offset value                             |
| <b>XTP4WINA</b>    | Wide or narrow paper flag                         |
| <b>XTP4LMRG</b>    | Line merge flag                                   |
| <b>XTP4DPLX</b>    | Duplex field                                      |

## Output for APSUX14

|                 |                           |
|-----------------|---------------------------|
| <b>XTP4CONV</b> | Do conversion flag        |
| <b>XTP4XOFF</b> | Horizontal offset value   |
| <b>XTP4YOFF</b> | Vertical offset value     |
| <b>XTP4WINA</b> | Wide or narrow paper flag |
| <b>XTP4LMRG</b> | Line merge flag           |
| <b>XTP4DPLX</b> | Duplex field              |

## Programming Considerations for APSUX14

The following general considerations apply to APSUX14:

- 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 appropriately.
  - 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 appropriately. If the data set does not use line merge, this flag is ignored.
  - If duplexing is desired, 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.

---

## Download for OS/390 Print Parameter Exit: APSUX15

Using the APSUX15 exit with Download for OS/390, you can provide additional job information (such as passthru print parameters) to be transmitted to the target Download for OS/390 server on AIX. Refer to *PSF for OS/390: Download for OS/390* for a description of this exit.

---

## Direct-Print Exit: APSUX16

APSUX16 is the exit given control by PSF in direct-printing mode. The exit is invoked for the following calls:

- Begin-data-set call
- Message call
- End-data-set call

### Begin-Data-Set Call (BDSC)

PSF invokes APSUX16 at the beginning of the data set. When APSUX16 receives control, no processing has 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 APSUXP16 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 has been incremented by 1 for each page that has been 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 APSUXP16 control block, PSF abends unless the received page counter has been incremented by 1 for each page that has been 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 setting XTP16RPC.

Use XTP16APS to specify that PSF should 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 due to buffer shortages. The amount of PSF buffer storage is controlled by the BUFNO parameter in the PRINTDEV. See Chapter 6, “PRINTDEV Parameters” on page 41 for more information about the BUFNO parameter.

**Attention:** If you are using channel extenders, check with the vendor of your channel extender before setting XTP16APS. Many channel extenders will 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 APSUX16 Begin-Data-Set Call

When this exit is entered, the contents of the registers are as follows:

|                    |   |
|--------------------|---|
| <b>Register 1</b>  | Pointer to the address of parameter area APSGEXTP |
| <b>Register 13</b> | Pointer to the caller's save area                 |
| <b>Register 14</b> | Return address                                    |
| <b>Register 15</b> | Entry-point address                               |

|                 |  |
|-----------------|--|
| <b>XTP16SPI</b> | Stacked page ID                        |
| <b>XTP16DCB</b> | DCB address of the application program |
| <b>XTP16DEV</b> | Unit address of printer                |
| <b>XTP16ETY</b> | Call type                              |

## Output for APSUX16 Begin-Data-Set Call

|                 |                                |
|-----------------|--------------------------------|
| <b>XTP16MRK</b> | Forms mark flag                |
| <b>XTP16RPC</b> | Received page counter flag     |
| <b>XTP16APS</b> | Acknowledgement per sheet flag |

## Message Call for APSUX16

PSF invokes APSUX16 before issuing most FSA-generated messages (APSnnnl and APSnnnA).

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. If a message is intended for the operator or the security console, however, the exit cannot redirect it to be printed with the data set.

## Input for APSUX16 Message Call

When the PSF direct-print exit is entered, the contents of the registers are as follows:

|                    |   |
|--------------------|---|
| <b>Register 1</b>  | Pointer to the address of parameter area APSGEXTP |
| <b>Register 13</b> | Pointer to the caller's save area                 |
| <b>Register 14</b> | Return address                                    |
| <b>Register 15</b> | Entry-point address                               |

|                 |  |
|-----------------|--|
| <b>XTP16MSG</b> | Message number (APSnnnl or APSnnnA)  |
| <b>XTP16ACT</b> | Message action to be taken by PSF:<br><b>XTP16WTO</b> Put the message to the operator console.<br><b>XTP16SEC</b> Put the message to the security console.<br><b>XTP16PAP</b> Print the message with the data set. |

|                 |  |
|-----------------|--|
| <b>XTP16SPI</b> | Stacked page ID                        |
| <b>XTP16DCB</b> | DCB address of the application program |
| <b>XTP16DEV</b> | Unit address of the printer            |
| <b>XTP16ETY</b> | Call type                              |

## Output for APSUX16 Message Call

|                 |  |
|-----------------|--|
| <b>XTP16ACT</b> | 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:<br><b>XTP16WTO</b> Put the message to the operator console.<br><b>XTP16SEC</b> Put the message to the security console.<br><b>XTP16PAP</b> Print the message with the data set. |
|-----------------|--|

**Note:** The exit cannot request that a message be 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)

APSUX16 is invoked at the end of each data set—that is, after the data set is printed.

### Input for APSUX16 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 APSUX16 End-Data-Set Call

None.

## Programming Considerations for APSUX16

The following general considerations apply to APSUX16:

- 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, and a pointer to the exit 16 parameter list, APSUXP16.
- The following fields in APSUECA and APSUECE are valid when exit 16 is called:

|                 |                         |
|-----------------|-------------------------|
| <b>ECAECEP</b>  | APSUECE address         |
| <b>ECAUCOMP</b> | APSUCOM address         |
| <b>ECAUSAVE</b> | Save area               |
| <b>ECAWKBUF</b> | Exit work buffer        |
| <b>ECEDFDEF</b> | FORMDEF from PRINTDEV   |
| <b>ECEDPDEF</b> | PAGEDEF from PRINTDEV   |
| <b>ECEDPF</b>   | DPF is attached         |
| <b>ECEIMPS</b>  | Impression count        |
| <b>ECEJOBIM</b> | Job impression count    |
| <b>ECEJOBLP</b> | Job logical page count  |
| <b>ECEJOBPP</b> | Job physical page count |
| <b>ECELPCNT</b> | Logical page count      |
| <b>ECEPPCNT</b> | Physical page count     |
| <b>ECEPRTDT</b> | Device type from OPC    |
| <b>ECEPRTM</b>  | Device model            |
| <b>ECEPRTMD</b> | Device model from OPC   |
| <b>ECEPRTT</b>  | Device type             |
| <b>ECEPSFD</b>  | PSF Direct is attached  |
| <b>ECEREC</b>   | Record count            |
| <b>ECERPM</b>   | 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 invoked from the FSA subtask. The end-data-set call is invoked from the NST subtask.

Because the APSUX16 calls are invoked 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 should 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 will appear on the SDUMP title page if an abend occurs.
- The page and record counts—ECEIMPS, ECEJOBIM, ECEJOBPL, ECEJOBPP, ECELPCNT, ECEPPCNT, and ECEREC—are counts not of the number of pages printed, but of the pages and records that PSF has processed. PSF may 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 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, ECEJOBPL, 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.

- Execution 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 APS955I
APS001I APS022I APS044I APS050I APS060I APS067I APS605I APS957I
APS002A APS023I APS045I APS051I APS062I APS068I APS607I
APS003A APS025I APS046I APS052I APS063I APS226I APS608I
APS004I APS038I APS047I APS053I APS064I APS516I APS610I
APS005I APS042I APS048I APS054I APS065I APS517I APS922I

```

- In response to a message generated by any of the following:

- The APSRMARK utility—messages APS580I to APS599I
- The APSRFLU utility—messages APS1700I to APS1711I
- The operator interface—messages APS620I to APS699I, with the exception of message APS691I
- The trace formatter utility, APSTRFMT—messages APS960I to APS979I

- The exit can change the message action (XTP16ACT), as follows:



- If the exit does not alter XTP16ACT, PSF takes the action 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) cannot be suppressed. If the exit requests that an action message be suppressed, PSF ignores the request and sends the action message to the operator.
  - The exit cannot request that a message be 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 19. Understanding AFP Performance

This chapter 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 running on the host system.

For information about printer throughput rates (given in pages per minute, lines per minute, impressions per minute, and so on), refer to *AFP: Printer Information*.

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 performing error handling and error recovery. To understand AFP performance, consider the following:

- 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 usually stated in impressions per minute (IPM). It depends on several factors, including:

- Mechanical speed
- Internal processing speed
- 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, usually 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 being 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 will be 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 230.

## 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 could 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 cut-sheet 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 throughput.

For additional information about AFP printers, RPQs, and performance, refer to *AFP: Printer Information*.

---

## 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 of resources
- Size of the print job

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 sent to the printer consists of the following:

- Print data—text and images that print on the page
- Control information—the data defining 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, data 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 given printer will handle 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 utilization, attachment limitations, and printer throughput.

**Formatted 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 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.

**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 utilization 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 7,200 bytes for each square inch of image at 240-pel density, higher data rates and processor utilizations might be required.

**Complex image** data consists of several independent raster patterns, or single raster patterns that have been compressed. Examples are graphic objects that have been 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 once 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 utilization 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 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 terminates 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.

### 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 have to be recached. 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.

### 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. To learn which AFP printers support resident fonts, refer to *AFP: Printer Information*.

### 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 downloading 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 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 being used.

For information about how to deactivate font pruning, see the XTP7PRNS and XTP7PRND fields described on page 304.

**Note:** You can deactivate the pruning of double-byte fonts without deactivating the pruning of 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. This improves performance on future jobs that use the same fonts. For more information, see Chapter 15, “Marking Fonts for Font Capture” on page 145.

## **Size of Print Jobs**

Printing several short print jobs requires that job-initialization code and job-termination code be invoked frequently. Therefore, short jobs require more processor power than longer jobs, and can affect printer throughput.

## **Using Outline Fonts**

The character set for an outline font is generally much smaller than its equivalent for a raster font, and therefore requires less downloading of data. Further, coded fonts for outline fonts of different point sizes can share the same character set in the printer, reducing the time for downloading. Also, code pages can be shared. Therefore, when raster font downloading time would affect printer throughput, the use of outline fonts might improve printer performance. For information on how to use APSUX07 to automatically map raster fonts to outline fonts (for printers that support outline fonts), see “Considerations Related to Outline Fonts” on page 208 and “Considerations Related to Mapping to Outline Fonts” on page 209. This mapping will slightly increase PSF’s use of the central processor.

When you request that an outline font be reloaded, you might experience some performance degradation if several coded fonts are using the same code page or character set. All the fonts sharing the code page or character set will be 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 will be deleted (and made unusable) at the same time.

---

## **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 COMPRESS keyword in the PRINTDEV statement.

PSF itself can do line-data compression, which is activated by specifying the optional 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 containing 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 containing other kinds of data.
- The line data contains a significant number of blank strings or trailing blanks containing 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 (COMPRESS Parameter)” on page 238.

---

## 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 usually limited. When configuring 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, refer to 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.

With the ESCON card installed in a printer, the printer can be attached to the host with an ESCON channel, which can send and receive data faster than a parallel S/370 channel. IBM ESCON channel-attached printers that have a PCI card with Release 9.6 microcode and the 64 KB I/O buffer microcode enhancement installed might provide additional performance improvements. For information about printers that support the ESCON card, refer to *AFP: Printer Information*.

### Local-Attached Printer

For a description of attachment methods, see “Local-Attached Devices” on page 9.

The capacity of a control unit can limit the throughput of the attached printers. When PSF uses VTAM, more processor utilization is required than with channel attachment.



## Communication-Attached Printer

For a description of attachment methods, see “Communication-Attached Devices” on page 10.

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 utilization is required than with channel attachment.

Table 16 shows the approximate maximum data rates that can be reached with the attachments listed. The rates 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.

| 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 the Distributed Print Function (DPF) of PSF for OS/2 or PSF Direct. PSF can send the print data set to PSF Direct or DPF over an SDLC line, over a local token ring, or over a remote token ring.

**PSF Direct:** In a PSF Direct configuration for PSF/2, PSF for AIX, or InfoPrint for AIX, PSF sends the print data set to PSF Direct, which passes the data set to a printer attached to PSF/2, PSF for AIX, or InfoPrint for AIX. PSF Direct does not support remote resource caching.

When installing and configuring PSF Direct, consider the following:

- Data rate capability of the printer
- Data rate capability of the host attachment
- Data rate requirements of the print data set

**DPF:** DPF works with Resource Saving either OFF or ON.

In a DPF configuration, PSF sends the print data set to DPF, to be spooled and later printed. DPF requires only one PS/2. The PS/2 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 for PSF/2, PSF for AIX, or InfoPrint for AIX, depending on attachment speeds.

**Size of the VTAM Request Unit (RU):** A VTAM request unit (RU) contains system data information for a communication-attached printer. An increase in VTAM RU size might increase printing rate, because processing larger (and therefore fewer) RUs contributes to higher efficiency.

The maximum size of the RU, however, 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 allowed by the printer, whichever is smaller. Generally, use the largest RU size suitable for the printer or attachment being used.

**VTAM PSNDPAC and DELAY Parameters:** VTAM includes a coat-tailing algorithm 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 in order to achieve full utilization of high-speed communication links. VTAM provides a DELAY parameter for this purpose; specify **DELAY=0**.

For more information about coat-tailing, refer to *OS/390 eNetwork 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 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 should specify a value from 3 to 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, refer to *NCP: System Support* for your system.

## TCP/IP-Attached Printer

For a description of TCP/IP-attached printers, see Chapter 20, “Connecting a TCP/IP-Attached IPDS Printer” on page 243.

The throughput of TCP/IP-attached printers can be limited by the following factors:

- The configuration of your TCP/IP networks.

The default maximum transmission unit (MTU) size, specified on the GATEWAY statement in the TCP/IP profile on the MVS 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 4,096 bytes and a dedicated LAN. Otherwise, for printers on an Ethernet network, the MTU size must be below 1,500 bytes. For more information, see page 246.

- A large amount of network traffic.
- Communication-line speed, if data is transmitted over telephone lines.

Table 16 on page 235 shows the approximate maximum data rates that can be reached on telephone lines with different line speeds. The rates shown are not always achievable; they depend on network and system loading, tuning parameters, and the characteristics of the printer data stream.

Also, when PSF sends data to TCP/IP-attached printers, more processor utilization is required than for channel-attached printers.

## The Host System

The following factors 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 being 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 that has 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 system 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.

The 3380 and 3390 DASD devices can handle large quantities of image data. Cached data on a DASD device can hold frequently accessed page segments.

### System Tuning

A host system running many applications and driving printers with PSF must be tuned correctly, or balanced, to satisfy the data-rate requirements of the printers.

**Blank Compression for Line Data (COMPRESS Parameter):** The COMPRESS parameter 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 decompressing the blanks.

Blank compression is most likely to improve printer throughput when a printer and the jobs being 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 above characteristics are present, you might not see improvement in printer throughput. Increased CPU cycles might be used to perform the blank compression, but if the line data does not contain significant interior blanks, the volume of data 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 has eliminated all trailing blanks because blank truncation was specified.

Because the COMPRESS parameter is a PRINTDEV keyword and affects all line data jobs sent to the printer, the system programmer should carefully check the characteristics of the jobs being printed, and the above listed characteristics for the attachment and printer, before implementing the new parameter.

**ACK Per Sheet:** APSUX07 can be used to indicate to PSF to only 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 only improves performance when PSF is driving channel-attached printers. If you are using channel extenders, check with the vendor of your channel extender before enabling ACK per sheet. Many channel extenders do not work correctly when ACK per sheet is enabled.

**Printer System 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 utilization.

Table 17 shows the approximate system data rates that some printers require while printing different data applications that have pages containing a large amount of data. The data rate capability of high-speed printers is usually 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 one or more of these printers. Host processor utilization 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 230.

Table 17. Approximate Maximum Printer Data Rates

| Description of data    | Printer data rate (bytes per second) |                             |                        |                                   |                                  |                               |                                   |                                    |
|------------------------|--------------------------------------|-----------------------------|------------------------|-----------------------------------|----------------------------------|-------------------------------|-----------------------------------|------------------------------------|
|                        | VTAM-Attached                        |                             | ESCON Channel-Attached |                                   |                                  |                               |                                   |                                    |
|                        | IP20<br>3174<br>(20<br>PPM)          | IP32<br>3174<br>(32<br>PPM) | IP60<br>(60<br>PPM)    | IP3000<br>Simplex<br>(173<br>PPM) | IP3000<br>Duplex<br>(346<br>PPM) | IP4000<br>IS2<br>(480<br>PPM) | IP4000<br>ID1/ID2<br>(780<br>PPM) | IP4000<br>ID3/ID4<br>(1002<br>PPM) |
| Formatted line data    | 12 000                               | 20 000                      | 35 000                 | 102 000                           | 200 000                          | 220 000                       | 330 000                           | 600 000                            |
| Simple composed text   | 12 000                               | 20 000                      | 35 000                 | 102 000                           | 200 000                          | 220 000                       | 330 000                           | 600 000                            |
| Complex text and image | 6 600                                | 34 000                      | 172 500                | 430 000                           | 714 000                          | 700 000                       | 380 000                           | 1 000 000                          |
| Uncompressed image     | 41 000                               | 44 000                      | 375 000                | 890 000                           | 1 400 000                        | 1 000 000                     | 1 200 000                         | 1 700 000                          |
| Compressed image       | 6 400                                | 25 000                      | 335 000                | 880 000                           | 700 000                          | 500 000                       | 585 000                           | 1 200 000                          |

**Note:** IP = InfoPrint; IS2, ID1-4 are models; PPM = Pages per minute

System 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, use UNBLOCK in the DATAACK parameter on PRINTDEV statements in the PSF startup procedure. This causes PSF to report and correct errors that can affect printer throughput in some circumstances, unless the default (BLOCK) is used for a particular job.

**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<sup>16</sup> 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 utilization per page is used. As print rate increases, processor utilization *per unit of time* increases.

<sup>16</sup> As specified in JES2 trackcell parameter, for example. A TRKCELL value of at least 5 is recommended.

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, APSUX07, 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 number of output buffers is controlled by the BUFNO parameter on the PRINTDEV statement. For more information, see Chapter 6, PRINTDEV Parameters.

For each printer not attached to VTAM or TCP/IP, PSF acquires a buffer pool equal to 32 KB times the BUFNO value in the PRINTDEV statement. The output buffer pool is fixed in storage above the 16 MB line.

PSF uses the minimum BUFNO values if a BUFNO value less than the minimum is specified:

- The minimum BUFNO value is 1.
- For printers that support 16 KB I/O buffers, the minimum BUFNO value is 1.
- For printers that support 32 KB I/O buffers, the minimum BUFNO value is 2.
- For printers that support 64 KB I/O buffers, the minimum BUFNO value is 4.

For 3800 printers, PSF uses the maximum BUFNO value of 15 if a BUFNO value greater than 15 is specified.

As a general rule, the BUFNO value is set to the number of 32 KB buffers needed to support channel printer I/O operations. Table 18 recommends BUFNO values based on the complexity of your page.

| <i>Table 18 (Page 1 of 2). Recommended BUFNO values</i> |   |
|---|---|
| <b>BUFNO Value</b>                                      | <b>Application Types</b>  |
| 1–2   | Light and simple text; very 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. |

| <i>Table 18 (Page 2 of 2). Recommended BUFNO values</i> |  |
|---|--|
| <b>BUFNO Value</b>                                      | <b>Application Types</b>   |
| 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  | Very 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 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. Larger blocks, however, require more real storage.

Increasing the block size can increase processor utilization by increasing the print rate. Increasing the block size for jobs that are printing at rated speed can decrease processor utilization. Efficient removal of data from the spool is controlled by initialization parameters in the spooling system. For more information, refer to 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 a lot of 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 should be on different channels, and both should be separated from the printer channel.

**Functional Subsystem Interface (FSI):** In deferred-printing mode, JES communicates with PSF through the Functional Subsystem Interface. 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 should be assigned to different performance groups.

**Checkpointing:** A checkpoint provides the information needed to restart a print data set after an interruption or a recoverable error condition. When restarting, PSF instructs the printer to discard all unprinted pages, reposition to the last checkpoint, and resume printing from that point. If no checkpoints have been 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 should 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 utilization 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,<sup>17</sup> PSF issues a print buffered data command each time a checkpoint is taken. This command is required to make sure that the printer has completely checked the data for errors. After issuing this data command, PSF must wait until all the pages that have been sent to the printer have been printed. If the checkpoint intervals are too frequent, the performance of these printers can be slowed.

**Tracing:** Tracing is useful in diagnosing problems; however, tracing increases processor utilization and DASD requirements, and can decrease printer throughput. Unless you need to do problem determination, turn tracing off.

---

<sup>17</sup> 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 20. Connecting a TCP/IP-Attached IPDS Printer

This chapter describes how to configure MVS, TCP/IP, and PSF to print on TCP/IP-attached IPDS printers, including IPDS printers attached by use of the i-data 7913 IPDS Printer LAN Attachment.

---

### Requirements for Using a TCP/IP-Attached IPDS Printer

To print on TCP/IP-attached IPDS printers through PSF 3.1.0, you need TCP/IP 3.2 or higher installed and configured on MVS.

**Note:** 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, code the following EXEC statement in the appropriate PSF writer procedure:

```
//STEP01 EXEC PGM=APSPPIEP,REGION=4096K,PARM=(,,,tcpip_name)
```

where *tcpip\_name* is the name of the TCP/IP startup procedure. The first four parameters in the PARM parameter let you specify PSF tracing values. For a description of these parameters, refer to *PSF for OS/390: Diagnosis*.

For a list of the printers that can be TCP/IP-attached, refer to *AFP: Printer Information*.

---

### Overview

PSF views TCP/IP-attached printers in the same way as it views SNA-attached printers. PSF performs 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 58 shows.

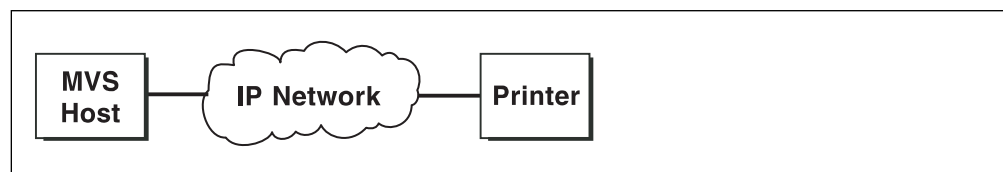


Figure 58. Appearance of the TCP/IP Connection between an MVS Host and a Printer

For example, in communications between an MVS host and a printer, as in Figure 58, the IP network performs like a single network. In fact, however, the network is physically like that shown in Figure 59 on page 244.

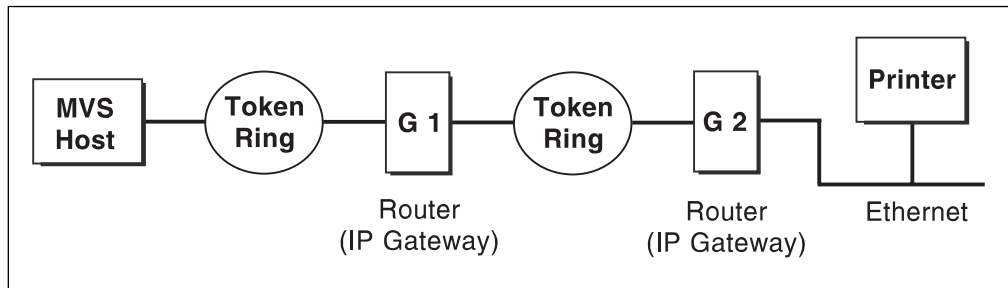


Figure 59. Actual Physical TCP/IP Connection between an MVS Host and a Printer

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

For a more detailed description of TCP/IP networks, refer to the following:

- *Internetworking with TCP/IP, Principles, Protocols, and Architecture*
- *TCP/IP Tutorial and Technical Overview*

## Connecting TCP/IP-Attached Printers

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

1. Define the MVS communications control unit to MVS.
2. Modify the TCP/IP profile on your MVS system, if necessary.
3. Install the i-data 7913 IPDS Printer LAN Attachment, if necessary (see Appendix K, "Installing the i-data 7913 IPDS Printer LAN Attachment" on page 347).
4. Ping the printer; see "Pinging the Printer" on page 246.
5. Define the printer as a writer-controlled printer to JES.
6. Define the printer to PSF with a PRINTDEV statement, including the IP address.

This section does not provide all the information you need to install and configure TCP/IP on your MVS system. For more information on installing TCP/IP, refer to *OS/390 eNetwork Communications Server: IP Configuration Guide*.

## Defining the Communications Control Unit to MVS

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

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

- *MVS/ESA Planning: Dynamic I/O Configuration*
- *OS/390 HCD Planning*
- *OS/390 HCD User's Guide*

## Modifying the TCP/IP Profile in MVS

The TCP/IP profile contains system configuration statements used to initialize the TCP/IP address space. Figure 60 on page 245 shows statements in the TCP/IP 3.2 profile. For information about the profile statements in TCP/IP 3.4 and higher, refer to:

- *OS/390 eNetwork Communications Server: IP Configuration Guide*

- *TCP/IP: Performance Tuning Guide*

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

```

ACBPOOLSIZE          1000
ADDRSTRANSULATIONPOOLSIZE 1500
CCBPOOLSIZE          150
DATABUFFERPOOLSIZE    160 32768
ENVELOPEPOOLSIZE    750
IPROUTEPOOLSIZE     300
LARGEENVELOPEPOOLSIZE 50
RCBPOOLSIZE         50
SCBPOOLSIZE         256
SKCBPOOLSIZE        256
SMALLDATABUFFERPOOLSIZE 256
TCBPOOLSIZE         512
TINYDATABUFFERPOOLSIZE 256
UCBPOOLSIZE         100

KEEPALIVEOPTIONS INTERVAL 10 SENDGARBAGE FALSE ENDKEEPALIVEOPTIONS
GATEWAY
; * Network First hop Linkname Packet size Subnet mask Subnet value
  9          =          BPCLAN  4096      0.255.255.0 0.99.12.0
  DEFAULTNET 9.99.12.254 BPCLAN  4096      0.255.255.0 0

```

Figure 60. Sample Parameters in the TCP/IP 3.2 Profile. This example is **not** a complete TCP/IP profile.

The statements in **bold** in Figure 60 require special considerations when you are printing from PSF on TCP/IP-attached printers:

### **DATABUFFERPOOLSIZE**

The DATABUFFERPOOLSIZE statement defines the number and size of the data buffers. For printing on TCP/IP-attached printers, IBM recommends that you specify at least the following:

- 160 data buffers
- 32 768 buffer size

### **SMALLDATABUFFERPOOLSIZE**

The SMALLDATABUFFERPOOLSIZE statement defines the number of small data buffers. For printing on TCP/IP-attached printers, IBM recommends that you specify at least 256 small data buffers.

### **TINYDATABUFFERPOOLSIZE**

The TINYDATABUFFERPOOLSIZE statement defines the number of tiny data buffers. For printing on TCP/IP-attached printers, IBM recommends that you specify at least 256 tiny data buffers.

### **KEEPALIVEOPTIONS**

The KEEPALIVEOPTIONS statement defines the interval between keep-alive transmissions. 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 has been exchanged between PSF and its connection partner, TCP sends keep-alive probes to the connection partner periodically. These periodic probes, called keep-alive transmissions, enable

TCP 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 frequency of keep-alive transmissions is controlled by the INTERVAL parameter on the KEEPALIVEOPTIONS statement. The frequency applies to all TCP applications that direct TCP to send keep-alive transmissions. The default frequency is after about two hours of inactivity.

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 transmissions.

Also, if any target host requires that the keep-alive packet contain data, specify SENDGARBAGE TRUE. For example:

```
KEEPALIVEOPTIONS INTERVAL 5 SENDGARBAGE TRUE ENDKEEPALIVEOPTIONS
```

### 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 MVS host. The MTU size must not exceed the maximum size that can be sent through the control unit; if it does, transmission problems will occur.

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

- 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 MVS host should 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 or OS/2 system that is running the **bootp** daemon program, which is required for initialization of the i-data 7913 IPDS Printer LAN Attachment.

If you have not already installed and configured the i-data 7913 IPDS Printer LAN Attachment, see Appendix K, “Installing the i-data 7913 IPDS Printer LAN Attachment” on page 347, for information about installing and configuring it.

## Pinging the Printer

To verify that the MVS system can establish a connection with the TCP/IP-attached printer, ping the printer from the MVS 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.

Here is an example of a successful ping:

```
EZA0458I Ping V3R1: Pinging host 9.99.12.33
(Use ATTN to interrupt.)
EZA0463I PING: Ping #1 response took 0.084 seconds.
Successes so far = 1.
```

And here is an example of an unsuccessful ping:

```
EZA0458I Ping V3R1: Pinging host 9.99.12.33
(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 360; otherwise, verify the following:

- The printer is powered on.
- The MVS address is unique in the TCP/IP network.

If the IP address of the MVS 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 MVS system is equal to the MTU size of the printer. For information about the values recommended for the MVS system, see page 246.
  - To change the MTU size for the MVS 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, refer to the documentation for your printer.

If these items are all in order, consult your TCP/IP network administrator about a possible network problem.

### **Ping Is Successful**

A successful ping usually indicates that the MVS 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.<sup>18</sup>

---

<sup>18</sup> If your installation specified a longer interval on the ARPAGE configuration statement in the TCP/IP profile, you may need to wait longer. For information about the ARPAGE statement, refer to *OS/390 eNetwork Communications Server: IP Configuration Guide* .

3. Enter the **ping** command again from the MVS system.

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

## Defining the Printer to JES

To define the printer to JES, see “Sample Initialization Statements for JES2” on page 56 and “Sample Initialization Statements for JES3” on page 61.

## Defining the Printer to PSF

Each TCP/IP-attached printer must be defined to PSF with a PRINTDEV statement in the PSF startup procedure. The following keyword is required on the PRINTDEV statement:

```
IPADDR='ip-address'
```

*ip-address*

Specifies the Internet Protocol (IP) address or host name for the printer or attachment.

Figure 61 on page 249 shows a sample procedure, APSWPROT, shipped with PSF, which you can modify to suit your installation. The PRINTDEV statement shows the required IPADDR keyword.

The APSWPROT procedure contains JCL parameters to produce a startup procedure for either 240-pel or 300-pel printers, but the references to 300-pel resolution are commented out. To create separate startup procedures, make two copies of APSWPROT, and then follow the commented instructions in APSWPROT. Keep one unchanged copy as your startup procedure for a resolution of 120-pel or 240-pel. In the second copy, delete or comment out all references to 240-pel resolution, and make all references to 300-pel resolution active. This changed copy is your startup procedure for 300-pel resolution.

If a printer can print both 240-pel and 300-pel jobs, create two separate FSAs, one for jobs with resolution of 120-pel or 240-pel jobs, and one for jobs with resolution of 300 pels.<sup>19</sup> The 240-pel resolution FSA uses 240-pel resources,<sup>20</sup> and the 300-pel resolution FSA uses 300-pel resources. Both FSAs are defined to drive the same printer; however, only one FSA can be active at one time for a given printer. You should define separate classes or destinations for each FSA. Your installation must route the appropriate jobs to the appropriate class or destination. The two FSAs can be defined in the same or in separate startup procedures.

---

<sup>19</sup> Although doing so is not recommended, you can drive the printer with just one FSA, if one of the following conditions is met:

- All the resources in the resource libraries are resolution-independent, and all the jobs that use inline resources or user libraries contain resources that are resolution-independent.
- All jobs that request a resolution that is different from the resolution-dependent resources in the PSF system and security libraries contain the resources at the requested resolution, either as inline resources or in the user's library.

<sup>20</sup> Even if a PSF user requests 120-pel resolution for an AFP print job being sent to the printer, PSF requires 240-pel resources from the library. Do not store 120-pel resources in the library.

```

//APSWPROT PROC
//***** THE PSF TCP/IP WRITER PROCEDURE *****
//*
//*01* MODULE-NAME = APSWPROT
//* $MOD(APSWPROT) COMP(APS) PROD(PSF) : RELEASE 3.1.0
//*
//*01* DESCRIPTIVE-NAME = START PROCEDURE FOR PSF:
//*
//*                               TCP/IP ATTACHED DEVICES
//*
//*01* STATUS = VERSION 3, RELEASE 1, 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 X060DB. THE
//*             FULL NAME OF THE EQUIVALENT COMPATIBILITY FONT IS
//*             X0GS10.
//*             THE FULL NAME OF THE SEPARATOR PAGE PAGEDEF IS
//*             P1V06483.
//*             THE FULL NAME OF THE SEPARATOR PAGE CORE FONT IS
//*             X060DB. THE FULL NAME OF THE EQUIVALENT COMPATIBILITY
//*             FONT IS X0GT15.
//*
//* REQUIRED ACTIONS =
//*   RESOLUTION - THIS START PROCEDURE IS SET UP FOR DRIVING
//*   A TCP/IP DEVICE AT 240 PEL RESOLUTION. TO CHANGE IT
//*   TO DRIVE 300P PEL RESOLUTION TCP/IP ATTACHED DEVICES:
//*
//*   A) COMMENT OUT THE FIRST PRINTDEV STATEMENT AND UNCOMMENT
//*   THE SECOND PRINTDEV STATEMENT
//*
//*   B) THE FONT02 DATA SET MUST CONTAIN THE CONVERTED
//*   DEFAULT FONTS LISTED IN THE PRINTDEV.
//*   THE FONTS ARE CONVERTED USING THE FONT
//*   CONVERSION UTILITY APSRCF30. SEE THE PSF
//*   SYSTEM PROGRAMMING GUIDE.
//*
//*01* CHANGE-ACTIVITY :
//* $H1=LAPS0001, HAF1103, 080195 DKU1JAU : TCP/IP SUPPORT
//* $DS=LAP0007, HPRF310, 050698 BDKUMAL: DEFAULT TO CORE FONT
//**** END OF SPECIFICATIONS ***/

```

Figure 61 (Part 1 of 2). Sample JCL Startup Procedure for a TCP/IP-Attached Printer (APSWPROT)

```

//STEP01 EXEC PGM=APSPPIEP,REGION=4096K
//JOBHDR OUTPUT PAGEDEF=V06483, /* JOB SEPARATOR PAGEDEF */
// FORMDEF=A10110,CHARS=60DB /* JOB SEPARATOR FORMDEF */
//JOBTLR OUTPUT PAGEDEF=V06483, /* JOB SEPARATOR PAGEDEF */
// FORMDEF=A10110,CHARS=60DB /* JOB SEPARATOR FORMDEF */
//DSHDR OUTPUT PAGEDEF=V06483, /* DS SEPARATOR PAGEDEF */
// FORMDEF=A10110,CHARS=60DB /* DS SEPARATOR FORMDEF */
//MSGDS OUTPUT PAGEDEF=A06462, /* MESSAGE DATASET PAGEDEF */
// FORMDEF=A10110 /* MESSAGE DATASET FORMDEF */
//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 MEDIUM OVERLAYS */
// DISP=SHR /* */
//PDEF01 DD DSN=SYS1.PDEFLIB, /* SYSTEM PAGEDEFS */
// DISP=SHR
//FDEF01 DD DSN=SYS1.FDEFLIB, /* SYSTEM FORMDEFS */
// DISP=SHR
/* ***** */
/* PRINTDEV */
/* ***** */
//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 */
// JOBHDR=*.JOBHDR, /* JOB HEADER SEPARATOR */
/* /* OUTPUT */
// JOBTRLR=*.JOBTLR, /* JOB TRAILER SEPARATOR */
/* /* OUTPUT */
// DSHDR=*.DSHDR, /* DATA SET HEADER */
/* /* SEPARATOR */
// MESSAGE=*.MSGDS, /* MESSAGE DATA SET OUTPUT */
// PAGEDEF=A06462, /* DEVICE PAGEDEF DEFAULT */
// FORMDEF=A10110, /* DEVICE FORMDEF DEFAULT */
// CHARS=(60DB), /* DEVICE DEFAULT FONT */
// PIMSG=YES, /* ACCUMULATE DATA SET */
/* /* MESSAGES */
// DATAACK=BLOCK, /* REPORT ALL DATA-CHECK */
/* /* ERRORS */
// TRACE=YES, /* CREATE INTERNAL TRACE */
// FAILURE=WCONNECT, /* PSF ACTION ON PRINTER */
/* /* FAILURE */
// TIMEOUT=REDRIVE, /* PSF ACTION ON TIMEOUT */
// MGMTMODE=IMMED, /* PRINTER MANAGEMENT MODE */
// DISCINTV=0, /* DISCONNECT INTERVAL IN */
/* /* SECONDS */
// IPADDR='xxx.xxx.xxx.xxx' /* IP ADDRESS FOR TCP/IP */
//PRT1 ENDCNTL

```

Figure 61 (Part 2 of 2). Sample JCL Startup Procedure for a TCP/IP-Attached Printer (APSWPROT)



## Operating a TCP/IP-Attached Printer

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

### Overview of the TCP/IP and PSF Interface

TCP/IP must be running on the MVS 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 TCP/IP abends, 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, PSF also abends because communication is terminated with the printer. The abend is not a restartable abend because there is a severe communication error that must be fixed.
- If an FSA is idle when TCP/IP abends or is stopped, the FSA remains active until a print job is acquired, at which time the FSA abends if TCP/IP is still not active. If TCP/IP has been recycled and is active again, the abend no longer occurs as it did in PSF 2.2.0. PSF now 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 once the link is established.

No TCP/IP services are invoked until a TCP/IP printer is started. At that time, hangs or abends in TCP/IP services could 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 should not be 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 the following:

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 Chapter 7, "Using Deferred-Printing Mode" on page 55.

### 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 following command from the MVS console:

**JES2:**

*\$Pprinter-name*

**JES3:**

\*VARY *printer-name*,OFF

\*CANCEL *printer-name*

where *printer-name* specifies the name of the printer FSA.

Then, you can turn off power to the printer and the i-data 7913 IPDS Printer LAN Attachment, if any.

- 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 may 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 on page 245.

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

```
MODIFY FSSname,FORCE,printer-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 may 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 on page 245.

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

```
MODIFY FSSname,FORCE,printer-name
```

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

---

## Chapter 21. Responding to PSF Errors

PSF messages are formatted and printed in accordance with a page definition and a form definition specified in the PSF startup procedure. If no default page definition and form definition are specified for formatting messages, PSF will 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, refer to *PSF for OS/390: User's Guide*.

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.

Messages not associated with a print data set—for example, messages related to separator-page processing—are sent to the operator's console. All messages contain information about the cause of the error. The messages are explained in *PSF for OS/390: Messages and Codes*.

---

## Accumulating PSF Error Messages

PSF usually accumulates error messages 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 accumulate messages, use the PIMSG parameter, either in the PRINTDEV statement or in the JCL. You can also use the PIMSG parameter to specify that PSF is to stop processing and purge a data set after a certain number of errors, or to redirect the messages; see “Redirecting the Message Data Set” on page 113. For more information on PIMSG, see “Printing PSF Error Messages” on page 114 and Table 2 on page 41. For information about using PIMSG in JCL, refer to *PSF for OS/390: User's Guide*.

---

## Repeating Error Messages

If a paper jam occurs in your printer or if the system operator enters a FORWARD or BACKWARD space command, some error messages may be repeated in the message data set. Usually, these messages indicate:

- Errors detected in the data stream when the page was initially sent to the printer
- Errors repeated when the page was retransmitted

---

## Recovering from PSF Errors

Under JES, PSF monitors pages as they pass through the printer, and records checkpoint data. 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.

For some kinds of error, sense-byte information may 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, refer to the publications for your printer.

If PSF is running in direct-printing mode, error conditions that cause data loss can also cause an abend.

---

## Recovering from Printer-Hardware Problems

PSF provides System Display and Search Facility (SDSF) notification to an operator authorized to receive messages for a printer. However, when the JES SDSF operator receives a PSF error message indicating that a printer-hardware condition requires intervention, the type of intervention is not specified. For more specific information about SDSF, refer to the SDSF publications listed in the “Bibliography” on page 389.

For most other printer-hardware problems, PSF invokes device-dependent error-recovery procedures, issues a message to the operator, and then takes appropriate action. These actions are described in *PSF for OS/390: Messages and Codes*. If operator intervention is required (to clear a paper jam, for example), the action taken depends on the type of printer being used.

If a printer is host-connected and channel-attached, PSF issues a message to the operator console, calling for operator intervention. PSF will not resume processing until the operator takes corrective action.

If the printer has a local-intervention timer, as does the 3820 printer for example, and if the timer expires, PSF issues a message. Action is then taken as defined in the FAILURE parameter of the PRINTDEV statement. You can also set the timer so that it never expires.

For a printer, such as the 3812 printer, that has no local-intervention timer, PSF waits for the printer operator to correct problems as they occur.

---

## Terminating a Printer FSA

If you are unable to purge or cancel a printer by use of the JES command, you can use the PSF operator interface to terminate the printer FSA. The syntax of the MODIFY command for terminating a printer FSA is as follows:

```
{MODIFY|F} FSSname, FORCE ,printer-name
```

To terminate 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.

*FSSname*

Specifies the name of the FSS that manages the printer FSA that is to be terminated. The *FSSname* parameter 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 *printer-name* is to be terminated.

*printer-name*

Specifies the printer FSA that is to be terminated. For JES2, either of two formats, PRINTR*nn* or PRT*nnnn*, can be used for *printer-name*.

**Note:** When processing the FORCE command, PSF issues APS6271.

If the FORCE command is unsuccessful on JES3 systems, use the JES3 command to terminate the printer FSA. That command is as follows:

\*FAIL,PRTXX

---

## Understanding Restartable Abends

PSF refreshes the FSA and restarts it for some abend conditions. (However, PSF does not request a restart more than three times within two hours for an abend code.) This processing enhancement increases the availability of PSF and reduces the need for operator intervention, while retaining the ability to provide diagnostic information.

To determine whether an abend is restartable, refer to *PSF for OS/390: Messages and Codes*. For an explanation of restartable abends, refer to *PSF for OS/390: Diagnosis*.

Because restarting after an abend may cause duplicate pages to print without any notification to the operator or job submitter, you may want to prevent PSF from restarting after abend conditions. See “Considerations on Restartable Abends” on page 204 for information about how to use exit APSUX07 to prevent PSF from restarting after restartable abends.

---

## Requesting Traces and Dumps

PSF provides two types of diagnostic aids: trace reports and formatted dumps. These aids provide a record of events within PSF that can be used in problem determination and diagnosis. For information on using PSF traces and dumps, refer to *PSF for OS/390: Diagnosis*.



---

## Chapter 22. Supporting AFP Output on Microfilm

PSF supports sending Advanced Function Presentation (AFP) data to microfilm <sup>21</sup> devices in deferred-printing mode. PSF does *not* support microfilm for MVS users who are using:

- PSF Direct
- Direct Print mode
- Distributed Print Facility (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 chapter assumes that you know how to configure PSF to enable a PSF user to print AFP data on paper output.<sup>22</sup> On the basis of this assumption, additional prerequisites and information are presented so that you can enable PSF users to also print AFP data to microfilm in the MVS environment. Just as you might need to refer to other documentation for printer-specific information, you may need to refer to documentation supplied by the manufacturer of the microfilm device. For a list of documents about microfilm, see the “Bibliography” on page 389.

---

### PSF Hardware Attachment Requirements for Microfilm Devices

PSF can physically attach to a microfilm device in the following ways, by specific protocols:

- Parallel channel
  - Block multiplexer
  - Data streaming
  - Two-channel switch—optional
- Enterprise Systems Connection (ESCON)
- IBM token ring
  - Protocol—System Network Architecture (SNA) Level 6.2
  - Speed—16 or 4 MBps

---

### PSF System Programmer Tasks for Microfilm Devices

This section presents a visual representation of how PSF and a microfilm device work together to print AFP information to microfilm. The rest of this chapter describes the process by which a user of PSF can be enabled to print AFP jobs to microfilm.

---

<sup>21</sup> *Microfilm* can mean either microfiche or 16 mm film.

<sup>22</sup> If you are not familiar with configuring PSF, IBM recommends that you read the introductory chapter in this publication and the information about defining channel-attached printers, ESCON-attached printers, and SNA-attached printers that use IBM token rings, as well as the chapters that describe working with libraries, changing the system environment, using PRINTDEV 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 may want to check the glossary.

Figure 62 on page 258 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.

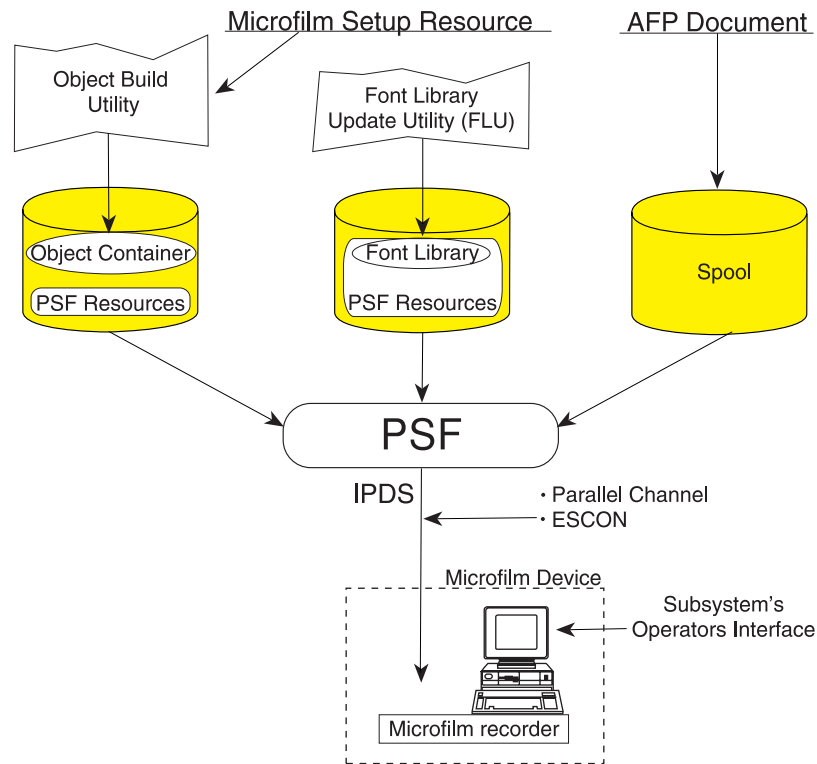


Figure 62. Example of AFP Data Being Transmitted to a Microfilm Device

Even though your system has been configured and installed as instructed by the manufacturer of the microfilm device, you still have some tasks to perform before the microfilm device and PSF can work together to enable PSF users to print AFP jobs on microfilm:

- Attach and define the microfilm device to MVS 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.

With this in mind, you must attach the microfilm device and define it to OS/390 or MVS, JES2 or JES3, and PSF.

## Defining the Microfilm Device to OS/390 or MVS

Whether the microfilm device is channel-attached or ESCON-attached, follow the procedures for attaching and defining channel-attached printers to OS/390 or MVS, as explained in Chapter 2, Defining Channel-Attached Printers to OS/390, by entering **AFP1** as the unit type.

## 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).

Performing the NPRO may take significantly longer on a microfilm device than on a printer; it depends on the microfilm device used.

When specifying a non-zero NPRO value, consider the following:

- While PSF is performing 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 perform 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 specifying an NPRO value of zero, consider the following:

- When PSF receives a request to drain the microfilm device, PSF does not make sure that all pages have been 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 may 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 there are no checkpoints).
- An NPRO is performed 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 sending AFP data to a microfilm device might not want to print header, trailer, or separator pages.
- Specify DATAACK=BLOCK in your PRINTDEV statement to prevent data checks from being reported.

For more information about defining printers to JES, see Chapter 7, Using Deferred-Printing Mode.

## Defining the Microfilm Device to PSF

You should create two separate PRINTDEVs: one for the jobs with 120-pel and 240-pel resolution, and one for those jobs with 300-pel resolution.<sup>23</sup> The 240-pel resolution PRINTDEV uses 240-pel resources,<sup>24</sup> and the 300-pel resolution PRINTDEV uses 300-pel resources. Both PRINTDEVs are defined to drive the same microfilm device; for a given microfilm device, however, only one PRINTDEV can be active at a time. You should 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 63 on page 262 shows the sample startup procedure supplied with PSF for microfilm. It contains all the JCL parameters 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 have been 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.

---

<sup>23</sup> 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.

<sup>24</sup> 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.

## PRINTDEV Parameters for Microfilm Devices

In Figure 63 on page 262, 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=
OBJCONDD=
```

**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 generating member names. IBM recommends that you use **H1** as a prefix to every member name for the microfilm setup resource parameter. In Figure 63, for example, *H1SETUPD* is specified as a default member name for the microfilm setup resource parameter.

**Note:** 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. The microfilm setup resource parameter is used only for sending output to a microfilm device. For syntax rules, see Table 2 on page 41.

As Figure 63 shows, the **COMSETUP=**keyword 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 may 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.

Before libraries are referenced by the **FONTDD**, the Font Conversion Utility needs to be run against them. For additional information, see “Using the Font Library Update (FLU) Utility” on page 265.

**OBJCONDD=***.label* identifies the DD statement specifying the object container library.<sup>25</sup> This keyword follows the rules of any other PSF resource library reference on the PRINTDEV statement; for the syntax, see Table 2 on page 41.

## Sample JCL Startup Procedure for Microfilm Devices

---

<sup>25</sup> 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.

```

//APSWPROM PROC
//***** THE PSF STARTUP PROCEDURE *****
//*
//*01* MODULE-NAME = APSWPROM
//* $MOD(APSWPROM) COMP(APS) PROD(PSF) : RELEASE 3.1.0
//*
//*01* DESCRIPTIVE-NAME = START PROCEDURE FOR MICROFILM
//*                               DEVICES
//*
//*01* STATUS = VERSION 3, RELEASE 1, 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
//*                X050DB.
//*                THE FULL NAME OF THE EQUIVALENT DEFAULT COMPATIBILITY
//*                FONT IS X0GS10.
//*                THE FULL NAME OF THE SEPARATOR PAGE PAGEDEF IS
//*                P1V06483.
//*                THE FULL NAME OF THE SEPARATOR PAGE CORE FONT IS
//*                X060DB.
//*                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 :
//* $00=0W????, HPRF220, 102494 BJ13JPG : Microfilm Support
//*$DR=LAPS0007, HPRF310,050698 BDKUMAL : Default to core font
//*
//**** END OF SPECIFICATIONS ***/

```

Figure 63 (Part 1 of 2). Sample JCL Startup Procedure for a Microfilm Device (APSWPROM)

```

//STEP01 EXEC PGM=APSPPIEP,REGION=4096K
//JOBHDR OUTPUT PAGEDEF=V06483, /* JOB SEPARATOR PAGEDEF */
// FORMDEF=A10110,CHARS=60DB, /* JOB SEPARATOR FORMDEF */
// COMSETUP=H1SETUPS /* JOB SEPARATOR COMSETUP */
//JOBTLR OUTPUT PAGEDEF=V06483, /* JOB SEPARATOR PAGEDEF */
// FORMDEF=A10110,CHARS=60DB, /* JOB SEPARATOR FORMDEF */
// COMSETUP=H1SETUPS /* JOB SEPARATOR COMSETUP */
//DSHDR OUTPUT PAGEDEF=V06483, /* DATA SET SEPARATOR PAGEDEF */
// FORMDEF=A10110,CHARS=60DB, /* 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.FONTLIB, /* 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 */
// OBJCOND=*.OBJCON01, /* OBJECT LIBRARY DD */
// JOBHDR=*.JOBHDR, /* JOB HEADER SEPARATOR OUTPUT */
// JOBTRLR=*.JOBTRLR, /* 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=(60DB), /* DEVICE DEFAULT FONT */
// PIMSG=YES, /* ACCUMULATE DATA SET MESSAGES */
// DATA=BLOCK /* UNBLOCK DATA CHECKS */

```

Figure 63 (Part 2 of 2). Sample JCL Startup Procedure for a Microfilm Device (APSWPROM)

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 67.

## 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 113. In the sample provided, APSWPROM, 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*,<sup>26</sup> which is stored in a PSF-specified host object container library, in a user library, or inline in the AFP data. You should specify a default COMSETUP parameter in the PRINTDEV statement, and the PSF users can also specify the COMSETUP parameter with JCL. The COMSETUP parameter is described in *PSF for OS/390: User's Guide*. If no default COMSETUP keyword is specified in 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 terminates.

You should have already installed or created and named resource libraries that contain resources 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 should 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 Font Libraries

Every time a 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 265.

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 63 on page 262—identifies both FONT01 (to support 120-pel and 240-pel resolution) and FONT02 (to support 300-pel resolution) for printing to microfilm.

---

<sup>26</sup> 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 265.

---

## Managing Resources and Selecting Special Processing Parameters

Microfilm support requires the following tasks, in addition to those 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 provided by the manufacturer of your microfilm device to generate microfilm setup resource object containers.<sup>27</sup> For more information about using a utility to generate object containers, refer to the instructions provided with your microfilm device.

The utility provided with the microfilm device checks the syntax of the setup parameters and places the setup information into a microfilm setup resource object container,<sup>28</sup> 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 4, *Working with Libraries*.

### Using the Font Library Update (FLU) Utility

Microfilm devices require that all fonts downloaded from the host contain Global Resource ID (GRID) information. If a requested font does not contain GRID information, the page or data set using that font is not printed.

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 usually 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 on how to make a Name-to-GRID user mapping table, see “Using the Font Mapping Tables” on page 150). 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.

---

<sup>27</sup> The utility that the manufacturer of your microfilm device provides is also supported by PSF/6000 in the AIX operating system. Microfilm setup resource object containers generated in either of these operating systems can be used by the other operating system.

<sup>28</sup> The object container is not an AFP-managed resource; it is a MO:DCA architecture resource.

## 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<sup>29</sup>—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 will be 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 has run 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 should be attempting to access the fonts, and no PSF FSA should be ACTIVE, while the FLU is running.
- The attributes of all font data sets must be as follows (the same requirements 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.

## Running the FLU

A sample job, APSWRFLU, is provided in SYS1.SAMPLIB (see Figure 64 on page 267). 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.

---

<sup>29</sup> The number of font data sets that can be input to the FLU on a single invocation is limited only by the operating system.



```

//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 64. 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 following DDs would be 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

```

## FLU Return Codes

The FLU returns only three valid return codes:

- **0:** The FLU has run successfully, and the appropriate font members have been updated. For information on the updates performed, see the informational messages in the SYSPRINT data set.
- **4:** The FLU updated all the members it could, but found fonts that were invalid and could not be updated. The invalid fonts cannot be used with the microfilm device until they are fixed and the FLU is run again. For information on the font members in error and the updates 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 on 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 in the event of a logic error, and it aborts processing. Fix the error and rerun the FLU.

---

## Printing to Microfilm

After you have attached the microfilm device, provided a startup procedure, run the utility 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 desired resolution.

To switch from one startup procedure (or PRINTDEV) to the other, use the following procedure:

- Drain the active PRINTDEV by using a normal printer stop command—`$PPRTxxx` in JES2, or `*V,PRTxxx,OFFLINE` in JES3.
- Start the other PRINTDEV by using a normal printer start command—`$SPRTyyy` 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 powering 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 perform 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 have been made for printing to microfilm. The microfilm device is handled in the same way as an AFP1 family printer. For information, see Chapter 13, Understanding PSF Data Protection and Security.

---

## 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 18, Using Installation Exits.

---

## 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 21, Responding to PSF Errors.

---

## **Recovering or Restarting AFP Print Jobs for Microfilm**

The restart and rerun processes have joint PSF and microfilm device interactions. Refer to 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. APSRMARK Sample Jobs

This appendix lists the APSRMARK sample jobs that are shipped with PSF. Table 19 lists the SYS1.SAMPLIB member names and associates each sample job with a corresponding printer resource diskette or cartridge.

| <i>Table 19. Sample Jobs in SYS1.SAMPLIB</i> |  |  |
|--|--|--|
| <b>Member name</b>                           | <b>Description</b>   | <b>Printer that <i>might</i> have these resident (some are optional font cards)</b>  |
| APSWMCPG                                     | AFCCU code pages   | Any AFCCU printer, 3930 printer (for a list of AFCCU printers, refer to <i>Advanced Function Presentation: Printer Information</i> ) |
| 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   |

### Notes:

1. None of the sample jobs listed in Table 19 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 may 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, using APSRCF30, and then mark it with these APSRMARK jobs. For more information, see “Using APSRMARK” on page 137.

## Appendix B. APSRMARK Long Report Format

Figure 65 shows an example of a long report format for resources marked with APSRMARK.

```

AP5580I      INDD=OUT2, MEMBER=ALL, REPORT, LONG
AP5580I 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=L111111.L222222.L333333.L444444.L555555
ENABLED FOR:   CAPTURE(DPF)=NO CAPTURE(PRINTER)=NO RESIDENT RESOURCE ACTIVATION=NO
AP5580I 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=L111111.L222222.L333333.L444444.L555555
ENABLED FOR:   CAPTURE(DPF)=YES CAPTURE(PRINTER)=YES RESIDENT RESOURCE ACTIVATION=YES
AP5580I COFACTRY 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
AP5580I CONOMARK 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
AP5580I FINOBOX RESOURCE TYPE: FORMDEF
ENABLED FOR:   CAPTURE(DPF)=NO CAPTURE(PRINTER)=NO RESIDENT RESOURCE ACTIVATION=NO
AP5580I M1UBLPK RESOURCE TYPE: OBJECT CONTAINER (COLOR MAPPING TABLE)
ENABLED FOR:   CAPTURE(DPF)=NO CAPTURE(PRINTER)=NO RESIDENT RESOURCE ACTIVATION=NO
AP5580I 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
AP5580I P1SEC5 RESOURCE TYPE: PAGEDEF
ENABLED FOR:   CAPTURE(DPF)=NO CAPTURE(PRINTER)=NO RESIDENT RESOURCE ACTIVATION=NO
AP5580I 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
AP5580I 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=L111111.L222222.L333333.L444444.L555555
ENABLED FOR:   CAPTURE(DPF)=YES CAPTURE(PRINTER)=NO RESIDENT RESOURCE ACTIVATION=NO

```

Figure 65 (Part 1 of 2). Printed Output of Printer-Resident Long Report

```

APS580I TIMYNEW1 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

```

Figure 65 (Part 2 of 2). Printed Output of Printer-Resident Long Report

An APSRMARK long report format contains four areas of data for each resource object:

| <b>Area Name</b>        | <b>Description</b>   |
|-------------------------|--|
| <b>Message number</b>   | The message number is contained in columns 1 to 7 and defines each new resource object listed in the report. All resource objects listed in the long report have the message number APS580I.   |
| <b>Resource name</b>    | The resource name is contained in columns 10 to 17 and identifies the resource object.   |
| <b>Data category</b>    | The data category is contained in columns 19 to 32 and indicates the type of data reported for a resource object. The data categories listed for a resource object are determined by its resource type. See "Data Categories" on page 275 for information about the different data categories. |
| <b>Data description</b> | The data description is contained in columns 34 to 122 and describes the settings for a data category. The settings for a resource object determine resident resource activation and resource capture.   |

Table 20 on page 275 shows examples of the four data areas on a long report.



Table 20. Data Areas on a 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<br>DSN=L1111111.L2222222.L3333333.L4444444.L5555555 |
| APS580I        | F1NOBOX       | RESOURCE TYPE: | FORMDEF  |
|                |               | ENABLED FOR:   | CAPTURE(DPF)=NO CAPTURE(PRINTER)=NO RESIDENT<br>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<br>DSN=DOSCHE.PRIVATE.MARKED.PSEGMENT.RESOURCE      |
|                |               | ENABLED FOR:   | CAPTURE(DPF)=YES CAPTURE(PRINTER)=NO RESIDENT<br>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 resource objects on the report. The other categories listed are determined by the resource type (see "Resource Type" on page 276).

## 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.

### **OBJECT CONTAINER** (*object container type*)

*object container type*

Specifies the type of object container. For example:

- COLOR MAPPING TABLE
- COMSETUP

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.

**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 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 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 shipped 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 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 shipped 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 using APSRMARK. The settings 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 Facility (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 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 shipped 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 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 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 listed in the data description area are:

**SYSTEM=nnnnnnnnnnnn**

Identifies the physical system where APSRMARK ran the last time the resource was marked.

**OS=xxx**

Identifies the operating system where APSRMARK ran the last time the resource was marked. MVS indicates MVS, MVS/ESA, or OS/390.

**VOLUME=xxxxxx**

Identifies the physical volume serial number for the device containing the PDS with this resource when APSRMARK was last used to mark the resource.

**DSN=xxxxxxxx.xxxxxxxxx.xxxxxxxxx.xxxxxxxxx.xxxxxxxxx**

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 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 21 indicates how the Enabled For settings are determined. All conditions listed in the table must be true for the setting to equal YES; otherwise, the setting equals NO.

Table 21 (Page 1 of 2). Enable For Settings

| Resource Type      | CAPTURE(DPF)=YES  | CAPTURE(PRINTER)=YES   | RESIDENT RESOURCE ACTIVATION=YES   |
|--------------------|---|--|--|
| FONT CHARACTER SET | <ul style="list-style-type: none"><li>The font is RASTER.</li><li>All Time(Marked) data is present.</li><li>All Management data is available and PUBLIC=YES.</li><li>All Object Origin data is present.</li></ul> | <p>OUTLINE:</p> <ul style="list-style-type: none"><li>The font is OUTLINE.</li><li>FGID data is specified for Internal ID, Change Level, or both, and is not zero.</li><li>Date and time data is available for Time(Created), Time(Revised), or Change Level.</li><li>PUBLIC=YES for Management.</li></ul> <p>RASTER:</p> <ul style="list-style-type: none"><li>The font is RASTER.</li><li>FGID data is specified for Internal ID, Change Level, or both, and is not zero.</li><li>WIDTH data is specified for Internal ID, Change Level, or both, and is not zero.</li><li>Date and time data is available for Time(Created), Time(Revised), or Change Level.</li><li>PUBLIC=YES for Management.</li></ul> | <p>OUTLINE:</p> <ul style="list-style-type: none"><li>The font is OUTLINE.</li><li>FGID data is specified for Internal ID, Change Level, or both, and is not zero.</li><li>PUBLIC=YES for Management.</li></ul> <p>RASTER:</p> <ul style="list-style-type: none"><li>The font is RASTER.</li><li>FGID data is specified for Internal ID, Change Level, or both, and is not zero.</li><li>WIDTH data is specified for Internal ID, Change Level, or both, and is not zero.</li><li>PUBLIC=YES for Management.</li></ul> |

Table 21 (Page 2 of 2). Enable For Settings

| Resource Type            | CAPTURE(DPF)=YES  | CAPTURE(PRINTER)=YES  | RESIDENT RESOURCE ACTIVATION=YES   |
|--------------------------|---|---|--|
| CODE PAGE                | <ul style="list-style-type: none"> <li>All Time(Marked) data is present.</li> <li>All Management data is available and PUBLIC=YES.</li> <li>All Object Origin data is present.</li> </ul> | <ul style="list-style-type: none"> <li>GCSGID data is specified for Internal ID, Change Level, or both, and is not zero.</li> <li>CPGID data is specified for Internal ID, Change Level, or both, and is not zero.</li> <li>Date and time data is available for Time(Created), Time(Revised), or Change Level.</li> <li>PUBLIC=YES for Management.</li> </ul> <p><b>Note:</b> Even if GCSGID data is not present, the CODE PAGE resource may 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.</p> | <ul style="list-style-type: none"> <li>GCSGID data is specified for Internal ID, Change Level, or both, and is not zero.</li> <li>CPGID data is specified for Internal ID, Change Level, or both, and is not zero.</li> <li>PUBLIC=YES for Management.</li> </ul> <p><b>Note:</b> It is possible to have a CODE PAGE that does not have a GCSGID and is therefore 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 resident resource activation can take place.</p> |
| OVERLAY and PAGE SEGMENT | <ul style="list-style-type: none"> <li>All Time(Marked) data is present.</li> <li>All Management data is available and PUBLIC=YES.</li> <li>All Object Origin data is present.</li> </ul> | No conditions are true; setting always equals NO.   | No conditions are true; setting always equals NO.  |

**Note:** PSF for OS/390 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 irrespective of the effect the Font Mapping Table support has on the actual font processing and font substitution.



---

## Appendix C. Font Resource Tables Supplied by PSF

This appendix contains information for intended Programming Interfaces.

The following font resource tables, which are provided with PSF, are intended as samples. The routines as provided may not be suitable for your requirements.

The APSRFTBL font resource table module is provided on the PSF distribution tapes.

---

### APSRFTBL Font Resource Table

Figure 66 on page 284 is a copy of the module APSRFTBL, which is provided in SYS1.SAMPLIB. The host code pages and character sets listed in this font resource table are equivalent to the resident symbol sets in the 4224, 4230, 4234, 4247, and 64XX printers, and should be 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, 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 specified in the Map Coded Font structured field, PSF terminates the current page and issues messages saying that the requested resource could not 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.

```

TITLE:  APSRFTBL RESIDENT FONT RESOURCE TABLES
APSRFTBL START 0
*/****START OF SPECIFICATIONS*****EXTPROC******/
**                                          */
**      MODULE NAME: APSRFTBL              */
**                                          */
**      DESCRIPTIVE NAME: RESIDENT FONT RESOURCE TABLES      */
**                                          */
**      FUNCTION: THIS TABLE ENABLES PSF/MVS 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.    @05C*/
**                                          */
**      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: 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 PRINTERS WHEN RUN IN 4234  */
**                  EMULATION MODE. IT CAN BE CHANGED          @05A*/
**                  BY THE USER. THE TABLE IS TERMINATED BY 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 PRINTERS WHEN RUN IN 4234 EMULATION MODE. */
**                  IT CAN BE CHANGED BY THE USER.            @05A*/
**                  THE TABLE IS TERMINATED BY A STRING OF    */
**                  CL8'FFFFFFFF'.                               */
**      LINKAGE: NONE                               */
**                                          */

```

Figure 66 (Part 1 of 10). APSRFTBL Font Resource Table

```

**/ ** 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 PRINTER. IT CAN BE CHANGED */
**/ ** BY THE USER. THE TABLE IS TERMINATED BY A */
**/ ** A STRING OF CL8'FFFFFFFF'. */
**/ ** LINKAGE: NONE */
**/ ** */
**/ ** ENTRY POINT: APSCP64X @03A*/
**/ ** PURPOSE: THIS IS A TABLE OF RESIDENT CODE PAGES IN THE */
**/ ** 64XX PRINTER. IT CAN BE CHANGED BY THE USER. */
**/ ** THE TABLE IS TERMINATED BY A STRING OF */
**/ ** CL8'FFFFFFFF'. */
**/ ** LINKAGE: NONE */
**/ ** */
**/ ** 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 - 6408/6412 RESIDENT CHAR SET MAPPING TABLE @03A*/
**/ ** APSCP64X - 6408/6412 RESIDENT CODE PAGE MAPPING TABLE @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 CODEPAGES 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=OW35644, HPRF220, 981028, BDKUEAS: EURO SUPPORT FOR IMPACT */
**/ ** PRINTERS @05A*/
**/ ** */
**/ ** *****END OF SPECIFICATIONS*****

```

Figure 66 (Part 2 of 10). APSRFTBL Font Resource Table

```

*/*****
*/
*/
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)." *
*****
          ENTRY APSCS24V          DO NOT REMOVE/CHANGE
APSCS24V DS  0D          DO NOT REMOVE/CHANGE
DC CL8'C0L00B0A',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'C0L00A0A',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
DC CL8'FFFFFFFF',XL2'FFFF',XL2'FFFF' DO NOT REMOVE/CHANGE

```

Figure 66 (Part 3 of 10). APSRFTBL Font Resource Table

```

*****
* 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
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 PAGE 924 IS TO BE USED ONLY WITH THE 4247 @05A*
* PRINTER WHEN RUN IN 4224 EMULATION MODE @05A*
*****
DC CL8'T1L00924',XL2'039C' CODE PAGE 924 @05A
DC CL8'FFFFFFFF',XL2'FFFF' DO NOT REMOVE/CHANGE

```

Figure 66 (Part 4 of 10). APSRFTBL Font Resource Table

```

*****
* THE FOLLOWING IS THE CHARACTER SET TABLE FOR THE 4234 PRINTER. IT *
* IS ALSO USED FOR THE 64XX PRINTERS WHEN RUNNING IN 4234 EMULATION *
* MODE. THE FIRST EIGHT (8) BYTES OF EACH ENTRY IS THE NAME OF @05A*
* 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)." *
*****
ENTRY APSCS34V DO NOT REMOVE/CHANGE
APSCS34V DS 0D DO NOT REMOVE/CHANGE
DC CL8'C0L00B0A',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'C0L00A0A',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

```

Figure 66 (Part 5 of 10). APSRFTBL Font Resource Table

```

*****
* THE FOLLOWING IS THE CODE PAGE TABLE FOR THE 4234 PRINTER. IT IS *
* ALSO USED FOR THE 64XX PRINTERS WHEN RUNNING IN 4224 EMULATION MODE.*
* THE FIRST EIGHT (8) BYTES OF EACH ENTRY IS THE NAME OF THE CODE @05A*
* PAGE. THE NEXT TWO (2) BYTE FIELD IS THE GLOBAL ID OF THE CORRE- *
* SPONDING 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'T1L0OCR1',XL2'037C' CODE PAGE 892
DC CL8'T1L0OCRB',XL2'037D' CODE PAGE 893
*****
* 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

```

Figure 66 (Part 6 of 10). APSRFTBL Font Resource Table

```

*****
* 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 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. @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'C0S0CB10',XL2'000B',XL2'0002' COURIERBOLD10 @04P
DC CL8'C0S0CI10',XL2'000B',XL2'0008' COURIERITAL10 @04P
DC CL8'C0L00AOA',XL2'0013',XL2'0000' OCR-A @04P
DC CL8'C0L00KATA',XL2'001A',XL2'0000' KATAKANA @04P
DC CL8'C0S0AE10',XL2'001A',XL2'0000' APL 10 @04P
DC CL8'C0D0GT10',XL2'001A',XL2'0000' GOTHIC TEXT10 @04P
DC CL8'C0D0GB10',XL2'001A',XL2'0002' GOTHIC BOLD10 @04P
DC CL8'C0S0CR12',XL2'0055',XL2'0000' COURIER 12 @04P
DC CL8'C0S0CB12',XL2'0055',XL2'0002' COURIER BOLD12@04P
DC CL8'C0S0CI12',XL2'0055',XL2'0008' COURIER ITAL12@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 TXT 15 @04P
DC CL8'C0S0CR15',XL2'00DF',XL2'0000' COURIER 15 @04P
DC CL8'C0S0CB15',XL2'00DF',XL2'0002' COURIER BOLD15@04P
DC CL8'C0S0CI15',XL2'00DF',XL2'0008' COURIER ITAL15@04P
DC CL8'C0S0CD15',XL2'00DF',XL2'0001' COURDBL-WID 15@04P
DC CL8'C0S0CW15',XL2'00DF',XL2'0009' COURDBL-WIDITA@04P
DC CL8'FFFFFFFF',XL2'FFFF',XL2'FFFF' DO NOT REMOVE

```

Figure 66 (Part 7 of 10). APSRFTBL Font Resource Table



```

*****
* 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
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'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'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'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'T1V10500',XL2'01F4' CODE PAGE 500 @04A
DC CL8'T1000803',XL2'0323' CODE PAGE 803 @04A
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'T1000892',XL2'037C' CODE PAGE 892 @04P
DC CL8'T1L00CR1',XL2'037C' CODE PAGE 892 @04P
DC CL8'T1000893',XL2'037D' CODE PAGE 893 @04P
DC CL8'T1L00CRB',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'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 66 (Part 8 of 10). APSRFTBL Font Resource Table

```

*****
* THE FOLLOWING IS THE CHARACTER SET TABLE FOR THE 64XX 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.@03C*
*****
ENTRY APSCS64X DO NOT REMOVE/CHANGE
APSCS64X DS 0D DO NOT REMOVE/CHANGE
DC CL8'C0L00B0A',XL2'0003',XL2'0000' OCR-B @03P
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'C0L00A0A',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 66 (Part 9 of 10). APSRFTBL Font Resource Table

```

*****
* THE FOLLOWING IS THE CODE PAGE TABLE FOR THE 64XX 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 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'T1L00CR1',XL2'037C' CODE PAGE 892 @03P
DC CL8'T1000893',XL2'037D' CODE PAGE 893 @03A
DC CL8'T1L00CRB',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 66 (Part 10 of 10). APSRFTBL Font Resource Table



---

## Appendix D. Field Descriptions for APSUX07

This appendix describes APSUX07 parameter areas and fields, which contain information for intended Programming Interfaces. The appendix presents the following:

- 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 APSUX07 needs in order 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 on APSGEXTP, see Figure 50 on page 171.

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.

Figure 67 on page 296 shows the format of XTP7.

| Offsets | Type                       | Length | Name     | Description  |
|---------|----------------------------|--------|----------|--|
| 0       | (0) CHARACTER              | *      | XTP7     |  |
| 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 APSUX07 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  | DELETION AT DATA SET END   |
|         | .... ..1.                  |        | XTP7TERM | TERMINATION OF FSA   |
|         | .... ...1                  |        | *        | RESERVED   |
| 10      | 1111 1111<br>(A) BITSTRING | 2      | XTP7RTYP | RESERVED<br>FLAGS SET BY PSF-INDICATES TO APSUX07 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 APSUX07 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   |
| 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' AND BDSC 'XTP7BDS'.  |
| 15      | (F) UNSIGNED               | 1      | XTP7IPCC | NUMBER OF COPIES OF PAGE TO PRODUCE.<br>- VALID ONLY WITH THE INTERRUPT MESSAGE XTP7IPMF.        |
| 16      | (10) BITSTRING             | 4      | XTP7NOTY | NOTIFICATION FLAGS SET BY APSUX07, 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   |
|         | ..11 1111                  |        | *        | RESERVED   |

Figure 67 (Part 1 of 6). XTP7 Parameter Area Format

|    |                |   |          |  |
|----|----------------|---|----------|--|
| 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   |
| 20 | (14) BITSTRING | 2 | XTP7RFLG | RESPONSE FLAGS   |
|    | 1... ....      |   | XTP7ETFF | EJECT TO FRONT FACING BETWEEN DOCUMENTS:<br>(VALID FOR INITIALIZATION 'XTP7INIT' AND BDSC 'XTP7BDS')<br>0 = ETFF ACTIVE<br>1 = BYPASS ETFF   |
|    | .1.. ....      |   | XTP7TDS  | TERMINATE THE DATA SET<br>0 = DO NOT TERMINATE<br>1 = TERMINATE  |
|    | ..1. ....      |   | XTP7EFFH | EJECT TO FRONT FACING BEFORE JOB HEADER:<br>(VALID FOR INITIALIZATION 'XTP7INIT' AND BDSC 'XTP7BDS')<br>0 = ETFF ACTIVE<br>1 = BYPASS ETFF   |
|    | ...1 ....      |   | XTP7HDF  | SELECT HARDWARE DEFAULT FONT (VALID FOR 3800 ONLY):<br>0 = USE HARDWARE DEFAULT<br>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):<br>0 = PURGE THE DATA SET<br>1 = MARK DATA SET AS UNPRINTABLE                                    |
|    | .... .1..      |   | XTP7PRNS | SINGLE BYTE FONT PRUNING FLAG (VALID ONLY ON INITIALIZATION CALL 'XTP7INIT')<br>0 = PRUNE SINGLE BYTE FONTS<br>1 = BYPASS PRUNING  |
|    | .... ..1.      |   | XTP7PRND | DOUBLE BYTE FONT PRUNING FLAG (VALID ONLY ON INITIALIZATION CALL 'XTP7INIT')<br>0 = PRUNE DOUBLE BYTE FONTS<br>1 = BYPASS PRUNING  |
|    | .... ...1      |   | XTP7IPMF | MARK FORM REQUEST (VALID FOR INITIALIZATION 'XTP7INIT' AND BDSC 'XTP7BDS')<br>SEE XTP7IPCC FOR COPIES.<br>0 = NO MARKING OF INTERRUPT MESSAGE PAGE<br>1 = MARK INTERRUPT MESSAGE PAGE MESSAGE PAGE |

Figure 67 (Part 2 of 6). XTP7 Parameter Area Format

|           |                |             |   |
|-----------|----------------|-------------|---|
| 1... ..   |                | XTP7IPOS    | OFFSET STACK INTERRUPT MESSAGE PAGE.<br>(VALID FOR INITIALIZATION 'XTP7INIT' AND BDSC 'XTP7BDS')<br>0 = NO OFFSET STACKING<br>1 = OFFSET STACKING   |
| .1.. ..   |                | XTP7MTOF    | MAP FONTS TO OUTLINE FONTS<br>(VALID ONLY ON INITIALIZATION CALL 'XTP7INIT')<br>0 = DO NOT MAP FONTS TO OUTLINE FONTS<br>1 = MAP FONTS TO OUTLINE FONTS   |
| ..1. .... |                | XTP7RFNF    | RECOVER FROM FONT-NOT-FOUND CONDITIONS (VALID ONLY ON INITIALIZATION CALL 'XTP7INIT').<br>0 = DO NOT QUERY HOST FONT LIBRARIES TO ENSURE THAT THE MAPPED FONT ACTUALLY EXISTS.<br>1 = QUERY THE HOST FONT LIBRARIES FOR THE MAPPED FONT. IF THE MAPPED DOES NOT EXIST, GO BACK AND USE THE ORIGINAL FONT.<br>NOTE: DUE TO THE ADDITIONAL LIBRARY QUERIES FOR EVERY FONT RESOURCE, PERFORMANCE WILL BE DEGRADED IF THIS OPTION IS TURNED ON. |
| ...1 .... |                | XTP7IPSU    | SUPPRESS INTERRUPT MESSAGE PAGES (VALID FOR INITIALIZATION 'XTP7INIT')<br>0 = ALLOW INTERRUPT MSG PAGES<br>1 = SUPPRESS INTERRUPT MSG PAGES   |
| .... 1..  |                | XTP738M0    | SET THE MEDIA ORIGIN TO THE 3800 MEDIA ORIGIN FOR THE USER'S DATA SET<br>(VALID FOR INITIALIZATION, 'XTP7INIT' AND BDSC 'XTP7BDS')<br>0 = DO NOT SET MEDIA ORIGIN TO 3800 MEDIA ORIGIN<br>1 = SET MEDIA ORIGIN TO 3800 MEDIA ORIGIN   |
| .... .1.. |                | XTP7MDS0    | SEND THE MESSAGE DATA SET TO A SYSOUT DATA SET (VALID FOR INITIALIZATION 'XTP7INIT' AND BDSC 'XTP7BDS')<br>0 = DO NOT SEND MSG DATA SET TO A SYSOUT DATA SET<br>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<br>(VALID FOR INITIALIZATION 'XTP7INIT' AND BDSC 'XTP7BDS')<br>0 = CONVERT TO MULTIPLE<br>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   |

Figure 67 (Part 3 of 6). XTP7 Parameter Area Format



|     |                |    |          |  |
|-----|----------------|----|----------|--|
| 28  | (1C) SIGNED    | 2  | XTP7RPS  | RRLV FOR PAGE SEGMENTS   |
| 30  | (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   |
| 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 ....      |    | XTP7CFM0 | 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      |    | XTP7CSM0 | FREE/KEEP SECUR. OVLY DEFAULT CRL                                |
| 79  | (4F) BITSTRING | 1  | XTP7CRF2 | CRL FLAGS  |
|     |                |    |          | 0 = FREE STORAGE   |
|     |                |    |          | 1 = KEEP THE CRL   |
|     | 1... ....      |    | XTP7CSPS | FREE/KEEP SECURITY 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..       |    | XTP7C2M0 | FREE/KEEP OVERLAY 240 CRL  |
|     | .... .1..      |    | XTP7C3M0 | 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  | XTP7CLM0 | 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                                 |

Figure 67 (Part 4 of 6). XTP7 Parameter Area Format

|     |                |   |          |  |
|-----|----------------|---|----------|--|
| 112 | (70) ADDRESS   | 4 | XTP7CLS0 | 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' AND BDSC 'XTP7BDS')      |
|     |                |   |          | 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' AND BDSC 'XTP7BDS')     |
|     |                |   |          | CORNER   |
|     |                |   |          | 0 = DO NOT SET JOB TRAILER MEDIA ORIGIN  |
|     |                |   |          | 1 = SET JOB TRAILER MEDIA ORIGIN TO TOP LEFT CORNER  |
|     | ..1. ....      |   | XTP7DHMO | SET THE DATA SET HEADER MEDIA ORIGIN TO TOP LEFT CORNER (VALID FOR INITIALIZATION 'XTP7INIT' AND BDSC 'XTP7BDS') |
|     |                |   |          | 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' AND BDSC 'XTP7BDS') |
|     |                |   |          | 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   |

Figure 67 (Part 5 of 6). XTP7 Parameter Area Format

|     |                |    |          |   |
|-----|----------------|----|----------|---|
|     | .... .1..      |    | XTP7WIDE | WIDE OR NARROW PAPER FLAG -<br>(VALID FOR INITIALIZATION "XTP7INIT"<br>AND BDSC "XTP7BDS"). THIS FLAG IS<br>ONLY VALID WHEN ONE OR MORE OF THE<br>3800 MEDIA ORIGIN FLAGS IS SET AND<br>THE PRINTER IS IN CUT-SHEET<br>EMULATION MODE:<br>1 = LANDSCAPE PAGES ARE GENERATED<br>FOR PRINTING WHEN IN CUT-SHEET<br>EMULATION MODE.<br>0 = PORTRAIT PAGES ARE GENERATED<br>FOR PRINTING WHEN IN CUT-SHEET<br>EMULATION MODE. |
|     | .... .1.       |    | XTP7IHIB | INHIBIT RECOVERY FLAG (VALID FOR<br>INITIALIZATION "XTP7INIT" AND BDS<br>"XTP7BDS" FOR FIRST DATA SET OF A<br>JOB):<br>0 = ALLOW RECOVERY.<br>1 = INHIBIT RECOVERY.   |
|     | .... ...1      |    | XTP7ENDC | END THE SNA LU1 CONVERSATION (VALID<br>ONLY ON INITIALIZATION CALL<br>"XTP7INIT"):<br>1 = END THE SNA LU1 CONVERSATION<br>WITHOUT ENDING THE SESSION.<br>0 = ONLY END THE SNA CONVERSATION<br>WHEN THE SESSION ENDS.  |
| 121 | (79) CHARACTER | 2  | XTP7ABRC | ABEND REASON CODE<br>(VALID ONLY ON TERMINATION<br>CALL 'XTP7TERM')   |
| 123 | (7B) CHARACTER | 1  | *        | RESERVED  |
| 124 | (7C) UNSIGNED  | 2  | XTP7FMT  | CURRENT DATA SET RESOLUTION VALUE SET BY<br>APSUX07   |
| 126 | (7E) BITSTRING | 1  | XTP7CRF3 | CRL FLAGS<br>0 = FREE STORAGE<br>1 = KEEP THE CRL   |
|     | 1... ....      |    | *        | RESERVED  |
|     | .1.. ....      |    | *        | RESERVED  |
|     | ..1. ....      |    | XTP7S2CF | FREE/KEEP SECUR. FONT 240 CRL   |
|     | ...1 ....      |    | XTP7S3CF | FREE/KEEP SECUR. FONT 300 CRL   |
|     | .... 1...      |    | XTP7S2M0 | FREE/KEEP SECUR. OVLY 240 CRL   |
|     | .... .1..      |    | XTP7S2PS | FREE/KEEP SECUR. SEG 240 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<br>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  | XTP72LM0 | POINTER TO OVLY 240 CRL   |
| 140 | (8C) ADDRESS   | 4  | XTP73LM0 | POINTER TO OVLY 300 CRL   |
| 144 | (90) ADDRESS   | 4  | XTP72LPS | POINTER TO SEGM/OCA 240CRL  |
| 148 | (94) ADDRESS   | 4  | XTP73LPS | POINTER TO SEGM/OCA 300 CRL   |
| 152 | (98) ADDRESS   | 4  | *        | RESERVED  |
| 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  | XTP72LS0 | POINTER TO SECUR OVLY 240 CRL   |
| 172 | (AC) ADDRESS   | 4  | XT073LS0 | 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) CHARACTER | 1  | XTP7MFG1 | MORE RESPONSE FLAGS   |
|     | 1... ....      |    | XTP7APSF | ACKNOWLEDGEMENT PER SHEET<br>FLAG (VALID ONLY ON<br>INITIALIZATION CALL "XTP7INIT")<br>0 = ACK PER PAGE<br>1 = ACK PER SHEET  |
| 185 | (B9) CHARACTER | 24 | *        | RESERVED  |

Figure 67 (Part 6 of 6). XTP7 Parameter Area Format

## XTP7 Fields

|                 |   |
|-----------------|---|
| <b>XTP7CID</b>  | Contains the control block identifier XTP7.   |
| <b>XTP7LENG</b> | Contains the length of the XTP7 DSECT.  |
| <b>XTP7ETYP</b> | 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.  |
| <b>XTP7RTYP</b> | Information flags set by PSF to indicate the type of resource being 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, and TERM), all XTP7RTYP flags are set to '0'.   |
| <b>XTP7DSTY</b> | 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'.  |
| <b>XTP7AUX</b>  | PSF is processing auxiliary resources. This includes all resources 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.  |
| <b>XTP7PDS</b>  | PSF is processing a user's print data set.  |
| <b>XTP7DSAT</b> | 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.   |
| <b>XTP7PDFT</b> | PSF is processing the default page definition or form definition.   |
| <b>XTP7PJHD</b> | PSF is processing the job header.   |
| <b>XTP7PJTR</b> | PSF is processing the job trailer.  |
| <b>XTP7PDSH</b> | PSF is processing the data set header.  |
| <b>XTP7PMDS</b> | PSF is processing the message data set.   |
| <b>XTP7PLDS</b> | PSF is processing the last print data set in the job; a job trailer will be printed.  |
| <b>XTP7ACSE</b> | Valid only on the INIT and BDSC calls to APSUX07 when CSE is being used. PSF initially sets this field to X'00'. This value causes PSF to not check whether a page fits 2 up on the sheet. If this field is set to X'01', PSF checks the first page printed, using a new copy group, to see if it fits 2 up on the sheet. If this field is set to X'02', PSF checks all the front side pages to see if they fit 2 up on the sheet. If XTP7ACSE is set to any other value, it defaults to X'00'. |
| <b>XTP7IPCC</b> | Valid only on the INIT and BDSC calls to APSUX07. PSF initially sets this field to X'00', which causes 1 copy of the interrupt message page to be printed. If flag XTP7IPMF is set, and APSUX07 sets this field to a binary value from 1 to 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 “3800 Forms Marking” on page 207.

#### **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 should 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 calling 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 (TERM) function. This is for both normal and abnormal termination.

**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** 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. APSUX07 can set this field to B'1' to bypass the ETFF function. This flag is valid only for continuous-forms printers, and can be accessed only through INIT and BDSC calls to APSUX07.

**XTP7TDS** PSF sets this flag to B'0'. If the exit sets this flag to B'1', PSF terminates the current data set. This flag is valid for the BDSC, RAC, and RLC exit calls to APSUX07.

**XTP7EFFH** PSF sets this flag to B'0' to indicate that an eject to front facing is to be done before the job header page. APSUX07 can set this field to B'1' to bypass the ETFF function. This flag is valid only for

continuous-forms printers; it can be accessed only through INIT and BDSC calls to APSUX07.

- XTP7HDF** PSF sets this flag to B'0' to indicate that the printer should use the hardware default font. If APSUX07 sets this field to B'1', PSF lets the printer substitute the first font in the current font list and replace the hardware default font. This flag is valid only with 3800 printers and during calls to APSUX07.
- XTP7HQUE** Valid only when APSUX07 has 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. APSUX07 can set this field to B'1', to request that the data set be sent to JES and marked as unprintable rather than purged.
- XTP7PRNS** Valid only on the INIT call to APSUX07. 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 232 and "Glossary" on page 363. If APSUX07 sets this field to B'1', PSF does not prune single-byte fonts.
- XTP7PRND** Valid only on the INIT call to APSUX07. 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 232 and "Glossary" on page 363. If APSUX07 sets this field to B'1', PSF is not to prune double-byte fonts.
- XTP7IPMF** Valid only on the INIT and BDSC calls to APSUX07. PSF initially sets this field to B'0' to indicate that no form marks are required for the interrupt message page. If APSUX07 sets this field to B'1', PSF marks the interrupt message page with form marks.
- XTP7IPOS** Valid only on the INIT and BDSC calls to APSUX07. PSF initially sets this field to B'0' to indicate that no offset stacking is required for the interrupt message page. If APSUX07 sets this field to B'1', the printed output is offset-stacked, starting at the interrupt message page.
- XTP7MTOF** Valid only on the INIT call to APSUX07. PSF initially sets this field to B'0' to indicate that fonts are not mapped to outline fonts. If APSUX07 sets this flag to B'1', PSF maps fonts to an equivalent outline font if the printer supports outline fonts. PSF does not map fonts used on auxiliary data sets, fonts found inline or in a user library, or fonts substituted by Exit 7. Double-byte raster coded fonts are prefix-mapped to equivalent outline coded fonts. However, a double-byte raster font section requested by character set and code page name is not mapped to

outline, because there is no outline equivalent for just one section.

**XTP7RFNF** Valid only on the INIT call to APSUX07. 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. If the mapped font does not exist inline or in the host libraries, PSF issues an error message and terminates the page. If APSUX07 sets this flag to B'1', then, when PSF has mapped a raster font to an outline font, it first queries to see if the font exists inline or in the host libraries. If the mapped font does not exist there, PSF uses the original font. Because there are additional library queries for every mapped font resource, PSF performance is degraded.

**XTP7IPSU** Valid only on the INIT call to APSUX07. PSF initially sets this field to B'0' to indicate that interrupt message pages are desired. For more information, see "Considerations on Suppressing an Interrupt Message Page" on page 202. If APSUX07 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.

**XTP738MO** Valid only on the INIT and BDSC 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 APSUX07 sets this flag to B'1', PSF sets the user's data set media origin to the 3800 media origin.

**XTP7MDSD** Valid only on the INIT and BDSC calls. PSF initially sets this field to the PIMSG keyword setting on the OUTPUT statement for the message data set. If APSUX07 sets this flag to B'0', PSF does not send the message data set to a SYSOUT data set. If APSUX07 sets this flag to B'1', PSF sends the message data set to a SYSOUT data set.

**XTP7HCA** Reserved; do not use.

**XTP7C2SI** Valid only on the INIT and BDSC 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 APSUX07 sets this flag to B'1', PSF converts a multiple-celled IM1 image to a single IOCA image.

**XTP7RRLV** PSF provides the current reasonable resource loaded values (RRLVs) in the following fields. PSF uses the RRLVs to determine how many resources should remain loaded at the end of a user's

print data set. The exit can write to these fields to change any RRLVs. For the default values of RRLV, see Table 23 on page 325.

|                 |   |
|-----------------|---|
| <b>XTP7RPD</b>  | RRLV for page definitions.  |
| <b>XTP7RFD</b>  | RRLV for form definitions.  |
| <b>XTP7RMO</b>  | RRLV for medium overlays. PSF always sets this value to '0'. If APSUX07 sets this field to a non-zero value, PSF ignores that setting and sets the value to '0'.  |
| <b>XTP7RPS</b>  | RRLV for page segments.   |
| <b>XTP7RCF</b>  | RRLV for fonts. A double-byte font is considered one font.  |
| <b>XTP7ROC</b>  | RRLV for object containers. PSF always sets this value to '0'. If APSUX07 sets this field to a non-zero value, PSF ignores it and sets the value to '0'.  |
| <b>XTP7RLST</b> | PSF supplies information about the resource list (APSURLST) control block in the following fields:  |
| <b>XTP7LSTP</b> | 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.   |
| <b>XTP7RESC</b> | Contains the number of distinct resources that are being described in the RLST chain.   |
| <b>XTP7LSTC</b> | 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. |
| <b>XTP7OVLY</b> | PSF supplies the name of the overlay being 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')   |
| <b>XTP7PINF</b> | PSF provides the following information on the attached printer:   |
| <b>XTP7PRT</b>  | Printer type and model.   |
| <b>XTP7ATCH</b> | Attachment type—C=Channel, S=SNA, T=TCP/IP.   |
| <b>XTP7FMLY</b> | Device family—3800, 3820, or, for all other IBM printers, AFP1.   |
| <b>XTP7PRTT</b> | Device type—not intended for use in user programming.   |
| <b>XTP7PRTM</b> | Model number for the IBM 3800—for example, 3800 Model 3.<br><b>Note:</b> The IBM 3800 Model 6 is listed as a Model 3.   |
| <b>XTP7PFLG</b> | Printer flags.  |



**XTP7PSEC** PSF sets this flag to B'1', to indicate that the printer is a secure printer—that is, that 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 has terminated the data set (see XTP7TDS on page 303). The optional text can be used to explain why the data set has been terminated. 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' to '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 ten 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 ten concatenation retrieval lists (see “APSLCRL Parameter Area (Concatenation Retrieval List)” on page 308) at every call to the exit. If the storage for a CRL has been freed (see XTP7CRLF, XTP7CRF2 on page 307) or if no library has been found for a resource type, PSF sets the pointer to '0'.

**XTP7MFLG** Flags that the exit can modify to request certain functions.

**XTP7JHMO** Valid only on the INIT and BDSC calls. PSF initially sets this field to B'0' to indicate that the job header media origin is not set to the top left corner. If APSUX07 sets this flag to B'1', PSF sets the job header media origin to the top left corner.

**XTP7JTMO** Valid only on the INIT and BDSC calls. PSF initially sets this field to B'0' to indicate that the job trailer media origin is not set to the top left corner. If APSUX07 sets this flag to B'1', PSF sets the job trailer media origin to the top left corner.

**XTP7DHMO** Valid only on the INIT and BDSC calls. PSF initially sets this field to B'0' to indicate that the data set header media origin is not set to the top left corner. If APSUX07 sets this flag to B'1', PSF sets the data set header media origin to the top left corner.

- XTP7MDMO** Valid only on the INIT and BDSC calls. PSF initially sets this field to B'0' to indicate that the message data set media origin is not set to the top left corner. If APSUX07 sets this flag to B'1', PSF sets the message data set media origin to the top left corner.
- XTP7RSTR** Valid only on the TERM call. PSF initially sets this field to B'1' if the abend is restartable or to '0' if no abend occurred or if the abend is not restartable. If this field is set to B'1' (restartable abend), APSUX07 can set this field to '0' to cause PSF not to restart.
- XTP7WIDE** Valid only on the INIT and BDSC calls. PSF initially sets this field to B'0' to indicate that portrait pages are generated for printing in cut-sheet emulation mode. If APSUX07 sets this flag to B'1', PSF generates landscape pages for printing in cut-sheet emulation mode.
- XTP7IHIB** Valid only on the INIT and BDSC calls. PSF initially sets this field to B'0'. If APSUX07 sets this field to B'1', PSF inhibits recovery.
- XTP7ENDC** Valid only on the INIT call. PSF initially sets this field to B'0'. If APSUX07 sets this field to B'1', PSF ends the SNA LU1 conversation when the NPRO timer expires or after no job to print is available for one minute and the last page printed has been stacked.
- XTP7ABRC** Valid only on the TERM call. This field contains the abend reason code if the FSA is terminating abnormally. It contains X'0000' if the FSA is terminating normally.
- XTP7FMT** Valid only on INIT and BDSC calls. PSF initially sets this field to '0' to indicate that no format resolution has been specified by the exit. For more information, see “Considerations for Setting a Format Resolution Value” on page 207. If APSUX07 sets this field to either 240 or 300, PSF uses that as the specified format resolution value.
- 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)”) at every call to the exit. If the storage for a CRL has been freed (see XTP7CRLF, FTP7CRF2, and XTP7CRF3), or if no library has been found for a resource type, PSF sets the pointer to '0'.

---

## APSLCRL Parameter Area (Concatenation Retrieval List)

The concatenation retrieval list (CRL) is a control block that PSF builds to pass information on 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 XTP7CRLP, XTP7CRLF, and XTP7CRF2 sections of APSGEXTP (see page 307.).

If a CRL does not exist (either because no library was specified or because PSF has 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 68 shows the format of APSLCRL parameter area, and Figure 69 on page 310 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)<br>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)<br>1 = NOT A CDS<br>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 68. APSLCRL Parameter Area Format

| Offsets | Type           | Length | Name     | Description   |
|---------|----------------|--------|----------|---|
| 0       | (0) STRUCTURE  |        | CRLDATA  | DATA SET NAME AND VOLUME ENTRY. NUMBER OF ENTRIES IS EQUAL VALUE IN FIELD CRLRTRVD.   |
| 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.  |
| 2       | (2) SIGNED     | 2      | CRLCATN  | DOMAIN = (0 32767)<br>CONCATENATION NUMBER FOR DATA SET NAME IN CRDLSNM. GENERATED BY APSLJFCB OR APSLOPEN. DOMAIN = (0 255)<br>0 = FIRST OR ONLY DATA SET<br>1 = 2ND DATA SET        |
| 4       | (4) SIGNED     | 2      | CRLNVOL  | NUMBER OF VOLUMES FROM JFCBNVOL. DOMAIN = (1 256)   |
| 6       | (6) CHARACTER  | 44     | CRDLSNM  | JFCB INDEX BY CRLRTRVD. X'40' IN FIRST BYTE OF ENTRY INDICATES DATA SET NAME COULD NOT BE RETRIEVED.  |
| 50      | (32) CHARACTER | 0      | CRLVOLS  | DOMAIN = (ALPHANUMERIC)<br>VOLUME LIST  |
| 50      | (32) CHARACTER | 6      | CRLAVOLS | VOLUME SERIAL NUMBERS. INDEX BY CRLNVOL. X'FF' IN FIRST BYTE OF ENTRY INDICATES VOLUME SERIAL NOT DETERMINE. ONE VOLUME FOR PARTITION DATA SET<br>DOMAIN: 1-6 ALPHANUMERIC CHARACTERS |

Figure 69. Format of the CRL Data Area

## APSLCRL Fields

|                 |   |
|-----------------|---|
| <b>CRLPFIX</b>  | CRL prefix section.   |
| <b>CRLID</b>    | Contains the control block identifier CRL.  |
| <b>CRLLEN</b>   | Contains the length of the CRL, including the data set entries.   |
| <b>CRLPTF</b>   | Contains the date of the PTF.   |
| <b>CRLREL</b>   | Contains the release level.   |
| <b>CRLEID</b>   | Contains an encoded object ID.  |
| <b>CRLSPN</b>   | Contains the subpool number.  |
| <b>CRLBASE</b>  | CRL base section.   |
| <b>CRLDDNM</b>  | Contains the DDNAME for the data set entries.   |
| <b>CRLRTRVD</b> | Contains the number of data set entries; if no data set names are retrieved, it is set to '0'.                |
| <b>CRLCONC</b>  | Contains the number of concatenated data sets; if this is not a concatenated data set, CRLCONC is set to '1'. |

|                 |   |
|-----------------|---|
| <b>CRLLOWCN</b> | Contains the lowest concatenation number for this data set mapping.     |
| <b>CRLHIGCN</b> | Contains the highest concatenation number for this data set mapping.    |
| <b>CRELEND</b>  | End of the CRL base section; can be used as addressability for CRLDATA. |

## CRL Data Area

|                 |  |
|-----------------|--|
| <b>CRLDATA</b>  | This area contains the data set names and volume entries. The number of entries is equal to the value in CRLRTRVD.   |
| <b>CRLSDELN</b> | Contains the length of the data set and volume serial list. This field is used to increment to the next entry mapped by CRLDATA.   |
| <b>CRLCATN</b>  | 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.                   |
| <b>CRLNVOL</b>  | Contains the number of volumes in CRLVOLS; can be used as a maximum index for CRLAVOLS.  |
| <b>CRLDSNM</b>  | Contains the name of the partitioned data set. A X'40' in the first byte indicates that the data set could not be retrieved.   |
| <b>CRLVOLS</b>  | Volume list.   |
| <b>CRLAVOLS</b> | 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. Usually, 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:** APSUX07 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 313.

Figure 70 on page 312 shows the format of APSURLST.

| 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<br>INSERT TEXT                          |
| 24      | (18) CHARACTER | 16     | RLSTRID  | RESOURCE IDENTIFICATION  |
| 24      | (18) CHARACTER | 8      | RLSTNAME | RESOURCE NAME  |
| 24      | (18) BITSTRING | 2      | RLSTGCID | GRAPHIC CHARACTER SET<br>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<br>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      |        | *        | RESERVED   |
|         | 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<br>RESOURCE IN THE RLST CHAIN<br>WILL BE #5 |
| 42      | (2A) UNSIGNED  | 1      | RLSTMSGL | LENGTH OF USER'S MESSAGE<br>INSERT POINTED AT BY RLSTMSGP          |
| 43      | (2B) CHARACTER | 1      | *        | RESERVED   |

Figure 70 (Part 1 of 2). APSURLST Parameter Area Format

|    |                |    |          |                                  |
|----|----------------|----|----------|----------------------------------|
| 44 | (2C) CHARACTER | 4  | RLSTFLGS | FLAGS                            |
| 44 | (2C) BITSTRING | 1  | RLSTMISC | MISCELLANEOUS FLAGS              |
|    | 1... ..        |    | RLSTOFNT | PRINTER SUPPORTS                 |
|    | .1... ..       |    | RLSTGRID | OUTLINE FONTS                    |
|    | ..11 1111      |    | *        | RESOURCE NAME IS A GRID          |
|    |                |    |          | RESERVED                         |
| 45 | (2D) BITSTRING | 1  | RLSTAFLG | FLAGS USED ONLY AT ACCESS TIME   |
|    | 1... ..        |    | RLSTLOAD | FORCE LOAD (RELOAD)              |
|    | .1... ..       |    | RLSTMSG  | ISSUE SUB MESSAGE                |
|    | ..1. ....      |    | RLSTRRI  | RELOAD REQ. IGNORED (SET BY PSF) |
|    | ...1 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) CHARACTER | 24 | *        | RESERVED                         |

Figure 70 (Part 2 of 2). APSURLST Parameter Area Format

## APSURLST Fields

|                 |  |
|-----------------|--|
| <b>RLSTCID</b>  | Contains the control block identifier RLST.  |
| <b>RLSTLENG</b> | Contains the length of the RLST control block.   |
| <b>RLSTPTRS</b> | Pointers.  |
| <b>RLSTPRVS</b> | 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.  |
| <b>RLSTNEXT</b> | 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.   |
| <b>RLSTEXTP</b> | 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.  |
| <b>RLSTMSGP</b> | The pointer to the message insert text. The exit may add up to 64 bytes of text to the end of PSF message APS563I. This field is used with FLSTMSGI and RLSTMSG. For more information on 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. |

|                 |  |
|-----------------|--|
| <b>RLSTRID</b>  | Contains information about the resource represented by the RLST entry. All fields contain information that PSF supplies to the exit.   |
| <b>RLSTNAME</b> | The resource member name. For resources other than fonts, this name contains all eight characters of the member name—for example, P106060. For coded fonts, this eight-byte field contains all eight 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 wishes to make a substitution of the resource. Substitutions must be name for name, GRID for GRID, and so on. |
| <b>RLSTGCID</b> | The two-byte graphic character set global identifier (GCSGID) in the GRID for this font.   |
| <b>RLSTCPID</b> | The two-byte code page global identifier in the GRID for this font.  |
| <b>RLSTFGID</b> | The font global identifier in the GRID for this font.  |
| <b>RLSTFW</b>   | The font width in the GRID for this font.  |
| <b>RLSTPDS</b>  | 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.<br><br>For more information about CRLCATN, see page 311.<br><br>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.  |
| <b>RLSTATTR</b> | Attributes of this resource.   |
| <b>RLSTDFT</b>  | If set, indicates this is a default page definition or form definition.<br><br><b>Note:</b> This flag is set only for the page definition or form definition specified in the PRINTDEV statement.  |
| <b>RLSTINL</b>  | If set, indicates that this is an inline resource.   |
| <b>RLSTSPS</b>  | If set, indicates that this is a soft page segment.  |



|                  |  |
|------------------|--|
| <b>RLSTMEF</b>   | If set, indicates that this is a multiple-entry font.  |
| <b>RLSTSEC</b>   | If set, indicates that the resource is from a security library.  |
| <b>RLSTUSER</b>  | If set, indicates that the resource is from a user library.  |
| <b>RLSTSTAT</b>  | Contains information that affects the deletion status of the resource.   |
| <b>RLSTAUX</b>   | 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.   |
| <b>RLSTDEP</b>   | If set to '1', indicates that this resource has dependencies; that is, it has 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. |
| <b>RLSTNEST</b>  | If set to '1', indicates that this resource has been 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.                       |
| <b>RLSTPLR</b>   | 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.  |
| <b>RLSTRESC</b>  | 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.  |
| <b>RLSTMSGGL</b> | 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.  |

|                  |   |
|------------------|---|
| <b>RLSTFLGS</b>  | Flags.  |
| <b>RLSTMISC</b>  | General flags   |
| <b>RLSTOFONT</b> | 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'.  |
| <b>RLSTGRID</b>  | Set by PSF to B'1' when the RLSTNAME field contains a GRID for a coded font. Otherwise, set to B'0'.  |
| <b>RLSTAFLG</b>  | Flags valid only during the resource-access call (RAC).   |
| <b>RLSTLOAD</b>  | 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.<br><br><b>Note:</b> If the resource concerned is a page definition or a form definition, the load involved is from DASD to PSF virtual storage. For other resources, setting this flag causes the resource to be reloaded into printer storage. |
| <b>RLSTSMMSG</b> | Issues substitution message APS563I. PSF always sets this flag to '0'. If the exit has requested a substitution of this resource and it sets RLSTSMMSG to '1', PSF issues message APS563I.  |
| <b>RLSTRRI</b>   | Set by PSF to B'1' to indicate that PSF is to ignore a load request (RLSTLOAD) for this resource.   |
| <b>RLSTLFLG</b>  | Flags valid only during the resource-load call (RLC).   |
| <b>RLSTLCF</b>   | Informational flag set by PSF to indicate that PSF is loading a coded font.   |
| <b>RLSTLCP</b>   | Informational flag set by PSF to indicate that PSF is loading a code page.  |
| <b>RLSTLFN</b>   | Informational flag set by PSF to indicate that PSF is loading a font character set.   |
| <b>RLSTDFLG</b>  | Flags valid only during the resource-delete call (RDC).   |

- RLSTKEEP** Keep or delete this resource. Before calling the exit for the RDC, PSF sets this flag in accordance with its current pruning algorithm. By modifying this field, the exit tells PSF which resources should remain loaded or should 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 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 has not been 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.

---

## 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 on defining fonts and the Map Coded Font (MCF) structured field, refer to *Advanced Function Printing: Host Font Data Stream Reference*.

Figure 71 on page 318 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 71. 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 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.
- RLSTFNMM** 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 APSUX07

*Table 22 (Page 1 of 3). Input and Output Summary for APSUX07 Processing Indicators. Input is marked "I," and output is marked "O." If the indicator is both, it is marked "IO." For definitions of these fields, see "XTP7 Fields" on page 302 and "APSURLST Fields" on page 313.*

|          | INIT | BDSC | RAC | RLC | RDC | TERM |
|----------|------|------|-----|-----|-----|------|
| XTP7ETYP | I    | I    | I   | I   | I   | I    |
| XTP7RTYP |      |      | I   | I   | I   |      |
| XTP7DSTY |      |      |     |     |     |      |
| XTP7AUX  |      |      | I   | I   |     |      |
| XTP7PDS  |      | I    | I   | I   | I   |      |
| XTP7DSAT |      |      |     |     |     |      |
| XTP7PDFT |      |      | I   | I   |     |      |
| XTP7PJHD |      |      | I   | I   |     |      |
| XTP7PJTR |      |      | I   | I   |     |      |
| XTP7PDSH |      |      | I   | I   |     |      |
| XTP7PMDS |      |      | I   | I   |     |      |
| XTP7PLDS |      | I    |     |     | I   |      |

Table 22 (Page 2 of 3). Input and Output Summary for APSUX07 Processing Indicators. Input is marked "I," and output is marked "O." If the indicator is both, it is marked "IO." For definitions of these fields, see "XTP7 Fields" on page 302 and "APSURLST Fields" on page 313.

|                   | INIT | BDSC | RAC | RLC | RDC | TERM |
|-------------------|------|------|-----|-----|-----|------|
| XTP7ACSE          | IO   | IO   |     |     |     |      |
| XTP7IPCC          | IO   | IO   |     |     |     |      |
| XTP7NOTY          | IO   | IO   | IO  | IO  | IO  |      |
| XTP7RFLG          |      |      |     |     |     |      |
| XTP7ETFF          | IO   | IO   |     |     |     |      |
| XTP7TDS           |      | O    | O   | O   |     |      |
| XTP7EFFH          | IO   | IO   |     |     |     |      |
| XTP7HDF           | IO   | IO   | IO  | IO  | IO  | IO   |
| XTP7HQUE          |      | O    | O   | O   |     |      |
| XTP7PRNS          | IO   |      |     |     |     |      |
| XTP7PRND          | IO   |      |     |     |     |      |
| XTP7IPMF          | IO   | IO   |     |     |     |      |
| XTP7IPOS          | IO   | IO   |     |     |     |      |
| XTP7MTOF          | IO   |      |     |     |     |      |
| XTP7RFNF          | IO   |      |     |     |     |      |
| XTP7IPSU          | IO   |      |     |     |     |      |
| XTP738MO          | IO   | IO   |     |     |     |      |
| XTP7MDSO          | IO   | IO   |     |     |     |      |
| XTP7HCA           | IO   |      |     |     |     |      |
| XTP7C2SI          | IO   | IO   |     |     |     |      |
| XTP7RRLV          | IO   | IO   | IO  | IO  | IO  |      |
| XTP7RLST          |      |      | I   | I   | I   |      |
| XTP7OVLY          |      |      | I   | I   |     |      |
| XTP7PINF          | I    | I    | I   | I   | I   | I    |
| XTP7MSGP and MSGL |      | O    | O   | O   |     |      |
| XTP7CRLF          | IO   |      |     |     |     |      |
| XTP7CRF2          | IO   |      |     |     |     |      |
| XTP7CRLP          | I    |      |     |     |     |      |
| XTP7MFLG          |      |      |     |     |     |      |
| XTP7JHMO          | IO   | IO   |     |     |     |      |
| XTP7JTMO          | IO   | IO   |     |     |     |      |
| XTP7DHMO          | IO   | IO   |     |     |     |      |
| XTP7MDMO          | IO   | IO   |     |     |     |      |
| XTP7RSTR          |      |      |     |     |     | IO   |
| XTP7WIDE          | IO   | IO   |     |     |     |      |
| XTP7IHIB          | IO   | IO   |     |     |     |      |
| XTP7ENDC          | IO   |      |     |     |     |      |
| XTP7ABRC          |      |      |     |     |     | I    |
| XTP7FMT           | IO   | IO   |     |     |     |      |
| XTP7CRF3          | IO   |      |     |     |     |      |
| XTP7CRP2          | I    |      |     |     |     |      |
| RLSTPTRS          |      |      |     |     |     |      |

Table 22 (Page 3 of 3). Input and Output Summary for APSUX07 Processing Indicators. Input is marked "I," and output is marked "O." If the indicator is both, it is marked "IO." For definitions of these fields, see "XTP7 Fields" on page 302 and "APSURLST Fields" on page 313.

|          | INIT | BDSC | RAC | RLC | RDC | TERM |
|----------|------|------|-----|-----|-----|------|
| RLSTPRVS |      |      | I   | I   | I   |      |
| RLSTNEXT |      |      | I   | I   | I   |      |
| RLSTEXTP |      |      | I   | I   | I   |      |
| RLSTMSGP |      |      | O   |     |     |      |
| RLSTRID  |      |      | I   | I   | I   |      |
| RLSTNAME |      |      | 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   |      |
| RLSTMSGI |      |      | O   |     |     |      |
| RLSTFLGS |      |      |     |     |     |      |
| RLSTMISC |      |      | I   | I   | I   |      |
| RLSTAFLG |      |      | IO  |     |     |      |
| RLSTLFLG |      |      |     | I   |     |      |
| RLSTDFLG |      |      |     |     | IO  |      |





---

## Appendix E. 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 72 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=1024K  
//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 72. Sample JCL for Converting the MVS PAGEDEF Resource for Use by VSE

The sample JCL in Figure 72 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 catalogued.

**TEST**

Specifies the VSE sublibrary in which the converted resource is to be catalogued.

**Note:** The converted resource is entered in the VSE LIBRARY.SUBLIBRARY under the same name as the MVS resource.

When a job is being networked from JES to POWER, a special POWER utility name should 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 73. Sample JCL to Route Job to Reader

The sample JCL in Figure 73 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 MVS installation-defined punch class, for example, *b*.

**Note:** With this utility, the converted resource job stream runs without operator intervention.

## Appendix F. Reasonable Resource Loading Values (RRLV)

This appendix lists the default reasonable resource loading values. (See also “Partial Control of Resource Deletion” on page 219.)

**Note:** PSF deletes page definitions and form definitions from virtual storage, not from printer storage.

*Table 23. Reasonable Resource Loading Values (RRLV)*

| Resource          | Device                                      | Default value     |
|-------------------|---|-------------------|
| Page definitions  | 3800  | 6                 |
|                   | 3820  | 6                 |
|                   | All other printers                          | 6                 |
| Form definitions  | 3800  | 6                 |
|                   | 3820  | 6                 |
|                   | All other printers                          | 6                 |
| Overlays          | 3800  | 0 <sup>30</sup>   |
|                   | 3820  | 0 <sup>30</sup>   |
|                   | All other printers                          | 0 <sup>30</sup>   |
| Page segments     | 3800  | 0                 |
|                   | 3820  | 0                 |
|                   | All other printers                          | 0                 |
| Coded fonts       | 3800  | 10                |
|                   | 3800 printer with storage expansion feature | 64                |
|                   | 3820  | 24                |
|                   | All other printers                          | PRT <sup>31</sup> |
| Object containers | 3800  | 0                 |
|                   | 3800 with storage expansion feature         | 0                 |
|                   | 3820  | 0                 |
|                   | All other printers                          | 0                 |

<sup>30</sup> If an overlay is specified in the PRINTDEV, this overlay is kept.

<sup>31</sup> 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 G. 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.

---

### 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 more information about 300-pel printers, refer to *AFP: Printer Information*.

### Using APSRCF30

APSRCF30 is shipped as part of PSF. After PSF has been 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 shipped in both 240-pel and 300-pel resolutions; therefore, you do not have to convert them. Outline fonts, of course, are resolution-independent and do not need to be converted.

Before converting fonts with APSRCF30, you need to:

1. Define a new font library for 300-pel fonts, if one does not already exist. SYS1.FONT300 is the recommended name for your 300-pel system font library.

You should use the existing 240-pel font library as a model, and allocate the new 300-pel font library similarly. The 300-pel font library will require 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* have to be converted with APSRCF30. See "Converting All 240-Pel Fonts in Source Library (APSRAL30)" on page 329. The sample JCL converts all members whose file names start with *C0*:

- Coded font members whose filenames start with *X0* should not be used as input to APSRCF30.
- Code page members whose filenames start with *T1* should not be used as input to APSRCF30.

3. Make any changes in the JCL for APSRCF30 that are needed for your installation. See "APSRCF30 Sample Jobs" on page 329.

The JCL supplied with APSRCF30 should 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 desired

font member names, using NAME on the PROC statement. For example, in the JCL for APSRCF30, specify:

```
C0LOAD10 EXEC PROC=CON,NAME=C0LOAD10
C0LOAD12 EXEC PROC=CON,NAME=C0LOAD12
:
C0S0AP13 EXEC PROC=CON,NAME=C0S0AP13
```

4. Submit the JCL.

After converting fonts with APSRCF30, you need to:

1. Change the PSF startup procedure to use the new 300-pel font library.
2. Arrange for PSF to have access to the code pages and coded fonts that have not been 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
```

3. If you have 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 should contain only the 240 concatenation list, and the FONT02 DD statement should contain 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 listed below, 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 may create an incomplete output file, which you should discard.

| RC         | Meaning  |
|------------|--|
| 4          | One or more of the conversion character data set members required by the conversion program could not 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 could not be found or could not 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 named in the input data set is not valid for any of the following reasons: <ul style="list-style-type: none"> <li>• The member contains structured fields that are not valid..</li> <li>• The font is not bounded-box.</li> <li>• The font is not 240-pel.</li> </ul> |
| 44         | An unexpected end-of-file occurred on the input data set.  |
| 88         | Insufficient virtual storage was available for program execution.  |
| 99         | Not valid font data. This return code is presented if any of the following structured fields in a font has a length less than 1:   |
| <b>FNI</b> | Font index   |

|            |                   |
|------------|-------------------|
| <b>FNM</b> | Font patterns map |
| <b>FNG</b> | Font patterns     |

## APSRCF30 Considerations

Do not convert the Bar Code|Optical Character Recognition font (Proram Number 5688-021). The conversion process usually distorts the font, making it unreadable by an optical scanning device. Check all the JCL for this font before running APSRCF30.

APSRCF30 uses a comparatively large amount of CPU resources, especially if started on a smaller S/390 processor complex. To run APSRCF30, you should have a minimum region size of 5 MB.

Before you invoke APSRCF30, you should allocate the target font library for the conversion.

For an example of JCL that can be changed, see “APSRCF30 Sample Jobs.”

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 shipped 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 supplied are provided as samples to assist you in tailoring JCL to reflect the bounded-box fonts that require converting.

You should allocate the font library used by the PROC before invoking 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).”

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 332 for sample JCL (APSWCF30) that you can modify and use when tailoring the JCL from your source library or from the individual sample jobs in SYS1.SAMPLIB. The JCL you use may 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 *CO*) 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 shipped 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=5120K
//*          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 invoke 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.

This should 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. Make sure that PSF has access to the code page and coded font members from the 240-pel font data set.

You can copy the code page and coded font members from the 240-pel font data set to the 300-pel font data set. Or, you can concatenate the 240-pel font data set after the 300-pel font data set in the PSF startup procedure.

## 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 will be output to your SYSUT3 data set.
- The output on SYSUT3 contains the JCL from SYSUT2, followed by up to 255 EXEC PROC statements representing the character sets in SYSUT1.

If the input PDS directory on SYSUT1 has more than 255 members whose names begin with *C0*, SYSUT3 will contain the SYSUT2 JCL repeated with up to another 255 EXEC PROC statements. This process will continue until all the SYSUT1 PDS members whose names begin with *C0* are processed. MVS has a limitation of 255 EXEC statements in a job.

- You can edit the SYSUT3 output before submitting the job.

- The output of this program always refers to the inline procedure “CON.”

## Example for Selected Font Conversion

Using the following format, and updating the JCL as needed for the font you have 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=5120K
/**          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
```



---

## Appendix H. Installation Verification Program (IVP) Example

The following is a sample of output obtained by running the installation verification program (IVP) on a 240-pel resolution printer.

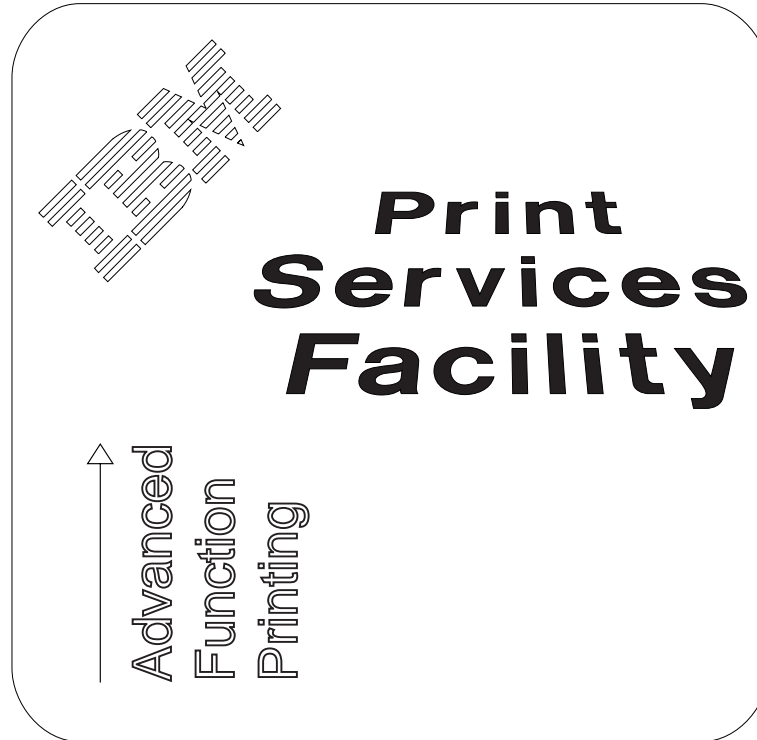


Figure 74. IVP Output Printed with 240-Pel Resolution



---

## Appendix I. PSF Programming Interfaces

The interfaces identified in this appendix let the customer write programs that use the services of PSF. Only the interfaces identified in this appendix should be used to request or receive the services of PSF.

---

### Intended Programming Interfaces

APSGEXTP  
APSLCRL  
APSXUCOM  
APSUBTBL  
APSUECA  
APSURLST  
APSUXP16

---

### Interfaces for Tailoring PSF

APSRFTBL  
APSRGTNM  
APSRNTGM

---

### Installation-Wide Exits

APSUX01  
APSUX02  
APSUX03  
APSUX04  
APSUX05  
APSUX06  
APSUX07  
APSUX14  
APSUX15  
APSUX16





## Appendix J. Mapping Tables for Outline Fonts

This appendix 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 150.

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

Tables 25 and 24 give samples of the information in the tables:

*Table 24. 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 25. Font-Name-to-GRID Table, APSRNTGM*

| FCS NAME | FLAGS | TEKID | FGID <sup>1</sup> | GCSGID  | FW      | VFS     |
|----------|-------|-------|-------------------|---------|---------|---------|
| C0D0GT10 | 0     | X'05' | 0040              | 0000    | 000     | 000     |
| CZN200   | 0     | X'1F' | 2308              | 2039    | 0       | 0       |
| CON200B0 | 0     | X'05' | 2308              | 2039    | 80      | 240     |
| FFFFFFFF | X'FF' | X'FF' | X'FFFF'           | X'FFFF' | X'FFFF' | X'FFFF' |

<sup>1</sup>Specifying zeros (0) for FGID means do *not* map fonts.

Figure 75 on page 340 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, to change entries, or to 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                                       */
*/

```

Figure 75 (Part 1 of 3). Global Resource Identifiers (GRIDs) to Font Names

```

*/ *      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                                        */
*/ *      CHANGE ACTIVITY                                       */
*/ *      $00=LASP0006,HPRF220, 930903, BJ13ATF: RELEASE 2.2.0 @00A*/
*/ *      $01=OW08340, HPRF220, 941020, BJ13DDA: Outline Font D-APAR @01A*/
*/ *      */
*/ *      *****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
*/ *

```

Figure 75 (Part 2 of 3). Global Resource Identifiers (GRIDs) to Font Names

```

*****
**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
APSRGTCPS 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)
**      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 75 (Part 3 of 3). Global Resource Identifiers (GRIDs) to Font Names

Figure 76 on page 343 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 TABLE PROVIDES THE ABILITY FOR THE USER
*/*                TO ADD ENTRIES FOR CODE PAGE AND CHARACTER SET
*/*                MEMBER NAME-TO-GRID MAPPING.
*/*
*/*      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.
*/*                THE FORMAT OF EACH ENTRY IS AS FOLLOWS:
*/*                BYTES 0-7 FONT CHARACTER SET MEMBER NAME
*/*                BYTE 8 FLAGS -RESERVED, MUST BE ZERO.
*/*                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)
*/*                BYTES 14-15 FONT WIDTH
*/*                BYTES 16-17 VERTICAL FONT SIZE
*/*                THE TABLE IS TERMINATED BY A STRING OF
*/*                CL8'FFFFFFFF'.
*/*      LINKAGE: NONE

```

Figure 76 (Part 1 of 3). Font Names to Global Resource Identifiers (GRIDs)

```

**/ * 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*/
**/ * */
**/ * *****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 76 (Part 2 of 3). Font Names to Global Resource Identifiers (GRIDs)

```

*****
**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'FFFFFFF'     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'FFFFFFF'     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 76 (Part 3 of 3). Font Names to Global Resource Identifiers (GRIDs)





---

## Appendix K. Installing the i-data 7913 IPDS Printer LAN Attachment

This appendix 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 a system—either AIX or OS/2—on which the **bootp** daemon program is running.

For a list of the printers that require the i-data 7913 IPDS Printer LAN Attachment, refer to *AFP: Printer Information*.

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 as follows:

- 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 or OS/2.
- 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 additional information, and for instructions for downloading corrective service 7913 microcode, refer to *i-data 7913 IPDS Printer LAN Attachment for Token Ring Installation Guide* or to *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 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 a system, either AIX or OS/2, that is running the TCP/IP **bootp** daemon program. The **bootp** daemon controls the 7913 initialization process.

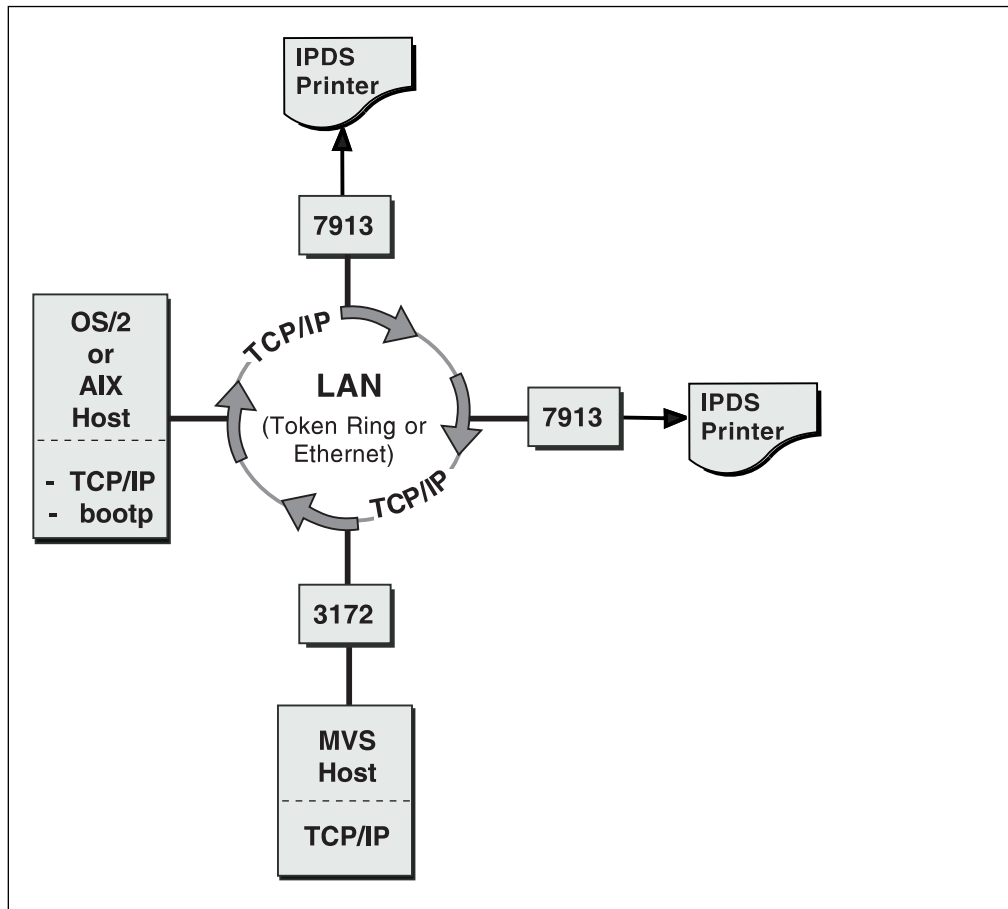


Figure 77. i-data 7913 IPDS Printer LAN Attachment in a TCP/IP LAN Environment

Figure 77 shows a sample TCP/IP LAN configuration, with an AIX or OS/2 host running TCP/IP and the TCP/IP **bootp** daemon. The MVS host and the AIX or OS/2 host can be attached directly to the same LAN as the i-data 7913 IPDS Printer LAN Attachment, as shown in Figure 77, 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 have the following AIX hardware installed:
  - IBM AIX Version 3.2.5 for the RS/6000 (Program Number 5756-030) or higher. If the **bootp** and the i-data 7913 IPDS Printer LAN Attachment are on different token-ring LAN token-ring segments, PTF U422283 is required.
  - The TCP/IP (**bosnet.tcpip.obj**) option of the AIX operating system.
- If you are installing the i-data 7913 IPDS Printer LAN Attachment on an OS/2 system, you must have the following OS/2 software already installed:

- OS/2 Version 2.11 or above
- TCP/IP Version 2 or above
- You must have either an IBM token-ring network or an Ethernet LAN installed and operating correctly.
- Your AIX or OS/2 LAN must coexist with TCP/IP. If not, refer to *TCP/IP Tutorial and Technical Reference*.
- You must have obtained 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 TCP/IP network. This section 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 Figure 78. The data flow is as follows:

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.

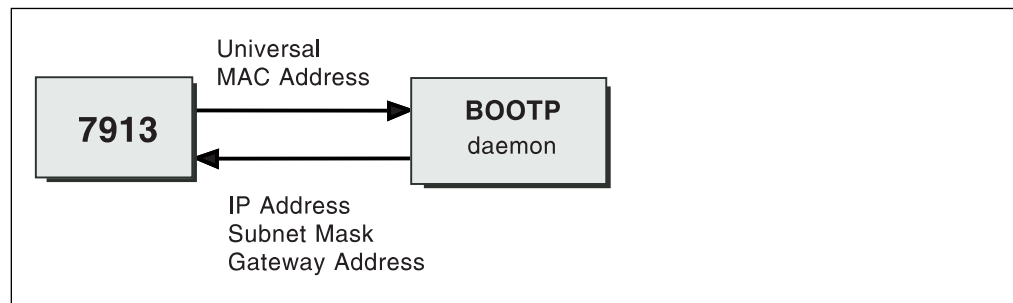


Figure 78. Data Flow between the i-data 7913 IPDS Printer LAN Attachment and the LAN

### 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, refer to the following TCP/IP Requests 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 TCP/IP network administrator.

| 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, AIX, and OS/2. In UNIX or AIX, the **bootptab** file is accessed under:

/etc

In OS/2, the **bootptab** file is in the following subdirectory:

\tcpip\etc

To configure the **bootptab** file for the 7913, you need to add the values from the worksheet (see Table 26).

### 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 section 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 as follows:

#### Basic bootp Configuration

```
P4028: ht=6: ha=00036EFFFFFF; ip=9.99.12.33:\
sm=255.255-255.0: gw=9.99.12.254:

P3812: ht=6: ha=00036E00049F: ip=9.99.12.33:\
sm=255.255.255.0: gw=9.99.12.254
```

The values used in Example 1 are as follows:

#### **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 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 has a unique MAC address 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 page 349.

**Note:** The IP address *must* be unique for each 7913 in your network.

#### **sm=subnet mask**

For a description of the subnet mask, see page 350.

#### **gw=gateway address**

For a description of the gateway address, see page 350.

### 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.

### More Complex bootp Configuration

```
# 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=9.9912.254:
# 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=9.99.12.33:
```

The values used in Example 2 are as follows:

#### **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 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 has a unique MAC address 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 page 349.

**Note:** The IP address must be unique for each 7913 in your network.

---

## Setting Up the bootp Daemon on AIX

After you have configured the **bootp** daemon, you need to set it up in the AIX environment. In order to do this, you must have **root** user authority.

Usually, the 7913 and the IP host 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 3.2.5 (or higher) with PTF U422283, the **bootp** daemon and the 7913 can be on different LAN segments.

The procedure for setting up the **bootp** daemon on AIX is as follows:

1. Add the following 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 is a comment and does not affect the **inetd** super-daemon.

2. Start the **bootp** daemon. There are two methods for doing this:

- You can use the **inetd** super-daemon; this is the usual 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'.more** command to view the contents of the file. If the following line appears and 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 the line above does not appear 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 have updated the **etc/inetd.conf** file, enter the following 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** daemon. If the **bootp** daemon is being managed by the **inetd** super-daemon, do not start the **bootp** daemon manually.

To start the **bootp** daemon manually, enter one of the following at the AIX command line prompt:

```
bootpd -s
```

or

```
bootpd -s -d -d -d -d -d
```

where **-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.

---

## Setting Up the bootp Daemon on OS/2

To use the **bootp** daemon on OS/2, the **SERVICES** file in the **ETC** directory (**tcPIP\etc\services**) must contain the following entries:

```
sbootp    67/udp    #bootp server
cbootp    68/udp    #bootp client
```

To test the **bootpd** command, enter the following on an OS/2 command line:

```
bootpd -d -d -d -d -d
```

where `-d` displays debugging information, such as the hardware address. Each additional `-d` increases the amount of debugging information displayed. You can specify `-d` up to five times.

When you are satisfied that the **bootpd** command is working correctly, you can add the command to your start procedure.

Add the **bootpc** command to the `tcPIP\bin\tcpstart.cmd` file (assuming that TCP/IP was installed in the `tcPIP` subdirectory). If the **bootp** daemon and the 7913 are on the same LAN segment (no bridges), enter the following:

```
start bootpd
```

If the **bootp** daemon and the 7913 are on the different LAN segments (bridges required), enter the following:

```
start bootpd -b
```

---

## 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 installing 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 79 on page 355.



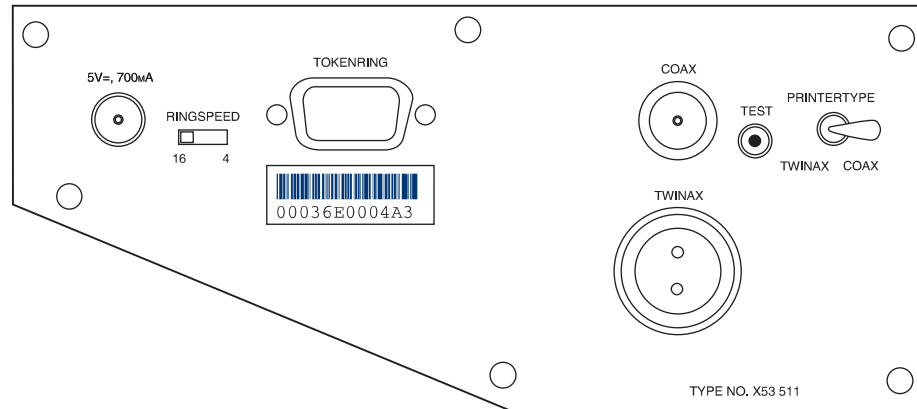


Figure 79. IBM Token-Ring Network Model of the i-data 7913 IPDS Printer LAN Attachment

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 may malfunction.

3. Set the RINGSPEED switch to one of the following:

**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 80.

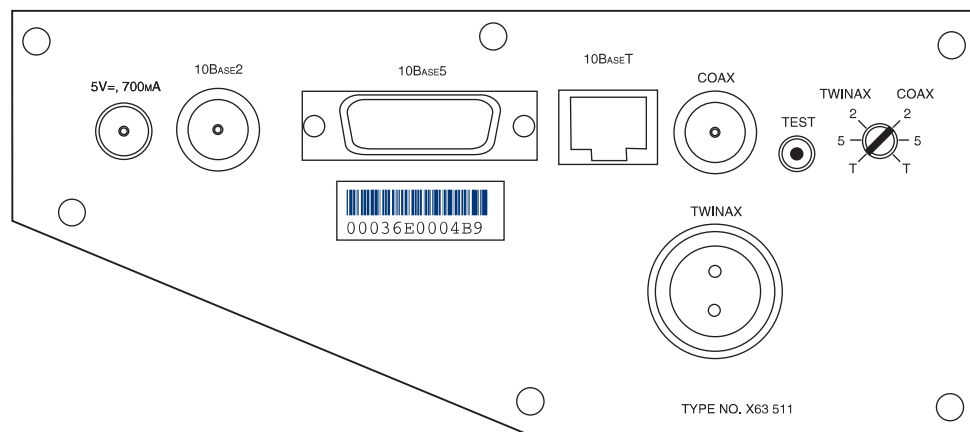


Figure 80. Ethernet LAN Model of the i-data 7913 IPDS Printer LAN Attachment

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 section describes the following 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 shipped 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, PSF for AIX, or PSF for OS/2.
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 cm (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 81 on page 357.

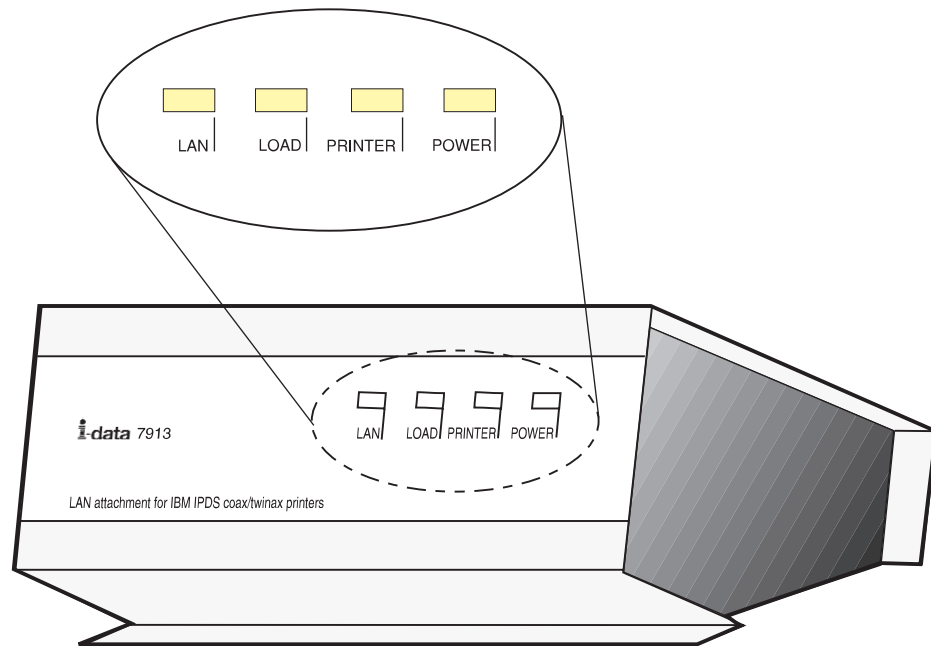


Figure 81. Status indicators on the i-data 7913 IPDS Printer LAN Attachment

**Note:** If the LAN light is not on, see “Troubleshooting the i-data 7913 IPDS Printer LAN Attachment” on page 360 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 349.

## 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 first to stop the PSF FSA for the printer from the MVS console by entering one of the following commands:

– For JES2:

```
$Pprinter-name
```

– For JES3:

```
*VARY printer-name,OFF  
*CANCEL printer-name
```

where *printer-name* is the name of the PSF FSA for the printer.

Then you can 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 PSF may not recognize for a long time that the attachment has been powered off. The length of that time depends on the value specified for the keep-alive transmissions, described on page 245.

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

```
MODIFY FSSname,FORCE,printer-name
```

**Note:** Before powering the i-data 7913 IPDS Printer LAN Attachment back on, be sure to power the printer off and then 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 356.

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 out on the printer. Figure 82 on page 359 shows a sample test page for the IBM token-ring network model; Figure 83 on page 359 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 360 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: 9.99.12.33
Netmask: FFFFFFF0
Default Gateway: 9.99.12.254
Broadcast Address: 9.99.12.255
Port: 5001
Response Timeout: 20 seconds
IP Defaults: None
Remote IP: Port:
Remote IP: Port:

```

Figure 82. 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: 9.99.12.33
Netmask: FFFFFFF0
Default Gateway: 9.99.10.22
Broadcast Address: 9.99.10.255
Port: 5001
IP Defaults: 9.99.10.132,255.255.255.0, 9.99.10.220
Remote IP: Port:
Remote IP: Port:

```

Figure 83. 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 9.99.12.33
```

Or, from the SDSF command area, enter:

```
ping 9.99.12.33
```

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.”

A successful ping usually indicates that the MVS 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 five minutes for TCP/IP to clear the Address Resolution Protocol (ARP) tables.<sup>32</sup>
3. Enter the **ping** command again from the MVS system to determine whether you receive a response.

If you receive a successful response to the **ping** command, indicating that there are duplicate IP addresses, consult your TCP/IP network administrator to resolve the IP address problem.

---

## Troubleshooting the i-data 7913 IPDS Printer LAN Attachment

This section contains symptoms and solutions for common problems you may encounter when installing or operating an i-data 7913 IPDS Printer LAN Attachment. For additional problem-solving information you may need during installation, refer to 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 27 on page 361 describes problems that might occur during the installation of the i-data 7913 IPDS Printer LAN Attachment.

Table 28 on page 362 describes problems that might occur during the operation of the i-data 7913 IPDS Printer LAN Attachment.

---

<sup>32</sup> You may 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, refer to *OS/390 eNetwork Communications Server: IP Configuration Guide*.

| <i>Table 27. Problems in Installing the 7913 Attachment</i>   |   |   |
|---|---|---|
| <b>Symptoms</b>   | <b>Possible problems</b>  | <b>Solutions</b>  |
| LAN light goes on briefly, then goes off. You have attached the 7913 to a LAN and turned it on.   | Token-ring speed set incorrectly.                                   | Set the ring speed to:<br><br><b>4</b> for 4-Mbps LANs<br><b>16</b> for 16-Mbps LANs<br><br><b>Attention:</b> 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 have attached the 7913 to a LAN and turned it on.  | The <b>bootp</b> daemon is not started.                             | Start the <b>bootp</b> daemon on an OS/2 or AIX system.   |
|   | The 7913 and the <b>bootp</b> daemon are separated by a LAN bridge. | <b>OS/2:</b> Specify the <b>-b</b> option when starting the <b>bootp</b> daemon.<br><br><b>AIX:</b> Make sure that you have PTF U422283 installed on AIX 3.2.5.   |
|   | The 7913 and the <b>bootp</b> daemon are separated by an IP router. | Configure the IP router to pass certain broadcast frames.   |
|   | The hardware type in the <b>bootptab</b> entry is set incorrectly.  | Set the hardware type to:<br><br><b>ht=6:</b> or <b>ht=tr:</b> for Token-Ring<br><b>ht=ethernet:</b> 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.   |
| No test page prints. You have 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:<br><br><b>Coax 2:</b> 10base2 to coax<br><b>Coax 5:</b> 10base5 to coax<br><b>Coax T:</b> 10baseT to coax<br><br><b>Twinax 2:</b> 10base2 to twinax<br><b>Twinax 5:</b> 10base5 to twinax<br><b>Twinax T:</b> 10baseT to twinax<br><br>For the token-ring LAN model, set the PRINTERTYPE switch correctly, either to COAX or TWINAX.<br><br><b>Attention:</b> Turn off the 7913 before changing the switch. |
| No response on ping to the 7913.  | The IP address of the MVS 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 28. Problems in Operating the 7913 Attachment*

| <b>Symptoms</b>   | <b>Possible problems</b>                                | <b>Solutions</b>   |
|---|---|--|
| 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 the printer on and restart the PSF FSA for the printer.                   |
|   | Duplicate address for the 7913.                         | Follow the steps on page 360 to verify that the address of the 7913 is unique. |
|   | Duplicate address for the MVS system.                   | Consult your network administrator.  |
|   | The MTU size for the MVS system is too large.           | For the recommended values of MTU, see 246.                                    |



---

# Glossary

---

## Sources

This publication includes terms and definitions from the *IBM Dictionary of Computing*, ZC20-1699.

Definitions reprinted from the *American National Dictionary for Information Processing Systems* are identified by the symbol (A) following the definition.

Definitions reprinted from a published section of the International Organization for Standardization (ISO) *Vocabulary—Information Processing* or from a published section of *Vocabulary—Office Machines* developed by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Committee (ISO/IEC JTC1/SC1) are identified by the symbol (I) following the definition. Because many ISO definitions are also reproduced in the *American National Dictionary for Information Processing Systems*, ISO definitions may also be identified by the symbol (A).

Definitions reprinted from working documents, draft proposals, or draft international standards of ISO Technical Committee 97, Subcommittee 1 (Vocabulary), Joint Technical Committee 1 are identified by the symbol (T) following the definition, indicating that final agreement has not yet been reached among its participating members.

Definitions that are specific to IBM products are so labeled—for example, “In SNA,” or “In the 3820 printer.”

---

## References

The following cross-references are used in this glossary:

**Contrast with.** This refers to a term that has an opposite or substantively different meaning.

**See.** This refers to multiple-word terms in which this term appears.

**See also.** This refers to related terms that have similar, but not synonymous, meanings.

**Synonym for.** This appears in the commentary of a less desirable or less specific term and identifies the preferred term that has the same meaning.

**Synonymous with.** This appears in the commentary of a preferred term and identifies less desirable or less specific terms that have the same meaning.

## A

**abend.** An abnormal end of task before completion of the task because of an error condition that cannot be recovered while the task is running.

**absolute coordinate.** One of the coordinates that identify the location of an addressable point with respect to the origin of a specified coordinate system. Contrast with *relative coordinate*.

**absolute positioning.** The establishment of a position within a coordinate system as an offset from the origin of the coordinate system. Contrast with *relative positioning*.

**access control.** In computer security, the methods and facilities used to ensure that a computer system and the data, system software, and application programs stored in it can be accessed only by authorized users in authorized ways.

**access method.** A method for moving data between main storage and I/O devices.

**accumulator.** (1) A 3800 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. (2) A register in which the result of an operation is formed.

**ACIF.** (1) AFP conversion and indexing facility. (2) A PSF utility program that converts a print file into MO:DCA (AFP), creates an index file for input data, and collects resources used by an AFP document into a separate file.

**ACK.** Acknowledgment. See also *Positive acknowledge reply*.

**acknowledge reply.** A printer-to-host reply that returns printer information (a positive reply) or reports exceptions (a negative reply).

**acknowledgment request.** A request from the host for information from the printer. An example of an acknowledgment request is the use of the ARQ flag by a host system to request an acknowledge reply from an attached printer.

**action code.** A software-generated or hardware-generated code that indicates a recovery

action. 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 MO:DCA data, page definition, or overlay.

**addressable point.** For page printers, any point in a presentation surface that can be addressed. Synonymous with *picture element*.

**Advanced function common control unit (AFCCU).** Converts the IPDS into a presentation format that is usable by the COM; 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 on presentation devices. AFP includes creating, formatting, archiving, retrieving, viewing, distributing, and printing information. See *presentation device*.

**AEA.** Alternative exception action.

**AFCCU.** Advanced function common control unit.

**AFP.** Advanced Function Presentation.

**AFP data stream.** A presentation data stream that is processed in AFP environments. MO:DCA-P is the strategic AFP data stream. IPDS is the strategic AFP printer data stream.

**AFPDS.** A term formerly used to identify the MO:DCA-P-based data stream interchanged in AFP environments. See also *MO:DCA-P* and *AFP data stream*.

**all-points-addressable (APA).** The ability to address, reference, and position text, overlays, and images at any defined position or pel on the printable area of the paper. This capability depends on the ability of the hardware to address and to display each picture element.

**AIX.** Advanced Interactive Executive

**AIX operating system.** IBM's implementation of the UNIX operating system. RS/6000 runs the AIX operating system.

**AOS.** Alternative offset stacker.

**APA.** All-points-addressable.

**APAR.** Authorized program analysis report.

**API.** Application program interface

**application program.** A program written for or by a user that applies to the user's work, such as a program that does inventory control or payroll processing.

**AR.** Activate resource (IPDS command).

**ARP.** Address resolution protocol.

**ARQ.** (1) Acknowledgment required. (2) A flag in the header of an IPDS command, requesting the subsystem to inform PSF when that command and all previous commands have been processed (a way of synchronizing PSF with subsystems). The NOP command following an End Object Container and Print Buffered Data command sequence is sent with an ARQ so that the COM can check the setup data and determine whether a resolution change is necessary.

**asynchronous.** Without regular time relationship; unexpected or unpredictable with respect to running program instructions.

**authorized program analysis report (APAR).** A report of a problem caused by a suspected defect in a current unaltered release of a program.

**auxiliary data set.** A data set that contains job header, data set header, job trailer, or message data. Contrast with *print data set*.

**auxiliary resource.** Fonts, page segments, overlays, page definitions, or form definitions associated with auxiliary data sets.

## B

**baseline.** In a font, the imaginary line on which successive characters are aligned in the inline direction.

**BCOCA.** Bar Code Object Content Architecture.

**bin.** A paper supply on cut-sheet printers. See also *cassette*.

**BIND.** The command sent by VTAM to a logical unit to establish a VTAM program-to-logical unit session.

**bitmap.** A coded representation in which each bit, or group of bits, represents or corresponds to an item: umdashfor example, a configuration of bits in main storage in which each group of bits corresponds to one pel of a display image.

**BOC.** Begin Object Container. Marks the beginning of the object.

**boiler plate.** Presentation data that is replicated on a page or within a group of pages, serving as instructions for (or auxiliary information about) the page or group of

pages—for example, instructions on how to fill out a form.

**bounded-box font.** A font designed to use bounded character boxes. Contrast with *unbounded-box font*.

**bounded-box relative metrics.** Units of measure that are expressed in 1000 units per Em-square.

**bounded character box.** A character-box design that does not require character positioning information to be included (as untuned pels) in the box. Contrast with *unbounded character box*.

**BTS.** Burster-trimmer-stacker.

**buffer pool.** An area of storage in which all the buffers of a program are kept.

**buffered pages.** Pages kept in printer control storage, waiting to be printed.

**burst.** To separate continuous-forms paper into single sheets.

**burster-trimmer-stacker (BTS).** A printer hardware feature that separates continuous forms into single sheets, trims the carrier strip from both edges of the forms, and stacks the sheets.

**byte multiplexer channel.** A multiplexer channel that interleaves bytes of data.

## C

**cache.** To store in a special-purpose buffer storage, which is smaller and faster than main storage. PSF holds a copy of data that was obtained from main storage and that will be needed next for processing.

**capture.** Some printers can capture downloaded fonts. These saved fonts automatically become new temporary printer-resident fonts. This improves performance on future jobs that use the same fonts. Both raster fonts and outline fonts can be captured.

**carriage control character.** An optional character in an input data record that specifies a write, space, or skip operation.

**cassette.** In a cut-sheet printer, a movable paper storage enclosure. See also *bin*.

**CCGSID.** Coded character set identifier.

**CCM.** Common Conversion Machine.

**CCW.** Channel command word.

**CDS.** Control data set.

**CF.** Coded font.

**channel-attached.** In PSF, a device that is linked to the host system exclusively by System/390 channel protocols. For example, a 3800 printer cabled to the host system with an IBM System/390 channel adapter is considered a channel-attached printer. Contrast with *SNA-attached*.

**channel code.** A number from 1 to 12 that identifies a position in the forms control buffer or a page definition. A carriage control character can select a position defined by a particular channel code.

**channel command word (CCW).** A command that directs data channel operations. CCWs built by Print Services Facility are sent to channel-attached printers.

**channel counter.** A counting device that identifies how many pages have been successfully received. Synonymous with *received page counter*.

**channel station.** In the printer, a page-counting device that counts the number of pages queued for printing.

**CHAR.** A data type for architecture syntax, which indicates one or more bytes to be interpreted as character information.

**character.** (1) A member of a set of elements that is used for the representation, organization, or control of data. Characters may be letters, digits, punctuation marks, or other symbols represented in the form of a spatial arrangement of adjacent or connected strokes or in the form of other physical conditions in data media. (2) A letter, numeral, punctuation mark, or special graphic used for the production of text. (3) A byte of data. (4) See also *graphic character*.

**character box.** The area that completely contains the graphic character raster pattern.

**character data.** Data in the form of letters and special characters, such as punctuation marks. See also *numeric data*.

**character graphic.** A visual representation of a character, other than a control character, that is normally produced by writing, printing, or displaying. (T)

**character group.** Any number of character graphics and character properties.

**character identifier.** The standard identifier for a character regardless of its style. For example, all uppercase As have the same character identifier. Synonymous with *graphic character identifier*.

**character increment.** The distance between the current print position and the next print position.

**character rotation.** The alignment of a character with respect to the baseline, measured in degrees in a clockwise direction. Examples are 0°, 90°, 180°, and 270°. This term usually applies to individual characters. Synonym for *rotation*. See also *orientation*.

**character set.** (1) A finite set of characters upon which agreement has been reached and which is considered complete for some purpose—for example, each of the characters in ISO Recommendation R646 “6- and 7-bit coded character sets for information processing interchange.” (2) For page printers, the font library member that contains the character graphics and their descriptions.

**checkpoint.** A point at which information about the status of a job and the system can be recorded so that the job step can be restarted later. For any print job, the user can specify that PSF/MVS take checkpoints based on either the number of logical pages processed or the number of processing seconds elapsed.

**code page.** (1) A set of assignments, each of which assigns a code point to a character. Each code page has a unique name or identifier. Within a given code page, a code point is assigned to one character. More than one character set can be assigned code points from the same code page. See also *coded font*. (2) A font component that associates code points and character identifiers. A code page also identifies how undefined code points are handled.

**Code Page Global Identifier (CPGID).** A unique code page identifier that can be expressed either as a 2-byte binary value or as a 5-digit decimal value.

**code point.** A 1-byte code representing one of 256 potential characters.

**coded font.** A font library member that associates a code page and a font character set. For double-byte fonts, a coded font associates more than one pair of code pages and font character sets.

**coded font local identifier.** A 1-byte identifier that the Map Coded Font structured field assigns to each coded font it selects. The identifier is then specified in the text-control sequence that precedes the string of text to be printed with the particular font.

**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.

**coded overlay.** An overlay that is stored in the printer in a coded (not raster) format. Contrast with *raster pattern overlay*.

**color mapping table.** An architected 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.

**color selection.** The ability to specify a color other than black to print data in more than one color. Some printers support selection of several colors, depending upon the color of the ribbon installed in the printer. Other printers support the selection of black or “color of media” that can cause white lettering on a background that has been shaded black, for example.

**COM.** Computer output microfilm. COM is the hardware controller that manages the microfilm print engine and processes the functions unique to COM.

**COM device.** See *microfilm device*.

**COM setup data.** Data that enables the PSF user 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.

**COM Setup Utility.** An MVS-based utility that creates an AFP object containing the COM setup data for a microfilm device.

**command.** A request from a terminal or a specification in a batch-processing print file for the performance of an operation or the execution of a particular program.

**committed copy counter.** A counting device that identifies the number of copies of a page that have been committed to printing but have not been counted by the committed page counter.

**committed page counter.** A counting device that identifies the number of pages that have been committed to printing and have been removed from the page buffer. Synonym for *transfer counter*.

**common-use forms.** A set of paper sizes selected as being the most common in use throughout the world.

**communication.** See *data communication*.

**communication-attached.** In PSF, pertaining to a device that is SNA-attached and that uses a communication controller. For example, a 3812 printer attached to a 3174 control unit that is attached to a 37X5 Communication Controller is considered a communication-attached printer. Contrast with *local-attached*.

**compatibility font.** An AFP font designed to emulate the uniformly spaced and fixed-pitch fonts used with line printers. Synonym for *IBM Compatibility font*.

**compatibility mode.** A mode of operation for printing the output of 3800 Model 1 application programs on a

3800 page printer with little or no change to the application or the JCL.

**completion code.** An indication that reflects the status of a task set at the time of its termination.

**composed-text data.** An obsolete term for text data and text control information that dictates the format, placement, and appearance of data to be printed. Currently called *MO:DCA-P data*.

**compressed pattern storage (CPS).** Storage that holds the extended (double-byte) fonts for the 3800 Model 6 or Model 8 printers.

**Computer output microfilm.** See *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 problems in data processing. See also *programming RPQ* and *RPQ*.

**concatenate.** (1) To link together. (2) To join two character strings.

**concatenated data set.** In MVS, 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 enables input data records to partially control their own formatting.

**condition code.** A code that reflects the result of an input/output, arithmetic, or logical operation.

**console.** A part of a computer used for communication between the operator or maintenance engineer and the computer. (A)

**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 separate them. Before printing, the forms are folded in a stack arrangement, with the folds along the perforations. Contrast with *cut-sheet paper*.

**continuous-forms stacker.** An output assembly on a continuous-forms printer that refolds and stacks the continuous forms after printing.

**control character.** A character that starts, changes, or stops any operation that affects the recording, processing, transmitting, or interpreting data (such as carriage return, font change, and end of transmission).

**control record.** A subset of structured fields that can be intermixed with line data records in a print data set.

**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. Synonymous with *medium map*. See also *internal copy group*.

**copy modification.** The process of adding, deleting, or replacing data on selected copies of certain pages of a print job.

**copy separation.** The method for distinguishing between consecutive copies of a single data set. In an continuous-forms stacker, the method consists of the alternation between one, two, or three vertical bars placed on the left carrier strip at the end of a copy. In the burster-trimmer-stacker, or on cut-sheet paper, copies are separated by offset stacking.

**core interchange font.** A uniformly spaced typographic font with specialized characters for different languages.

**CPGID.** See *Code Page Global Identifier*.

**CPS.** Compressed pattern storage.

**current print position.** The picture element that defines the character reference point or the upper-left corner of an image.

**cut-sheet paper.** Paper that is cut into separate, uniform-size sheets before being loaded into the printer. Contrast with *continuous forms*.

## D

**daemon.** In the AIX and OS/2 operating systems, a process begun by the root user or by the root shell that can be stopped only by the root user. Daemon processes generally provide services that must be available at all times, such as sending data to the printer. A daemon runs continuously, looking for work to do, performing that work, and waiting for more work. A daemon does not have a controlling terminal associated with it.

**DASD.** Direct access storage device.

**data check.** A synchronous indication of a condition caused by invalid data or incorrect positioning of data. The reporting of some data checks can be suppressed.

**data communication.** Transfer of information between functional units by means of data transmission according to a protocol. (I)

**data control block (DCB).** A control block used by access method routines in storing and retrieving data.

**data link.** Connection between two terminals operating according to a link protocol. Contrast with *telecommunication line*.

**data map.** An internal object in a page definition that specifies fonts, page segments, fixed text, page size, and the placement and orientation of text. Synonym for *page format*.

**data map transmission subcase.** An internal object that specifies the control information for printing line data. One data-map transmission subcase must appear in each data map of a page definition.

**data set.** (1) Synonym for *file*. (2) A named set of records stored and processed as a unit. See also *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.** Data control block.

**DCF.** Document Composition Facility.

**default.** Pertaining to an attribute, value, or option that is assumed when none is explicitly specified. (I)

**deferred-printing mode.** A printing mode that spools output through JES to a data set instead of printing it immediately. Output is controlled using JCL statements. Contrast with *direct-printing mode*.

**define group boundaries.** An IPDS command used to assign a common set of processing functions (such as "Keep as one entity") and/or characteristics (such as job source identification) to a group of pages, one document, or several documents.

**device manager.** (1) For channel-attached printers, the subcomponent of PSF that manages the interface to the printer, builds CCWs, issues I/O requests, and processes I/O interrupts. (2) For SNA-attached printers, the device manager is the PSF subcomponent that loads the PPCC during device initialization, processes completed I/O, establishes and controls an SNA session through the PPCC, and supplies printer information in the CCB. (3) For TCP/IP-attached printers, the PSF subcomponent that processes completed I/O, establishes and controls a network session through the TCP/IP subcomponent, and supplies printer information in the CCB.

**DGB.** See *define group boundaries*.

**direct access storage device (DASD).** A computer storage device in which access time is, in effect, independent of the location of the data.

**Direct Printer Services Subsystem (DPSS).** The PSF subcomponent that acts as the interface between PSF and an application program when the job entry subsystem 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 allows PSF exclusive use of a channel-attached printer. Output is printed immediately and is not spooled through JES. Contrast with *deferred-printing mode*.

**disabled mechanism.** A function of a printer that is temporarily out of operation. In such a case, PSF may use alternative options.

**Distributed Print Function (DPF).** A component of PSF/2 that you can install and use to print jobs from MVS, VSE, VM, or OS/400 systems. DPF receives PSF output and resources for spooling and printing with PSF/2. DPF allows PSF/MVS, PSF/VSE, PSF/VM, and OS/400 to send print files to a PS/2. DPF also stores PSF/MVS and PSF/VSE resources in the DPF resource library, so that the host system does not have to send PSF resources each time documents are spooled.

**document.** (1) One or more pages collected into a single job. (2) Data that has already been composed into pages and that contains a Begin Document and an End Document structured field. (3) In word processing, a collection of one or more lines of text that can be named and stored as a separate entity. (4) A publication or other written material pertaining to a specific subject or related subjects.

**Document Composition Facility (DCF).** An IBM licensed program that provides a text formatter called SCRIPT/VS. SCRIPT/VS can process files marked up with a unique set of controls and tags.

**document environment group.** An internal object that identifies overlay and suppression usage and defines the placement of the page or pages on the form. A document environment group is a required part of every form definition.

**dotted decimal notation.** The syntactical representation for a 32-bit integer that consists of four 8-bit numbers written in base 10 with periods (dots) separating them. It is used to represent IP addresses.

**double-byte coded font.** A font in which the characters are defined by 2 bytes, the first defining a coded font section, and the second defining a code point. Double-byte coded fonts are required to support languages requiring more than 256 graphic characters. Two bytes are required to identify each graphic character. A double-byte font is used to print Kanji. Contrast with *single-byte coded font*.

**double-dot image.** In Advanced Function Presentation, an image that is enlarged by doubling the pel pattern horizontally and vertically.

**download.** To transfer data from a processing unit to an attached device such as a microcomputer for processing.

**DPF.** Distributed Print Function.

**DPSS.** Direct Printer Services Subsystem.

**drain.** An operator action to halt the flow of jobs to a printer, usually to stop the printer or to change print options.

**drained state.** The condition in which job flow to the printer is stopped until the operator enters a command, such as START, that instructs PSF to begin sending jobs to the printer.

**DRM.** Dynamic reconfiguration management.

**DSC.** Data synchronous command

**duplex printing.** Printing on both sides of a sheet of paper. Contrast with *simplex printing*. See also *normal duplex printing* and *tumble duplex printing*.

**dynamic reconfiguration management (DRM).** A capability that enables a system programmer to add, delete, or change AFP1 or 3820 channel-attached printers from an I/O configuration. The system must be MVS/ESA SP 4.2.0 and above, and must be running JES2 4.2.0 and above.

## E

**EBCDIC.** Extended binary-coded decimal interchange code.

**ECB.** Event control block.

**EID.** Event ID.

**electronic forms.** A collection of constant data that is electronically composed in the host processor and can be merged with variable data on a page during printing.

**electronic overlay.** A collection of constant data, such as lines, shading, text, boxes, or logos, that is electronically composed in the host processor and stored in a library, and that can be merged with variable data during printing. Contrast with *page segment*. See also *page overlay* and *medium overlay*.

**embedded text control.** One or more bytes of control information within a set of text. The text control itself is not printed, but it controls certain operations on following text.

**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 normal work. Contrast with *disabled*.

**end-user interface.** Any product method by which a customer can obtain the services of a product—for example, coding samples, commands and command lists, control statements, display panels, job control language, listings of data streams, messages, program link-editing control statements, and text-formatting commands. Not all products have end-user interfaces; some products provide their services through programming interfaces, and others provide their services only to other products. Contrast with *programming interface for customers*.

**EOC.** End object container, which marks the end of the object.

**equipment check.** An asynchronous indication of a printer malfunction.

**ERP.** Error recovery procedures.

**error recovery procedures (ERP).** Procedures designed to help isolate and, where possible, to recover from errors in equipment. The procedures are often used in conjunction with programs that record information on machine malfunctions.

**ES for OS/2.** Extended Services for OS/2 2.0 or later.

**ESA.** Enterprise System Architecture.

**escape character.** In a text-control sequence, the control character X'2BD3' that indicates the beginning of the sequence and the end of any preceding text.

**ESCON.** Enterprise system connection.

**ESTAE.** Extended specify task abnormal exit.

**ETFF.** Eject to front facing.

**Ethernet.** A 10-Mb baseband local area network that allows multiple stations to access the transmission medium at will without prior coordination, avoids contention by using carrier sense and deference, and resolves contention by using collision detection and transmission. Ethernet uses carrier sense multiple access with collision detection.

**EUI.** End user interface.

**event control block (ECB).** A control block used to represent the status of an event.

**EWS.** Early warning system.

**execution.** The process by which a computer carries out one or more of the instructions in a computer program. (I) (A)

**exception.** A condition in which the printer:

- Detects an invalid or unsupported command, order, control, or parameter value from the host.
- Finds a condition about which the host system must be notified.
- Detects a condition that makes it necessary for the host system to resend data.

**exception highlighting.** The markings placed on the printed page to indicate the location of a data-stream error.

**exit.** See *installation exit*.

**Extended Architecture (XA).** An MVS operating system.

**extended binary-coded decimal interchange code (EBCDIC).** A coded character set of 256 eight-bit characters.

**extended specify task abnormal exit (ESTAE).** A macroinstruction that a user can issue to intercept an abend.

## F

**FCB.** See *Forms Control Buffer*.

**FDEF.** Form definition.

**FGID.** See *Font Typeface Global Identifier*.

**fiche.** See *microfiche*.

**file.** A member of a partitioned data set or a sequential data set.

**fixed metrics.** Measurement information in specific units such as pels, inches, or centimeters for individual or collections of graphic characters. Contrast with *relative metrics*. See also *font metrics*.

**flash.** See *forms flash*.

**floating overlay.** Synonym for *page overlay*.

**FLU.** See *Font Library Update utility*.

**FOCA.** 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) (2) One size and one typeface in a particular type family, including letters, numerals, punctuation marks, special characters, and ligatures.

(3) A paired character set and code page that can be used together for printing a string of text characters. A double-byte font can consist of multiple pairs of character sets and code pages. See *coded font* and *double-byte coded font*.

**font character set.** Synonym for *character set*.

**Font Library Update utility (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.** PSF has internal tables that equate core raster fonts to core outline fonts. This comparing and matching activity is called mapping. The default is for PSF/MVS not to map fonts, but you can set XTP7MTOF to ON (B'1') in Exit 7 (APSUX07) to enable PSF to map raster fonts to 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 may 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 *character metrics*.

**font section.** Synonym for *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.

**form.** A division of the physical medium. Multiple forms 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 one form. Envelopes are an example of a physical medium that comprises only one form. The IPDS architecture defines four types of form: cut-sheets, continuous forms, envelopes, and computer output on microfilm. Each type of form has a top edge. A form has two sides—a front side and a back side. Synonymous with *sheet*. See also *medium*, *sheet*, and *page*.

**format.** (1) A specified arrangement of such things as characters, fields, and lines; usually used for displays, printouts, or files. (2) To arrange such things as characters, fields, and lines. (3) (v.) To prepare a document for printing in a specified format.

**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. Resolution can be specified in these ways:

- By APSUX07 at BDSC time



- BY APSUX07 at INIT time
- By application programs in the MCF2 structured field
- On the OUTPUT statement with keyword RESFMT

**formatted print records.** 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.

**FORMDEF.** A JCL parameter that specifies a form definition. See *form definition*.

**form definition.** A resource used by PSF that defines the characteristics of the form, which includes such functions as overlays to be used (if any), paper source (for cut-sheet printers), duplex printing, text suppression, the position of MO:DCA-P 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 forms control buffer is a line-printer control that is similar to the punched-paper, carriage-control tape used on IBM 1403 printers. On AFP page printers, the forms control buffer is replaced by the page definition. See *page definition*.

**forms flash.** In the 3800 printer, the function of the printer that enables it to print user-prepared photographic images with variable MO:DCA-P data. When forms flash is used, the printer operator must insert a frame containing a photographic negative into the printer.

**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.** Functional subsystem application.

**FSI.** Functional subsystem interface.

**FSS.** Functional subsystem.

**functional subsystem (FSS).** The PSF address space created by JES.

**functional subsystem application (FSA).** An area within the PSF functional subsystem that drives and manages a single printer.

**functional subsystem interface (FSI).** A set of services that provides for communication between the JES address space or direct printer services subsystem and the PSF functional subsystem.

**fuser counter.** A counting device in a printer that identifies how many pages have successfully passed the jam-recovery point defined by the printer. See *jam-recovery page counter*.

**fuser station.** The printer assembly that bonds the toned image to the paper by heat and pressure.

**FW.** Font width.

## G

**gateway.** A functional unit that interconnects two computer networks with different network architectures. A gateway connects networks or systems of different architectures. A bridge interconnects networks or systems with the same or similar architectures. (T)

**GCGID.** See *Graphic Character Global Identifier*.

**GCSGID.** See *Graphic Character Set Global Identifier*.

**GDDM.** Graphical Data Display Manager.

**generalized trace facility (GTF).** An optional OS/VS service program that records significant system events, such as supervisor calls and start I/O operations, for use in problem determination.

**global resource identifier (GRID).** An 8-byte identifier that identifies a coded font resource. A GRID contains the following fields, in the order shown:

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, in 1440ths of an inch.

**GOCA.** Graphics object content architecture.

**graphic character.** See *character graphic*.

**Graphic Character Global Identifier (GCGID).** An alphanumeric character string used to identify a specific graphic character. A GCGID can be from 4 to 8 bytes long.

**graphic character identifier.** See *character identifier*.

**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.

**GRID.** See *global resource identifier*.

**grid.** Format for the frame output on a microfiche—row, column, numeric, alpha, start alphanumeric value.

**GTF.** Generalized trace facility.

**guaranteed print labeling.** A method of print labeling that ensures the integrity of the identification label by preventing the user from writing over the label.

## H

**HAID.** Host-assigned ID.

**hard page segment.** (1) A page segment that is declared in the Map Page Segment structured field and loaded in the printer as a resource that can be reused during the job without being reloaded to the printer. (2) A page segment that is contained within another element, as in an inline resource. Contrast with *soft page segment*.

**hardcopy.** (1) A copy of a display image generated on an output device such as a printer or a plotter, in a form that can be carried away. (T) (2) A printed copy of machine output in a visually readable form—for example, printed reports, listings, documents, and summaries.

**hardware default font.** The font that the printer uses if no other font is specified.

**HCD.** Hardware configuration definition.

**hexadecimal.** Pertaining to a numbering system with base of 16; valid numbers use the digits 0 through 9 and the characters A through F, where A represents 10 and F represents 15.

**host font.** See *host resource*.

**host processor.** The processing unit to which the page printers are attached through a data-transfer interface.

**host resource.** A resource found either in a system library, in a user library, or inline in the print data set.

**host system.** (1) A data processing system that prepares programs and the operating environments for another computer or controller. (2) The data processing system to which a network is connected and with which the system can communicate.

**HRI.** Human readable interpretation.

## I

**IBM compatibility fonts.** A group of fonts supplied as part of Print Services Facility, Print Management Facility, and Application System/400. Many of these fonts are derived from fonts created for specific IBM printers (such as the IBM 3800 Model 1) or applications (such as Document Composition Facility). The fonts are called compatibility fonts because they make it possible

for applications created for the 3800 Model 1 printer to be migrated to page printers with no need to change the fonts specified in the applications. Examples of IBM compatibility fonts are APL, Boldface, Document, Essay, Format, Gothic, Letter Gothic, Orator, Prestige, Roman, Script, Serif, and Text type families, as well as a set of Proprinter Emulation fonts.

**ID.** Identification.

**identification label.** In PSF, a protected set of resources (fonts, overlays, page segments) that are used to label PSF output for security purposes. Contrast with *security label*.

**image.** A pattern of toned and untoned pels that form a picture.

**image block.** A structure that contains a raster pattern and the instructions for placing the pattern on the page.

**image cell.** A part of an image that saves storage by defining only part of a raster pattern. Each image cell must also contain information defining the placement of its raster pattern within the complete image. An image cell can be replicated to fill a defined area.

**image data.** A pattern of bits with 0 and 1 values that define the pels in an image. (A 1-bit is a toned pel.)

**imbedded text control.** See *embedded text control*.

**IML.** Initial microprogram load.

**IMM.** Invoke Medium Map.

**impact printer.** A printer in which printing results from mechanical impacts. Contrast with *nonimpact printer*.

**impression.** The data printed on one side of a sheet. Printer speed is often measured in impressions per minute (ipm).

**initial microprogram load (IML).** (1) The action of loading microprograms into computer storage. (2) The action of loading licensed internal code from a diskette to writable control storage.

**initial program load (IPL).** (1) The initialization procedure that causes an operating system to begin. (2) The process of loading system programs and preparing a system to run jobs.

**initialize.** (1) In programming languages, to give a value to a data object at the beginning of its lifetime. (I) (2) To set counters, switches, addresses, or contents of storage to zero or other starting values at the beginning of, or at prescribed points in, the operation of a computer routine. (A) (3) To prepare for use—for example, to initialize a diskette. See also *initial program load*.

**inline.** The direction of successive characters in a line of text. Synonymous with *inline direction*.

**inline direction.** The direction of successive characters in a line of text.

**inline resource.** A resource contained in the print data set.

**input/output (I/O).** Pertaining to a device whose parts can perform an input process and an output process at the same time. (I)

**installation exit.** A subcomponent, installed and maintained for or by a customer installation, that provides control or functions specific to that installation, such as calling exits; providing defaults for job header, trailer, and data-set header separator-page exits; and supporting customer-written exits for logical records, SMF records, message processing, and resource management. PSF invokes these exits at certain predetermined times.

**intelligent printer data stream (IPDS).** An all-points-addressable data stream that makes it possible to position text, images, and graphics at any defined point on a printed page.

**interface.** A shared boundary. An interface can be a hardware component used to link two devices, or it can be a portion of storage or registers accessed by two or more computer programs.

**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 family.

**intermediate device.** In the IPDS architecture, a device that operates on the data stream and is situated between a printer and a presentation services program in the host. Examples are devices that capture and cache resources and devices that spool the data stream.

**internal copy group.** A copy group within a print data set instead of within a form definition. See also *copy group*. Synonymous with *internal medium map*.

**internal medium map.** The term used in MO:DCA architecture to describe a medium map within a print data set instead of a form definition. See also *medium map*. Synonymous with *internal copy group*.

**internal object.** Groups of structured fields that can be included as part of a resource or a print data set but cannot be accessed separately.

**intervention-required exception.** An error that causes printing to stop until an operator performs a required action.

**Internet Protocol (IP).** In TCP/IP, a protocol that routes data from its source to its destination in an Internet environment.

**Internet Protocol (IP) address.** The 32-bit address defined by the Internet Protocol, standard 5, Request for Comment (RFC) 791. It is usually represented in dotted decimal notation.

**I/O.** Input/output.

**I/O error manager.** The PSF subcomponent that analyzes I/O errors, determines the recovery action, and directs I/O error recovery and cleanup.

**IOCA.** Image object content architecture.

**IOS.** I/O supervisor.

**IPDS.** Intelligent printer data stream. This is the strategic AFP printer data stream.

**IPL.** Initial program load.

**ISO forms.** Pertaining to a set of paper sizes selected from those made standard by the International Organization for Standardization for use in data processing.

## J

**jam recovery copy counter.** A counting device in a printer that identifies how many copies of a page have successfully passed the printer-defined jam recovery point but have not been counted by the jam recovery page counter.

**jam recovery page counter.** A counting device in a printer that identifies how many pages have successfully passed the printer-defined jam recovery point. See also *fuser counter*.

**JCL.** Job control language.

**JES.** Job entry subsystem.

**job control language (JCL).** A language of control statements used to identify a computer job or describe its requirements to an operating system.

**job entry subsystem (JES).** An MVS subsystem that receives jobs into the system, converts them to internal format, selects them for execution, processes their output, and purges them from the system.

**job header.** A page in the printed output that indicates the beginning of a user job. A job can contain one or more data sets. 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.

**JSI.** Job submission interface.

## K

**kanji.** Nonphonetic Chinese characters used in writing Japanese. In a font representing kanji characters, each character is represented by a double-byte font.

**Kerning.** The printing of adjacent graphic characters so that their areas overlap on the left or right side.

## L

**LAN.** 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. Contrast with *portrait page presentation*.

**LASI.** Library access system interface.

**LCPC.** Load Code Page Control (IPDS command).

**LF.** Load Font (IPDS command).

**LFC.** Load Font Control (IPDS command).

**LFCS.** Load Font Char Set Control (IPDS command).

**LFE.** Load Font Equivalence (IPDS command).

**library.** A file or a set of related files—for example, a page definition library containing one or more page definition files. A library often exists as a partitioned data set in MVS, and as a minidisk in VM.

**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.** Synonym for *resource object*.

**LID.** Local identifier.

**ligature.** A single character representing two or more input characters—for example, *ff*, *ffi*, and *ffl* are characters that may be presented as ligatures.

**line data.** Data prepared for printing on a line printer such as a 3800 Model 1 Printing Subsystem. Line data is usually characterized by carriage-control characters and table reference characters. Contrast with MO:DCA-P data.

**line descriptor.** Specifications that describe how input data records are formatted into individual print lines. Line descriptors are interpreted by PSF when formatting printed output.

**line merging.** Printing two or more records of line data at the same location on the page. Line merging is used with line data to mix different fonts on the same line, to underscore or overstrike, and, on impact printers, to create darker print.

**line-mode printing.** Every 3800 printer can operate in “line mode” as a non-AFP printer. Print jobs that run in “line mode” do not use PSF or 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. Contrast with *page printer*.

**lines per inch (lpi).** (1) A unit of measurement for the specification of baseline placement. (2) A measure of the number of lines per vertical inch of paper.

**local area network (LAN).** (1) A computer network located on a user's premises within a limited geographical area. Communication within a local area network is not subject to external regulations; however, communication across the LAN boundary may be subject to some form of regulation. (T) (2) A network in which a set of devices are connected to one another for communication and that can be connected to a larger network.

**local-attached.** In PSF, an SNA-attached device that does not have a communications controller in its configuration. For example, a 3812 printer connected to a channel-attached 3174 control unit that is defined to the host system through VTAM is considered a local-attached printer. Contrast with *communication-attached*.

**local identifier.** A 1-byte identifier assigned to parts of the data stream to facilitate PSF processing. For example, the Map Coded Font structured field assigns each coded font a local identifier. When a coded font is required for processing, this identifier is specified in the Set Coded Font Local text control.

**logical page.** (1) The area on a surface of a form that is formatted for printing. (2) A collection of data that can be printed on one side of a sheet of paper. (3) A presentation space. One or more object areas or data

blocks can be mapped to a logical page. A logical page has specifiable characteristics, such as size, shape, orientation, and offset, and is rectangular. Orientation and offset are specified relative to a medium coordinate system. See also *form* and *page*.

**logical page origin.** (1) The point on the logical page from which positions of images, graphics, page overlays, and text with 0-degree inline direction are measured. (2) The point on the logical page represented by Xp=0, Yp=0 in the Xp coordinate system.

**logical unit (LU; L-unit).** L-unit (1) In SNA, a port through which an end user accesses the SNA network to communicate with another end user, and through which the end user accesses the functions provided by system services control points (SSCPs). A logical unit can support at least two sessions, one with an SSCP and one with another logical unit, and may be able to support many sessions with other logical units. (2) A unit of linear measurement. For example, in Mixed Object Document Content Architecture (MO:DCA) and AFP data streams, the following measurements are used:

1 L-unit = 1/1440 inch

1 L-unit = 1/240 inch

**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 table of macro-generated constants that associate a logon-mode name with a set of session parameters. Each logon mode is identified by a logon-mode name.

**lpi.** Lines per inch.

**LU (logical unit) base.** The linear measurement base. The value defines, for any object, the meaning of the L-unit values. See also *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.** Synonym for *macroinstruction*.

**macroinstruction.** An instruction that causes the execution of a predefined sequence of instructions.

**magnetic ink character recognition (MICR).** Recognition of characters printed with ink that contains particles of a magnetic material. (I) (A)

**mandatory print labeling (MPL).** A class defined to the resource access control facility (RACF) that causes PSF to automatically label separator pages and data pages and to enforce the user-printable area.

**mapping.** See *font mapping*.

**marking.** A method of updating certain structured fields to identify a resource as printer-resident.

**maximum transmission unit.** In LANs, the largest possible unit of data that can be sent on a given physical medium in a single frame.

**media origin.** The first hardware addressable point on the physical page. The 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 top edge of the medium.

**medium.** (1) A physical carrier of electrical energy. (2) A physical material (for example, paper) in or on which data is represented (for example, printed). See also *form*.

**medium map.** A term used in MO:DCA architecture to describe 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. Synonymous with *copy group*. See also *internal medium map*.

**medium overlay.** An electronic overlay that is invoked by the medium map of a form definition for printing at a fixed position on the form. See *page overlay*.

**member ID.** The member name of a resource object, minus the 2-character prefix. For example, BITR is the member ID of the font whose member name is X1BITR.

**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) A data set on disk storage that contains queues of messages awaiting transmission to particular terminal operators or to the host system. (2) 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.

**metrics.** Synonym for *font metrics*.

**MICR.** Magnetic ink character recognition.

**microfiche.** Microfilm output on a rectangle piece of film, consisting of a matrix arrangement of data frames, index frames, and title information.

**microfilm.** A film containing a photographic record of printed matter, on a reduced scale.

**microfilm device.** Non-IBM 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 non-IBM utility that builds an object container for microfilm setup data and places it in the AFP library.

**Mixed Object Document Content Architecture (MO:DCA).** An IBM-architected, device-independent data stream for interchanging documents.

**mixed-pitch font.** A font that simulates a typographic font. The characters are in a limited set of pitches—for example, 10 pitch, 12 pitch, and 15 pitch.

**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.

**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 data page.** A page of print data consisting entirely of structured fields.

**MO:DCA document.** (1) A new term for the obsolete term *composed-text document*. (2) Data composed entirely of structured fields and containing a Begin Document structured field and an End Document structured field.

**MO:DCA print data set.** A print data set consisting entirely of structured fields.

**MO:DCA-P.** See *Mixed Object Document Content Architecture-Presentation Format*.

**modified-default form definition.** A form definition that was the default specified in the PRINTDEV statement of the PSF startup procedure, and that has been modified by the groupvalue parameter of the COPIES parameter or by the FLASH parameter in the JCL statement.

**modified-default page definition.** A page definition that was the default specified in the PRINTDEV statement of the PSF startup procedure, and that has been modified by a font list specified in any of the following:

- The CHARS parameter from the user JCL or the PRINTDEV statement
- The UCS parameter from the user JCL
- The JES default font in the current printer setup

**MOF.** Metric-only font.

**monospaced font.** A font in which the graphics characters have a uniform character increment. Synonymous with *uniformly spaced font*. Contrast with *proportionally spaced font*.

**MPL.** Mandatory print labeling.

**MTU.** Maximum transmission unit.

**multiple-entry font.** A font with multiple entries in the Map Coded Font 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.)

**multiple-up.** The printing of more than one page of application data on a single surface of a sheet of paper.

**Multiple Virtual Storage (MVS).** An IBM operating system consisting of MVS/System Product Version 1 and the MVS/370 Data Facility Product operating on a System/390 processor.

**multiplexer.** (1) 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. (T) (2) A device capable of interleaving events of two or more activities or capable of distributing events of an interleaved sequence to the respective activities. (A)

**multiplexer channel.** A channel designed to operate with a number of I/O devices simultaneously. Several I/O devices can transfer records at the same time by interleaving items of data.

**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. (I) (A)

**multipoint network.** More than two devices sharing the same transmission line at the same time.

**MVS.** Multiple Virtual Storage.

**MVS/ESA.** Multiple Virtual Storage/Enterprise System Architecture.

## N

**NA.** Not applicable.

**NACK.** See *negative acknowledgment reply*.

**NCP.** Network Control Program.

**NCP generation.** The process by which the host processor assembles and link-edits a macroinstruction program to produce a Network Control Program.

**negative acknowledge reply (NACK).** A reply from a printer to a host indicating that an exception has occurred.

**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.

**nonimpact printer.** A printer in which printing is not the result of mechanical impacts—for example, thermal printers, electrostatic printers, and photographic printers. Contrast with *impact printer*.

**NOP.** No operation.

**nonprocess runout (NPRO).** An operation that moves paper through the paper path without printing.

**nonswitched line.** A telecommunication line on which communications do not have to be established by dialing. Contrast with *switched line*.

**normal duplex printing.** Printing on both sides of the paper so that the sheets can be bound on the long edge of the paper. Contrast with *simplex printing*. See also *tumble duplex printing*.

**notify subtask (NST).** (1) A PSF subcomponent that returns processed data sets to JES or to the Direct Printer Services Subsystem 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.** Nonprocess runout.

**NST.** Notify subtask.

**null name.** A token name with X'FFFF' in the first 2 bytes.

**null value.** A parameter for which no value is specified.

**numeric data.** (1) Data represented by numerals. (2) Data in the form of numerals and special characters—for example, a date represented as 81/01/01. See also *character data*.

**N\_UP printing.** In basic N\_UP printing, the dividing 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.** (1) A resource or a sequence of structured fields within a larger entity, such as a page segment or a composed page. (2) A collection of structured fields, of which the first provides a begin-object function and the last provides an end-of-object function. One example of an object is a form definition.

**object container.** (1) A set of structured fields used to carry object data for a variety of objects. (2) A MO:DCA-architected envelope that contains data to be processed by COM; transparent to PSF. This data is classified as “non-presentation” data. A MO:DCA “Registered Object Identification” (ROID) field is defined for the object container to denote the data type. For a microfilm device, this ROID denotes the contained data as either setup data or tape label data.

**OC.** See *Object container*.

**offset stacking.** A function that offsets printed output pages for easy separation of print jobs.

**OGL/370.** Overlay Generation Language/370.

**OPC.** Obtain printer characteristics.

**Operating System/2 (OS/2).** IBM's operating system for the IBM Personal System/2 or a compatible.

**Operating System/400 (OS/400).** IBM's operating system for Application System/400.

**option.** (1) A specification that can be included in a statement to specify details about the execution of a program. (2) A choice offered from a list of possibilities.

**orientation.** 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. This usually applies to blocks of information. Character rotation applies to individual characters. See also *text orientation*.

**origin.** A pel position from which the placement and orientation of text, images, and page segments are specified. For example, pages, overlays, and page segments have origins.

**OS/2.** Operating System/2.

**OS/4.** Operating System/400.

**outline font.** A font-shape technology in which the graphic character shapes are represented in digital form by a series of mathematical expressions that define the outer edges of the strokes. The resulting graphic character shapes can be either solid or hollow. Outline fonts can be scaled (sized) to any size. The name of an IBM outline font character set has the prefix *CZ*. Contrast with *raster font*.

**overlay.** See *electronic overlay*.

**Overlay Generation Language/370 (OGL/370).** An IBM licensed program for use in creating graphics by designing objects for electronic overlays, such as lines, boxes, shadings, and irregular shapes.

**OVP.** Operator viewing page.

## P

**packet.** In data communication, a sequence of binary digits, including data and control signals, that is transmitted and switched as a composite whole. The data, control signals, and, possibly, error control information are arranged in a specific format. (I)

**page.** (1) A collection of data that can be printed on a one side of a physical sheet of paper or form. (2) The boundary for determining the limits of printing. See also *logical page* and *physical page*.

**PAGEDEF.** A JCL parameter that specifies a page definition. See *page definition*.

**page definition.** A resource used by PSF that defines the rules for transforming line data into MO:DCA-P data and text controls.

**page environment.** The size of the page or overlay control data to be used in MO:DCA-P data text blocks, and the page segments or fonts to be used.

**page format.** A subset of a page definition, containing controls that govern the arrangement of text on a page. See *data map*.

**page mode.** The mode of operation in which a page printer can accept a page of data from a host processor to be printed on an all-points-addressable output medium. Data may consist of pages composed of text, images, overlays, or page segments.

**page origin.** See *logical page origin*.

**page overlay.** An electronic overlay that can be invoked for printing and positioned at any point on the page by an Invoke Page Overlay structured field in the print data. See *medium overlay*.

**page position.** A control in the copy group to assign the top-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.** Any of a class of printers that accepts MO:DCA pages, constructed of page data and images, among other things. A page printer is a device that prints one page as a unit. (I) (A) Contrast with *line printer*.

**Page Printer communication component (PPCC).** The access method that provides the SNA communication interface between printers and PSF.

**Page Printer Formatting Aid/370 (PPFA/370).** An IBM licensed program for use in creating and storing form definitions and page definitions, which are resource objects for print-job management. These stored objects can then be used to format printed output.

**page segment.** A resource containing MO:DCA data and images, prepared before formatting and included during printing. A page segment assumes the environment of an object in which it is included.

**paging.** (1) The transfer of pages between real storage and auxiliary storage. (I) (A) (2) In System 390 virtual storage systems, the process of transferring pages between real storage and external page storage.

**parameter.** (1) A variable that is given a constant value for a specified application and that may denote the application. (I) (A) (2) An item in a menu for which the user specifies a value or for which the system provides a value when the menu is interpreted. (3) Data passed between programs or procedures.

**partial page.** A page that does not contain all the intended data. Partial pages can be printed after an error is sensed.

**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. Contrast with *sequential data set*.

**pass-through.** To gain access to another network.

**pattern storage (PST).** An area of storage that holds the raster patterns for fonts and images.

**PCC.** Page Counter Control (IPDS command).

**PDEF.** Page definition.

**PCL.** Printer control language.

**PDS.** Partitioned data set.

**pel.** Picture element. Synonymous with *pixel*.

**pending page queue (PPQ).** In PSF, a list of pages that has been processed by PSF but has not been released from JES.

**physical page.** A single surface (front or back) of a form. See also *form* and *page*.

**physical medium.** A physical entity on which information is presented. Examples of physical media are display screens, paper, foils, microfilm, and labels.

**physical unit (PU).** The part of a communication controller or cluster controller that manages the resources attached to the node.

**picture element (pel).** (1) An element of a raster pattern about which a toned area on the photoconductor might appear. See *raster pattern*. (2) The smallest unit that can be addressed by a page printer. See *addressable point*.

**PID.** Process ID.

**PING command.** The command that sends an Internet Control Message Protocol (ICMP) echo request packet to a gateway, router, or host with the expectation of receiving a reply.

**pipeline.** (1) A serial arrangement of processors or a serial arrangement of registers within a processor. Each processor or register performs part of a task and passes results to the next processor; several parts of different tasks can be performed at the same time. (2) The hardware path between the channel station or received-page station and the stacker.

**pitch.** (1) A unit of measurement for the width of a printed character, reflecting the number of times a graphic character can be set in one linear inch. (2) The character size represented by the number of characters that can be printed horizontally in an inch; for example,

10 pitch has 10 graphic characters per inch. Uniformly spaced fonts are measured in pitch. Contrast with *point*.

**pixel.** Synonym for *pel*.

**PLR.** Priority level resource.

**PMF.** Print Management Facility.

**point.** A unit of about 1/72 inch used in measuring type. Contrast with *pitch*.

**point size.** The height of a font, in points.

**point-to-point network.** An arrangement in which two devices share the same transmission line at the same time.

**POR.** Power-on reset.

**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. Contrast with *landscape page presentation*.

**Positive Acknowledge Reply (ACK).** In the IPDS architecture, a reply to an IPDS command which has its ARQ flag on and in which no exception is reported.

**power-on reset (POR).** See *initial microprogram load*.

**PPCC.** Page Printer Communication Component.

**PPFA/370.** Page Printer Formatting Aid/370.

**PPP.** Primary print process.

**PPQ.** Pending page queue.

**PPWTR.** Page printing writer.

**prefix mapping.** The process that PSF uses to change font names at the coded font level to a name appropriate to the font technology that the printer supports. For example, XZ is the prefix for an outline font. When the printer supports only bounded-box raster fonts, PSF changes the prefix to X0. X0 is the prefix for the same font in the raster font library.

**preprinted form.** A sheet of paper containing a preprinted design of constant data into which variable data can be printed. See also *electronic overlay* and *forms flash*.

**presentation device.** A device that produces character shapes, graphics pictures, images, or bar code symbols on a physical medium. Examples of physical media are display screens, paper, foils, microfilm, and labels.

**presentation text.** Synonym for MO:DCA data.

**printable area.** The area on a sheet of the 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 three kinds of print data set are composed text, line format, and mixed format.

**print data stream.** The data stream that is created by PSF and transmitted to the printer.

**print direction.** (1) The direction in which characters are added to a line. (2) In PSF, the specification of inline direction for the printing of text.

**print file.** A file created by an application program that contains the actual information to be printed and some of the data that controls the format of the printing. Print files can contain MO:DCA data, line data, or a combination of MO:DCA and line data.

**print job.** The data that the user submits to PSF to be printed. The user can request that a print job be printed as though it were multiple data sets.

**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.** (1) The measure of printed output against existing standards and in comparison with jobs printed previously. (2) The ability of some page printers to print data at more than one level of print quality, such as “draft” and “near-letter” quality.

**Print Services Facility (PSF).** A licensed program that manages and controls the input data stream and output data stream required by supported IBM page printers. PSF combines print data with other resources and printing controls to produce AFP output.

**priority level resource (PLR).** A marked resource known to PSF and deleted according to a modified deletion algorithm. When resources 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, a functional unit that interprets and executes instructions. (I) (A)

**programming interface for customers.** Any product method that enables a customer-written program to 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 end-user interfaces, and others provide their services only to other products. Contrast with *end-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 RPQ may be used in conjunction with computing system RPQs to solve unique problems in data processing.

**program status word (PSW).** An area in storage used to indicate the order in which instructions are executed, and to hold and indicate the status of the computer system.

**program temporary fix (PTF).** A temporary solution or bypass of a problem diagnosed by IBM as resulting from a defect in a current unaltered release of the program.

**proportionally spaced font.** A typographic font, or in some usages a mixed-pitch font. See *typographic fonts* and *mixed-pitch fonts*.

**protocol.** The meanings of and the sequencing rules for requests and responses by which network addressable units (PU, LU, SSCP, and VTAM programs) in a communication network coordinate and control data transfer operations and other operations.

**PSAF.** Print Services Access Facility.

**PSC.** PostScript.

**PSEG.** Page segment.

**PSF.** Print Services Facility.

**PSF Direct.** A function of PSF/2 or PSF for AIX that enables another PSF program—PSF/VM, PSF/MVS, PSF/VSE, or PSF/400—to print remotely, using the SNA LU 6.2 protocol, on PSF/2 or PSF for AIX printers. The PSF program sends the print data stream directly to the PSF/2 or PSF for AIX printer, bypassing the OS/2 or AIX spool. The operator of the originating system controls printing on the PSF/2 or PSF for AIX printers as if the printers were attached to the originating system.

**PST.** Pattern storage.

**PSW.** Program status word.

**PTF.** Program temporary fix.

**PTOCA.** Presentation test object content architecture.

**PU.** Physical unit.

## R

**RACF.** Resource Access Control Facility.

**raster font.** A font technology in which the graphic characters are defined directly by the raster bit map. Contrast with *outline font*.

**raster overlay.** See *raster pattern overlay*.

**raster pattern.** (1) A series of picture elements (pels) in scan lines to form an image. See also *page image*. (2) A pattern of bits with 0 (off) and 1 (on) that define the pels in an image. A 1-bit is a toned pel.

**raster pattern overlay.** An overlay loaded in the printer as a raster pattern rather than as a sequence of printer commands. Contrast with *coded overlay*.

**raster pattern storage (RPS).** An area of storage that holds raster patterns for fonts and images.

**RCB.** Resource control block.

**RDW.** 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. Conventionally, the total range of addresses available to the user has been provided by the main storage. (I) (A)

**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.** A counting device in a printer that identifies how many pages have been successfully received and their syntax checked. Synonymous with *channel counter*.

**record descriptor word (RDW).** Data preceding a variable record that specifies the length of the entire record, including the RDW.

**record interface.** Coordinates the transmission of the printer resources needed to print a document.

**recovery termination manager (RTM).** A program that handles all normal and abnormal termination of tasks by passing control to a recovery routine associated with the terminated function.

**region size.** The amount of main storage available for a program to run in.

**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. This makes it possible to use the same value for different point sizes and different raster-pattern resolutions. The use of relative values necessitates the definition of the units of measure for the Em-square, the point of the font, and perhaps the pel resolution of the raster. Contrast with *fixed metrics*. See also *font metrics* and *bounded-box relative metrics*.

**reload function.** The Resource Exit can request that a resource be “reloaded..” PSF will not use an existing version of the resource, but will load the resource from a host library.

**repositioning.** A process in which Print Services Facility, following an indication from the printer or from JES of a potentially recoverable error, locates the correct spool record for recomposing one or more pages for printing.

**Request for Comments (RFC).** In Internet communications, the document series that describes a part of the Internet suite of protocols and related experiments. All Internet standards are documented as RFCs.

**request for price quote (RPQ).** See *computing system RPQ*.

**request unit (RU).** In SNA, a message unit that contains control information such as a request code, or function management (FM) headers and end-user data, or both. Synonymous with *request*.

**resident resource.** A resource that resides in a printer (fonts or symbol sets) or in DPF (fonts, page segments, and overlays). Resident resources are identified by the APSRMARK utility of PSF or by the resident font resource table APSRFTBL.

**resolution.** In computer graphics, a measure of the sharpness of an image, expressed as the number of lines and columns on the display screen or the number of pels per unit of linear measure.

**resource.** (1) A collection of printing instructions used by Print Services Facility in addition to the print data set to produce the printed output. PSF resources include coded fonts, font character sets, code pages, page segments, overlays, form definitions, and page definitions. (2) Any source of aid used for performing a task—for example, disk storage space, computer processing time, and communications lines.

**Resource Access Control Facility (RACF).** An IBM licensed program that provides for access control by

identifying and verifying the users to the system, authorizing access to protected resources, logging the detected unauthorized attempts to enter the system, and logging the detected accesses to protected resources.

**resource-constrained condition.** Occurs when the printer does not have enough storage for the resources required to print the current page.

**resource manager.** Any control program function responsible for allocating a resource. In PSF, it maintains the resource control structure used to manage resources; directs the retrieval of resources from libraries; deletes resources from the printer; calls for Resource Exit functions; and includes the APSRMARK utility with which resources can be marked as printer-resident or in DPF.

**resource name.** The name under which a resource object is stored; the first 2 characters indicate the resource type.

|                  |                    |
|------------------|--------------------|
| <b>X0-XG, XZ</b> | Coded font         |
| <b>T1</b>        | Code page          |
| <b>C0-CG, CZ</b> | Font character set |
| <b>S1</b>        | Page segment       |
| <b>F1</b>        | Form definition    |
| <b>P1</b>        | Page definition    |
| <b>O1</b>        | Overlay            |

**resource object.** A member in a resource library. Synonym for *library member*.

**RFC.** Request for Comments.

**RLE.** Run length encoding.

**ROID.** Registered object ID.

**root.** The user name for the system user with the most authority.

**rotation.** Synonym for *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, which use similar or different architectures, at the reference model network layer.

**routine.** A program, or a sequence of instructions, that is called by a program and that may have some general or frequent use. (I) (A)

**RP.** Raster processor.

**RPC.** Remove procedure call.

**RPQ.** Request for price quotation. See *computing system RPQ*.

**RPS.** Raster pattern storage.

**RRB.** Record resource block.

**RRLV.** Reasonable resource loaded value.

**RTM.** Recovery termination manager.

**RU.** Request unit.

**rule.** A solid or patterned line of any weight, extending horizontally or vertically across a column, row, or page.

## S

**SBCS.** Single-byte character set.

**SCSI.** Small computer system interface.

**SDF.** Structured data field.

**SDLC.** Synchronous Data Link Control.

**SDSF.** System Display and Search Facility.

**security definition.** In MVS, a member containing the definitions for one identification label. These definitions include instructions for the overlay name, and the size and origin of paper to be used.

**security definitions library.** In MVS, a partitioned data set, or a series of concatenated partitioned data sets, containing the security definitions for an entire system.

**security label.** In a trusted computing base, a security label used to maintain multiple levels of security on a system. This label is a combination of a security class and a security level. Contrast with *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.** Synonym for *page segment*.

**sense data.** (1) Data describing an I/O error. Sense data is presented to a host system in response to a Sense I/O command. (2) In SNA and TCP/IP, the data sent with a negative response indicating the reason for the response.

**sequence number.** A 2-byte field in a structured field introducer that identifies the position of the structured field in the data set.

**sequential data set.** In MVS, a data set whose records are organized on the basis of their physical positions, such as on magnetic tape. Contrast with *partitioned data set*.

**SF.** Structured field.

**sheet.** A single piece of paper. For cut-sheet printers, a synonym for *form*. The IPDS architecture defines four types of sheets: cut-sheet forms, continuous forms, envelopes, and computer output on microfilm. Each sheet has a front and a back side. Some types of media consist of multiple sheets. For example, a roll of continuous forms can be divided at the perforations into rectangular sheets. Each sheet usually has carrier or tractor-feed strips. Microfilm is another example of a medium comprising multiple sheets, whereas an envelope comprises only one sheet.

**shift-out, shift-in (SOSI).** Special EBCDIC control codes that exist in the data stream to indicate the switches between double-byte fonts and single-byte fonts.

**simplex printing.** Printing on only one side of the paper. Contrast with *duplex printing*.

**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. Contrast with *double-byte coded font*.

**single-entry font.** A font with a single entry in the Map Coded Font structured field. This includes all single-byte fonts and any double-byte fonts being defined as coded fonts.

**skip.** (1) To ignore one or more instructions in a sequence of instructions. (2) A move of the current print position to another location.

**SMF.** System management facilities.

**SMIT.** System management interface tool.

**SMP/E.** System Modification Program/Extended.

**SNA.** Systems Network Architecture.

**SNA-attached.** In PSF, pertaining to a device that is linked to the host system through 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, a 3820 attached to a communication network that uses the LU 6.2

communication protocol to receive data from a communication controller is considered an SNA-attached printer. Contrast with *channel-attached*.

**soft page segment.** A resource that is not declared in the Map Page Segment structured field but is sent to the printer inline with data. Contrast with *hard page segment*.

**SOSI.** Shift-out, shift-in.

**specified resolution.** See *format resolution*.

**spot carbon.** Paper from which carbon is omitted in certain areas to suppress the printing of data on specific copies.

**SRM.** System Resource Manager.

**SSCP.** System services control point.

**SSI.** Subsystem Interface.

**stacked-page counter.** A counting device that identifies how many pages have been successfully stacked.

**startup procedure.** A procedure used to start PSF and to specify PSF initialization parameters and libraries that contain system resources.

**station ID.** For the 3800 printer, a 2-byte pipeline counter that is incremented when the last copy of a page reaches the point in the hardware represented by that counter.

**STM.** Sense type and model.

**storage.** (1) A unit into which recorded text can be entered, in which it can be retained and processed, and from which it can be retrieved. (T) (2) The action of placing data into a storage device. (I) (3) A storage device. (A)

**structured field.** A self-identifying string of bytes and its data or parameters.

**structured field introducer.** The first 8 bytes of a structured field, which indicate its length, type, and number.

**subgroup.** A set of modifications within 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.

**subpage.** A part of a logical page on which line data can be placed. In the page definition, multiple subpages

can be placed on a physical page as specified in the print data.

**subsystem interface (SSI).** See *functional subsystem interface*.

**supervisor call (SVC).** An instruction that interrupts a program being run and passes control to the supervisor to perform a service specified by the instruction.

**suppression.** See *text suppression*.

**suppression local ID.** A value assigned in the Map Suppression coded field to a suppression named in a Line Descriptor structured field. This value is contained in the Begin and End Suppression text controls.

**suppression number.** An identification number from 1 to 127 generated by the Map Suppression structured field and assigned to text designated for suppression in the data map transmission subcase.

**swapping.** A process that interchanges the contents of an area of real storage with the contents of an area in auxiliary storage.

**switched line.** A telecommunication line on which communications are established by dialing. Contrast with *nonswitched line*.

**switched major node.** In VTAM, a major node whose minor nodes are physical and logical units attached by switched SDLC links.

**SVC.** Supervisor call.

**symbol set.** A type of font that resides in a printer but has fewer attributes than can be specified for resident coded fonts.

**synchronous.** (1) Pertaining to two or more processes that depend on the occurrences of a specific event such as common timing signals. (l) (A)  
(2) Occurring with a regular or predictable time relationship.

**Synchronous Data Link Control (SDLC).** A discipline for managing synchronous information transfer over a data link connection.

**SYSGEN.** System generation. (A)

**SYSIN.** System input stream.

**SYSOUT.** System output stream.

**system input stream (SYSIN).** A data definition (DD) statement used to begin an in-stream data set.

**system library.** A resource library specified in the PRINTDEV statement, as distinguished from *user library*.

**system management facilities (SMF).** An optional control program that provides the means for gathering and recording information that can be used to evaluate system use.

**System Modification Program (SMP).** An IBM licensed program used to install software and software changes.

**system output stream (SYSOUT).** A data definition (DD) statement used to identify a data set as a system output data set.

**Systems Network Architecture (SNA).** The description of the logical structure, formats, protocols, and operational sequences for transmitting information units through networks, and controlling the configuration and operation of those networks. See also *Page Printer Communication Component*.

**systems services control point (SSCP).** In SNA, a focal point with an SNA network for managing the configuration, coordinating network-operator and problem-determination requests, and providing directory support or other session services for end users of the network.

## T

**table reference character (TRC).** An optional control character in an input record that specifies the font to be used to print the record. The table reference character corresponds to a font number defined in a page definition font list or to the order of font names listed in the job control CHARS parameter.

**TCP.** Transmission Control Protocol.

**TCP/IP.** Transmission Control Protocol/Internet Protocol.

**TCP/IP-attached.** In PSF, a device that is linked to the MVS system through a TCP/IP network and receives data from the MVS system using the application-layer IBM protocol for IPDS printers. Some TCP/IP-attached printers require the i-data 7913 IPDS Printer LAN Attachment. Contrast with *SNA-attached* and *channel-attached*.

**telecommunication line.** The part of a data circuit, external to the equipment, that connects to a data-switching exchange. Contrast with *data link*.

**terminate.** (1) In SNA products, a request unit that is sent by a logical unit (LU) to its system services control point (SSCP) to cause the SSCP to start a procedure

for ending one or more designated LU-LU sessions. (2) To stop the operation of a system or device. (3) To stop the execution of a program.

**text.** A graphic representation of information on an output medium. Text can consist of alphanumeric characters and symbols arranged in paragraphs, tables, and columns.

**text control.** Structured field data that controls the format, placement, and appearance of text.

**text control sequence.** A text control and its associated data.

**text orientation.** A description of the appearance of text as a combination of print direction and character rotation.

**text suppression.** The intentional omission of sections of text in copy groups as specified in a parameter in the form definition.

**throughput.** (1) A measure of the amount of work performed by a printer over a period of time—for example, the number of impressions per minute. (2) A measure of the amount of work performed by a computer system over a period of time—for example, number of jobs per day. (l) (A) (3) In data communication, the total traffic between stations per unit of time.

**title.** Eye-readable strip made up of one and sometimes two adjacent frame rows, usually located at the top of each microfiche. Titles can also be used with 16mm microfilm.

**TLE.** Tag logical element.

**token name.** An 8-byte name that can be given to all internal objects and resources.

**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.** A record of the execution of a computer program. It exhibits the sequences in which the instructions were executed. (A)

**transaction ID.** An identifier assigned by Document Processor, associated with an I/O buffer, and representing a form of resource.

**transfer counter.** A counting device that identifies how many pages have been committed to printing and have been removed from the page buffer. Synonym for *committed page counter*.

**transmission.** (1) The sending of data from one place for reception elsewhere. (A) (2) The sending of a print data set to a printer.

**Transmission Control Protocol (TCP).** A communications protocol used in the Internet and in any network that follows the U.S. Department of Defense standards for inter-network protocol. TCP provides a reliable host-to-host protocol between hosts in packet-switched communications networks and in interconnected systems of such networks. It uses the Internet protocol as the underlying protocol.

**Transmission Control Protocol/Internet Protocol (TCP/IP).** A set of communications protocols that support peer-to-peer connectivity functions for both local and wide area networks.

**TRC.** Table reference character.

**tumble duplex printing.** Duplex printing for sheets that are to be bound on the short edge of the paper regardless of whether the presentation is portrait or landscape. Contrast with *normal duplex printing*.

**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.** A collection of fonts all having the same style, weight, and width. Each font differs from the others by point size or type family.

**type size.** (1) A measurement in pitch or points of the height and width of a graphic character in a font. (2) One of the many attributes of a font; other attributes, for example, are weight and width.

**typographic font.** A font in which the distance between characters varies. The distance from one character to another is adjusted to improve the visual flow of text by eliminating excess space.

## U

**UCS.** Universal character set.

**unbounded-box font.** A font designed to use unbounded character boxes. Contrast with *bounded-box font*.

**unbounded character box.** A character box design that requires the character position information to be included (as untuned pels) within the box. Contrast with *bounded character box*.

**unformatted print records.** 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.

**uniformly spaced font.** A font in which all the characters have the same character increment. Contrast with *proportionally spaced font*.

**units of measure.** Counting methods that are either specific or relative. See also *font metrics* and *bounded-box relative metrics*.

**universal character set (UCS).** A printer feature that permits the use of a variety of character arrays. Synonymous with *font*.

**UPA.** 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 the JCL USERLIB parameter.

**user printable area (UPA).** The area within the valid printable area (VPA) in which user-generated data can print without causing an exception condition. See also *valid printable area*.

**utility program.** (1) A computer program in general support of computer processes—for example, a diagnostic program, a trace program, or a sort program. (T) Synonymous with *service program*. (2) A program designed to perform an everyday task such as copying data from one storage device to another. (A)

## V

**valid printable area (VPA).** The intersection of the physical page with the current logical page or the current overlay. See also *user-printable area*.

**value.** A quantity assigned to a constant, a variable, a parameter, or a symbol in a command.

**virtual storage.** (1) The storage space which may be regarded as addressable main storage by the user of a computer system and 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. (I) (A) (2) Addressable space that is apparent to the user as the processor storage space, from which the instructions and the data are mapped into the processor storage locations. Synonymous with *virtual memory*.

**Virtual Telecommunications Access Method (VTAM).** A set of programs that maintains control of

the communication between terminals and application programs running under DOS/VS, OS/VS1, and OS/VS2 operating systems.

**VPA.** Valid printable area.

**VTAM.** Virtual Telecommunications Access Method.

## W

**WCS.** Writable control storage.

**WOC.** Write object container.

**WOCC.** Write object container control.

**writable control storage (WCS).** Printer storage in which data can be entered, held, and retrieved. Writable control storage contains licensed internal-code instructions and other control information, such as the print buffer.

**write-to-operator (WTO).** An system service that enables a program to write a message notifying the system console operator of conditions that may need correcting.

**WTO.** Write-to-operator.

## X

**XA.** Extended Architecture.

**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.

**XFP.** X-Series Future Platform.

**Xm, Ym coordinate system.** The media coordinate system.

**Xp, Yp coordinate system.** The logical page coordinate system.

## Y

**Y-axis.** In printing, an axis parallel to the direction in which the paper moves through the printer. See also *X-axis*.

**Y-extent.** A measurement along the Y-axis.



## Numerical Entries

**16mm.** Microfilm with frames that are arranged in a 16mm-high strip, usually 100 feet long, to fit in a viewer cartridge. Microfilm uses sequential retrieval marks for view positioning.



---

## Bibliography

This bibliography lists the titles of publications containing additional information about PSF, Advanced Function Presentation, the MVS and OS/390 operating systems, and related products.

The titles and order numbers may change from time to time. To verify the current title or order number, consult your IBM marketing representative.

---

### Print Services Facility (PSF) for OS/390

| Publication  | Order Number |
|--|--------------|
| <i>IBM IP PrintWay Guide</i>                                       | S544-5379    |
| <i>IBM NetSpool Guide</i>  | G544-5301    |
| <i>PSF for AIX: Upload Configuration Guide for SNA</i>             | S544-5422    |
| <i>PSF for AIX: Upload Configuration Guide for TCP/IP</i>          | S544-5423    |
| <i>PSF for OS/390 Collection Kit CD-ROM</i>                        | SK2T-9267    |
| <i>PSF for OS/390: Customization</i>                               | S544-5622    |
| <i>PSF for OS/390: Diagnosis</i>                                   | G544-5623    |
| <i>PSF for OS/390: Download for OS/390</i>                         | G544-5624    |
| <i>PSF for OS/390: Introduction</i>                                | G544-5625    |
| <i>PSF for OS/390: Licensed Program Specifications</i>             | G544-5626    |
| <i>PSF for OS/390: Messages and Codes</i>                          | G544-5627    |
| <i>PSF for OS/390: Program Directory</i>                           | None         |
| <i>PSF for OS/390: User's Guide</i>                                | S544-5630    |
| <i>AFP Conversion and Indexing Facility: User's Guide</i>          | S544-5285    |
| <i>PSF: Security Guide</i>   | S544-3291    |
| <i>Program Directory for IP PrintWay Feature of PSF for OS/390</i> | None         |
| <i>Program Directory for NetSpool Feature of PSF for OS/390</i>    | None         |

---

### Advanced Function Presentation (AFP)

| Publication   | Order Number |
|---|--------------|
| <i>Advanced Function Presentation: Printer Information</i>              | G544-3290    |
| <i>Advanced Function Presentation: Printer Summary</i>                  | G544-3135    |
| <i>Advanced Function Printing: Host Font Data Stream Reference</i>      | S544-3289    |
| <i>AFP Toolbox for Multiple Operating Systems User's Guide</i>          | G544-5292    |
| <i>AFP Workbench for Windows and OS/2: Using the Viewer Application</i> | G544-3813    |
| <i>Guide to Advanced Function Presentation</i>                          | G544-3876    |
| <i>Overlay Generation Language/370 User's Guide and Reference</i>       | S544-3700    |
| <i>IBM Page Printer Formatting Aid: User's Guide</i>                    | S544-5284    |
| <i>Printing and Publishing Cluster Collection CD-ROM</i>                | SK2T-2921    |

---

## Advanced Function Presentation (AFP) Architecture

| Publication  | Order Number |
|--|--------------|
| <i>Advanced Function Presentation: Programming Guide and Line Data Reference</i> | S544-3884    |
| <i>Bar Code Object Content Architecture Reference</i>                            | S544-3766    |
| <i>Font Object Content Architecture Reference</i>                                | S544-3285    |
| <i>Graphics Object Content Architecture Reference</i>                            | SC31-6804    |
| <i>Image Object Content Architecture Reference</i>                               | SC31-6805    |
| <i>Intelligent Printer Data Stream Reference</i>                                 | S544-3417    |
| <i>Mixed Object Document Content Architecture Reference</i>                      | SC31-6802    |
| <i>Presentation Text Object Content Architecture Reference</i>                   | SC31-6803    |

---

## Advanced Function Presentation for Microfilm

To order any of the following microfilm publications, see your representative for Anacomp, Inc.

| Publication  | Order Number              |
|--|---------------------------|
| <i>DataMASTER II for XFP2000-Linked Systems Operator Guide</i>         | DMIIR21                   |
| <i>XFP2000 Bit-Mapped Imaging User's Guide</i>                         | NW-04-9403                |
| <i>XFP2000 COM System Operator Guide</i>                               | 8000R21                   |
| <i>XFP2000 Message Guide</i>   | 8KMSGR11                  |
| <i>XFP2000 Reference</i>   | XF-07-9201,<br>XF-07-9302 |
| <i>XFP2000 Software Release Notes</i>                                  | XF-14-9301                |
| <i>XFP2000 User's Guide</i>  | XF-04-9302                |
| <i>XFP2000 Workstation Executive Installation and Operations Guide</i> | XF-12-9201                |

---

## Fonts

| Publication   | Order Number |
|---|--------------|
| <i>ABOUT TYPE: IBM's Technical Reference for 240-pel Digitized Type</i>               | S544-3516    |
| <i>IBM AFP Fonts: Font Samples</i>  | S544-3792    |
| <i>IBM AFP Fonts: Font Summary</i>  | G544-3810    |
| <i>IBM AFP Fonts: Font Summary for AFP Font Collection</i>                            | S544-5633    |
| <i>IBM AFP Fonts: IBM's Typographic Primer for Digitized Type</i>                     | G544-3183    |
| <i>IBM AFP Fonts: Introduction to Typography</i>                                      | G544-3122    |
| <i>IBM AFP Fonts: Technical Reference for Code Pages</i>                              | S544-3802    |
| <i>IBM AFP Fonts: Technical Reference for IBM Chinese, Japanese, and Korean Fonts</i> | S544-5330    |
| <i>IBM AFP Fonts: Technical Reference for IBM Expanded Core Fonts</i>                 | S544-5228    |
| <i>IBM AFP Fonts: Type Transformer User's Guide</i>                                   | G544-3796    |

---

## Text Processing

| Publication   | Order number |
|---|--------------|
| <i>DCF/DLF General Information</i>                                      | GH20-9158    |
| <i>Document Composition Facility: Bar Code User's Guide</i>             | S544-3115    |
| <i>Document Composition Facility: SCRIPT/VS Text Programmer's Guide</i> | SH35-0069    |
| <i>Publishing Systems BookMaster General Information</i>                | GC34-5006    |
| <i>Publishing Systems BookMaster User's Guide</i>                       | SC34-5009    |
| <i>Using DisplayWrite/370</i>   | SH12-5172    |

---

## InfoPrint Manager for AIX

| Publication   | Order Number |
|---|--------------|
| <i>IBM InfoPrint Manager for AIX: Administrator's Guide</i>       | S544-5595    |
| <i>IBM InfoPrint Manager for AIX: User's and Operator's Guide</i> | S544-5596    |

---

## Print Services Facility (PSF) for AIX

| Publication  | Order Number |
|--|--------------|
| <i>AIX and Related Products Documentation</i>  | SC23-2456    |
| <i>Facts About PSF for AIX</i>   | G544-5305    |
| <i>IBM Page Printer Formatting Aid/6000: User's Guide Version 2.1</i>                        | S544-3918    |
| <i>IBM Print Services Facility for AIX: AIX for Print Services Facility Users</i>            | G544-3766    |
| <i>IBM Print Services Facility for AIX: AFP Conversion and Indexing Facility Version 2.1</i> | G544-3930    |
| <i>IBM Print Services Facility for AIX: Guide for Printer and COM Operators</i>              | S544-5286    |
| <i>IBM Print Services Facility for AIX: Licensed Program Specifications</i>                  | G544-3815    |
| <i>IBM Print Services Facility for AIX: Print Administration</i>                             | S544-3817    |
| <i>IBM Print Services Facility for AIX: Print Services Facility for AIX Users</i>            | G544-3814    |
| <i>IBM Print Services Facility for AIX: Print Submission</i>                                 | S544-3878    |

---

## Print Services Facility for OS/2 (PSF/2)

| Publication   | Order Number |
|---|--------------|
| <i>IBM Print Services Facility for OS/2: A Guide to Using PSF/2</i>   | G544-5225    |
| <i>IBM Print Services Facility for OS/2: An Installation Cookbook for AS/400 and Token Ring Networks</i>            | G544-3966    |
| <i>IBM Print Services Facility for OS/2: An Installation Cookbook for System/370 and Token Ring Networks</i>        | G544-3965    |
| <i>IBM Print Services Facility for OS/2: Facts about PSF/2</i>  | G544-3890    |
| <i>IBM Print Services Facility for OS/2: Network Configuration Guide for System/370 and Communication Manager/2</i> | S544-3911    |
| <i>IBM Print Services Facility for OS/2: Printer Attachments Guide</i>  | G544-5215    |

---

## EDMSuite OnDemand

| <b>Publication</b>  | <b>Order Number</b> |
|---|---------------------|
| <i>EDMSuite OnDemand: Administrator's Reference</i>                                   | G544-5293           |
| <i>EDMSuite OnDemand: Installation and Configuration Guide for UNIX Servers</i>       | G544-5598           |
| <i>EDMSuite OnDemand: Installation and Configuration Guide for Windows NT Servers</i> | G544-5526           |
| <i>EDMSuite OnDemand: Introduction and Planning Guide</i>                             | G544-5281           |

---

## Printers

| <b>Publication</b>   | <b>Order Number</b> |
|--|---------------------|
| <i>Reference Manual for the IBM 3800 Printing Subsystem Models 3 and 6</i>                               | GA32-0050           |
| <i>IBM PagePrinter 3812 Introduction and Planning Guide</i>  | G544-3265           |
| <i>IBM 3816 Page Printer Operating Instructions</i>  | GA34-2075           |
| <i>IBM 3825 Page Printer Product Description</i>   | G544-3482           |
| <i>IBM 3827 Page Printer Product Description</i>   | G544-3194           |
| <i>IBM 3828 Advanced Function MICR Printer Product Description</i>                                       | G544-3361           |
| <i>IBM 3900 Advanced Function Printer Product Description</i>  | GA32-0135           |
| <i>IBM 3912 and 3916 Page Printer Getting Started</i>  | S544-3898           |
| <i>IBM LaserPrinter 4028 Introduction and Planning Guide</i>   | S544-4258           |
| <i>User's Guide for the IBM LaserPrinter 4029 Series</i>   | ZA40-0542           |
| <i>IBM 4224 Printer Models 1xx and 2xx Product and Programming Description Manual</i>                    | GC31-2551           |
| <i>IBM 4230 Printer Product and Programming Description Models 102 and 202</i>                           | GC40-1701           |
| <i>IBM 4234 Printer Models 007, 008, 011, and 012 Product and Programming Description</i>                | GC31-3879           |
| <i>Attachment Configuration Handbook for Printers that Use the Advanced Function Common Control Unit</i> | S544-3977           |
| <i>InfoPrint 60 Finisher Application Design Guide</i>  | S544-5643           |
| <i>InfoPrint 62 Introduction and Planning Guide</i>  | G544-5384           |
| <i>InfoPrint 3000 Introduction and Planning Guide</i>  | G544-5563           |
| <i>InfoPrint 4000 and 3900 Advanced Function Printers Introduction and Planning Guide</i>                | G544-5427           |
| <i>InfoPrint Color 100 Introduction and Planning Guide</i>   | G544-5612           |
| <i>InfoPrint Hi-Lite Color Introduction and Planning Guide</i>   | G544-5420           |
| <i>IOCP and ESCON Reference</i>  | GC38-0401           |

---

## i-data

| <b>Publication</b>   | <b>Order Number</b> |
|--|---------------------|
| <i>i-data 7913 IPDS Printer LAN Attachment for Ethernet Installation Guide</i>   | none                |
| <i>i-data 7913 IPDS Printer LAN Attachment for Token Ring Installation Guide</i> | none                |

## OS/390 Version 2 Release 6

| <b>Publication</b>  | <b>Order Number</b> |
|---|---------------------|
| <i>IBM Online Library Omnibus Edition OS/390 Collection</i>         | SK2T-6700           |
| <i>OS/390 HCD Planning</i>  | GC28-1750           |
| <i>OS/390 HCD User's Guide</i>                                      | SC28-1848           |
| <i>OS/390 JES2 Commands</i>   | GC28-1790           |
| <i>OS/390 JES2 Initialization and Tuning Guide</i>                  | SC28-1791           |
| <i>OS/390 JES2 Initialization and Tuning Reference</i>              | SC28-1792           |
| <i>OS/390 JES2 Messages</i>   | GC28-1796           |
| <i>OS/390 JES3 Commands</i>   | GC28-1798           |
| <i>OS/390 JES3 Initialization and Tuning Guide</i>                  | SC28-1802           |
| <i>OS/390 JES3 Initialization and Tuning Reference</i>              | SC28-1803           |
| <i>OS/390 JES3 Messages</i>   | GC28-1804           |
| <i>OS/390 MVS Authorized Assembler Services Guide</i>               | GC28-1763           |
| <i>OS/390 MVS Authorized Assembler Services Reference ALE-DYN</i>   | GC28-1764           |
| <i>OS/390 MVS Authorized Assembler Services Reference ENF-IXG</i>   | GC28-1765           |
| <i>OS/390 MVS Authorized Assembler Services Reference LLA-SDU</i>   | GC28-1766           |
| <i>OS/390 MVS Authorized Assembler Services Reference SET-WTO</i>   | GC28-1767           |
| <i>OS/390 MVS Diagnosis: Tools and Service Aids</i>                 | SY28-1085           |
| <i>OS/390 MVS Initialization and Tuning Reference</i>               | SC28-1752           |
| <i>OS/390 MVS IPCS User's Guide</i>                                 | GC28-1756           |
| <i>OS/390 MVS IPCS Commands</i>                                     | GC28-1754           |
| <i>OS/390 MVS JCL Reference</i>                                     | GC28-1757           |
| <i>OS/390 MVS JCL User's Guide</i>                                  | GC28-1758           |
| <i>OS/390 MVS Recovery and Reconfiguration Guide</i>                | GC28-1777           |
| <i>OS/390 MVS System Codes</i>                                      | GC28-1780           |
| <i>OS/390 MVS System Commands</i>                                   | GC28-1781           |
| <i>OS/390 MVS System Management Facilities (SMF)</i>                | GC28-1783           |
| <i>OS/390 MVS System Messages, Vol 1 (ABA-ASA)</i>                  | GC28-1784           |
| <i>OS/390 MVS System Messages, Vol 2 (ASB-EWX)</i>                  | GC28-1785           |
| <i>OS/390 MVS System Messages, Vol 3 (GDE-IEB)</i>                  | GC28-1786           |
| <i>OS/390 MVS System Messages, Vol 4 (IEC-IFD)</i>                  | GC28-1787           |
| <i>OS/390 MVS System Messages, Vol 5 (IGD-IZP)</i>                  | GC28-1788           |
| <i>OS/390 MVS Using the Functional Subsystem Interface</i>          | SC28-1911           |
| <i>OS/390 Printing Softcopy BOOKs</i>                               | S544-5354           |
| <i>OS/390 Security Server (RACF) General User's Guide</i>           | SC28-1917           |
| <i>OS/390 Security Server (RACF) Security Administrator's Guide</i> | SC28-1915           |
| <i>OS/390 Security Server (RACF) System Programmer's Guide</i>      | SC28-1913           |
| <i>OS/390 SDSF Guide and Reference</i>                              | SC28-1622           |
| <i>OS/390 SMP/E User's Guide</i>                                    | SC28-1740           |
| <i>OS/390 V2R6 PDF Library Collection CD-ROM</i>                    | SK2T-6718           |

## eNetwork Communications Server

| Publication  | Order Number |
|--|--------------|
| <i>OS/390 eNetwork Communications Server: IP Application Programming Interface Guide</i> | SC31-8516    |
| <i>OS/390 eNetwork Communications Server: IP Configuration Guide</i>                     | SC31-8513    |
| <i>OS/390 eNetwork Communications Server: IP Network Print Facility</i>                  | SC31-8522    |
| <i>OS/390 eNetwork Communications Server: IP Programmer's Reference</i>                  | SC31-8515    |
| <i>OS/390 eNetwork Communications Server: SNA Resource Definition Reference</i>          | SC31-8565    |

## MVS/Enterprise Systems Architecture (ESA) Version 5

| Publication   | Order Number |
|---|--------------|
| <i>MVS/ESA SP V5 Authorized Assembler Services Guide</i>  | GC28-1467    |
| <i>MVS/ESA SP V5 Diagnosis: Tools and Service Aids</i>  | LY28-1845    |
| <i>MVS/ESA SP V5 Hardware Configuration Definition: User's Guide</i>                                  | SC33-6468    |
| <i>MVS/ESA SP V5 Initialization and Tuning Reference</i>  | GC28-1452    |
| <i>MVS/ESA SP V5 IPCS Commands</i>  | GC28-1491    |
| <i>MVS/ESA SP V5 IPCS User's Guide</i>  | GC28-1490    |
| <i>MVS/ESA SP V5 JCL Reference</i>  | GC28-1479    |
| <i>MVS/ESA SP V5 JCL User's Guide</i>   | GC28-1473    |
| <i>MVS/ESA SP V5 JES2 Commands</i>  | GC23-1443    |
| <i>MVS/ESA SP V5 JES2 Initialization and Tuning Guide</i>   | SC23-1453    |
| <i>MVS/ESA SP V5 JES2 Initialization and Tuning Reference</i>   | SC23-1454    |
| <i>MVS/ESA SP V5 JES2 Messages</i>  | GC28-1488    |
| <i>MVS/ESA SP V5 JES3 Commands</i>  | GC23-1444    |
| <i>MVS/ESA SP V5 JES3 Initialization and Tuning Guide</i>   | SC23-1455    |
| <i>MVS/ESA SP V5 JES3 Initialization and Tuning Reference</i>   | SC23-1456    |
| <i>MVS/ESA SP V5 JES3 Messages</i>  | GC28-1489    |
| <i>MVS/ESA SP V5 Planning: B1 Security</i>  | GC28-1440    |
| <i>MVS/ESA SP V5 Planning: Operations</i>   | GC28-1441    |
| <i>MVS/ESA SP V5 Programming: Authorized Assembler Services Reference, Volume 1 (ALESERV-DNALLOC)</i> | GC28-1475    |
| <i>MVS/ESA SP V5 Programming: Authorized Assembler Services Reference, Volume 2 (ENFREQ-ITTFMTB)</i>  | GC28-1476    |
| <i>MVS/ESA SP V5 Programming: Authorized Assembler Services Reference, Volume 3 (LLACOPY-SDUMPX)</i>  | GC28-1477    |
| <i>MVS/ESA SP V5 Programming: Authorized Assembler Services Reference, Volume 4 (SETFRR-WTOR)</i>     | GC28-1478    |
| <i>MVS/ESA SP V5 System Codes</i>   | GC28-1486    |
| <i>MVS/ESA SP V5 System Commands</i>  | GC28-1442    |
| <i>MVS/ESA SP V5 System Management Facilities</i>   | GC28-1457    |
| <i>MVS/ESA SP V5 System Messages, Volume 1</i>  | GC28-1480    |
| <i>MVS/ESA SP V5 System Messages, Volume 2</i>  | GC28-1481    |
| <i>MVS/ESA SP V5 System Messages, Volume 3</i>  | GC28-1482    |



| <b>Publication</b>  | <b>Order Number</b> |
|---|---------------------|
| <i>MVS/ESA SP V5 System Messages, Volume 4</i>                | GC28-1483           |
| <i>MVS/ESA SP V5 System Messages, Volume 5</i>                | GC28-1484           |
| <i>MVS/ESA SP V5 Using the Functional Subsystem Interface</i> | SC28-1505           |
| <i>RACF V2 General User's Guide</i>                           | SC23-3728           |
| <i>RACF V2 Security Administrator's Guide</i>                 | SC23-3726           |
| <i>RACF V2 System Programmer's Guide</i>                      | SC23-3725           |
| <i>SMP/E R8 User's Guide</i>                                  | SC28-1302           |
| <i>System Display and Search Facility Guide and Reference</i> | SC23-0408           |

---

## TCP/IP

| <b>Publication</b>  | <b>Order Number</b> |
|---|---------------------|
| <i>Internetworking with TCP/IP, Principles, Protocols, and Architecture</i> | SC31-6144           |
| <i>TCP/IP Tutorial and Technical Reference</i>                              | GG24-3376           |

---

## TCP/IP for MVS

| <b>Publication</b>   | <b>Order Number</b> |
|--|---------------------|
| <i>TCP/IP for MVS: Customization and Administration Guide</i>      | SC31-7134           |
| <i>TCP/IP for MVS: Application Programming Interface Reference</i> | SC31-7187           |
| <i>TCP/IP for MVS: Programmer's Reference</i>                      | SC31-7135           |

---

## VTAM and NCP

| <b>Publication</b>   | <b>Order Number</b> |
|--|---------------------|
| <i>ACF/INCP/SSP Version 3 Resource Definition Reference</i>                | SC30-3254           |
| <i>Advanced Communications Function for VTAM, Version 3, Programming</i>   | SC23-0115           |
| <i>Advanced Communications Function for VTAM, Version 2, Programming</i>   | SC27-0611           |
| <i>Advanced Communications Function for VTAM, Version 3, Customization</i> | SC23-0112           |
| <i>NCP Resource Definition Guide</i>                                       | SC30-3449           |
| <i>NCP: Resource Definition Reference</i>                                  | SC30-3254           |
| <i>NCP: System Support</i>   | SC30-3447           |
| <i>Network Program Products: Bibliography and Master Index</i>             | GC30-3353           |
| <i>Network Program Products: General Information</i>                       | GC23-0108           |
| <i>VTAM Resource Definition Reference</i>                                  | SC31-6552           |
| <i>VTAM Version 3 Diagnosis Reference</i>                                  | LY30-5582           |
| <i>VTAM Version 3 Messages and Codes</i>                                   | SC31-6433           |
| <i>VTAM Version 3 Operation</i>  | SC23-0113           |
| <i>VTAM Version 4 Diagnosis Guide</i>                                      | LX75-0204           |
| <i>VTAM Version 4 Diagnosis Quick Reference</i>                            | LX75-0203           |
| <i>VTAM Version 4 Messages and Codes</i>                                   | SC31-6418           |

---

## Systems Network Architecture

| <b>Publication</b>   | <b>Order Number</b> |
|--|---------------------|
| <i>SNA Customization</i>   | LY43-0110           |
| <i>SNA Formats (SNA Reference Summary)</i>                           | GA27-3136           |
| <i>SNA Resource Definition Reference</i>                             | SC31-8565           |
| <i>SNA Technical Overview</i>  | GC30-3073           |
| <i>SNA Transaction Programmer's Reference Manual for LU Type 6.2</i> | GC30-3084           |
| <i>Systems Network Architecture Formats</i>                          | GA27-3136           |

---

## AS/400

| <b>Publication</b>                            | <b>Order Number</b> |
|---|---------------------|
| <i>AS/400 Data Description Specifications</i> | SC41-962            |
| <i>AS/400 Command Language Reference</i>      | SC41-3722           |
| <i>AS/400 Printer Device Programming</i>      | SC41-3713           |

---

## Information/System

| <b>Publication</b>                        | <b>Order Number</b> |
|---|---------------------|
| <i>Introducing the Information/Family</i> | GC34-4045           |

---

# Index

## A

- abends, restartable
  - controlling in APSUX07 204
  - understanding 255
- ACQINTV parameter 41
- active environment group (AEG) 203
- adding font mapping
  - outline to raster 152
  - raster to outline 151
- Address Resolution Protocol tables 247
- AEG (active environment group) 203
- AFP performance 229
  - See also performance factors
- AFP/COM
  - attaching channel-attached printers 259
  - attaching ESCON-attached microfilm device 259
  - channel-attached 7
  - checkpointing JCL parameters 259, 268
  - COM-Recorder, SNA-attached 34
  - COMSETUP keyword 261
  - defining a microfilm device to PSF 261
  - defining microfilm device to OS/390 7
  - defining microfilm device to PSF 260
  - defining to JES2 259
  - defining to JES3 259
  - ESCON-attached 7
  - FONTDD keyword 261
  - hardware requirements 257
  - libraries, PSF 264
  - library data set 265
  - NPRO JCL parameter 259
  - OBJCOND keyword 261
  - object containers 325
  - operating a microfilm device 268
  - performance considerations 268
  - PRINTDEV parameters, PSF 261
  - PSF system programmer tasks 257
  - redirecting messages 264
  - resources 265
  - security considerations 268
  - suppressing (redirecting) messages 264
  - switching setup procedures 268
  - system overview 257
  - tasks, PSF system programmer 257
  - utility 265
- algorithm, resource-deletion 219
- anamorphic scaling 317
- APPL statement 15
- APPLID parameter 41
- APSLCRL parameter area 308
- APSRCF30 font conversion program 327
  - description 327
  - error handling 328
- APSRFTBL resident font table 134
- APSRGTNM grid parts to font member names 340
- APSRMARK
  - considerations 136
  - control statement parameters, general 140
  - control statement parameters, specific 138
  - control statement syntax 137
  - control statements for DPF 160
  - control statements for resident fonts 141
  - description of 135
  - diagnosing problems 143
  - long report format 273
  - long report, generating 142
  - page segments used with DPF 164
  - procedure for using 137
  - run APSW4028 first 144
  - short report, generating 142
- APSRNTGM font member names to grid parts 343
- APSUBTBL block-letter tables 184
- APSURLST parameter area 311
- APSUX01 exit 179
- APSUX02 exit 179
- APSUX03 exit 179
- APSUX04 exit 187
  - input 187
  - output 188
  - programming considerations 189
- APSUX05 exit 193
  - input 194
  - output 194
  - programming considerations 194
- APSUX06 exit 195
  - input 196
  - output 196
  - programming considerations 196
- APSUX07 exit 198
  - begin-data-set call (BDSC) 212
  - double-byte font 216
  - field descriptions
    - APSLCRL fields 310
    - APSURLST fields 313
    - CRL data area 311
    - RLSTHDR fields 318
    - XTP7 fields 302
  - initialization call (INIT) 211
  - invoking 198
  - processing indicators 319
  - programming considerations 200
    - 3800 compatible media origin 201
    - 3800 default font 207

APSUX07 exit (*continued*)  
   programming considerations (*continued*)  
     3800 forms marking 207  
     converting multiple-celled IM1 image 203  
     cut-sheet emulation mode 205  
     data-set termination 206  
     eject-to-front-facing function 203  
     ending the SNA LU1 conversation 210  
     form definitions 205  
     forms marking of an interrupt message page 205  
     inhibiting recovery 210  
     inline resources 208  
     IOCA, GOCA, and BCOCA 201  
     mapping to outline fonts 209  
     outline fonts 208  
     overlay 203  
     page definitions 205  
     page segment 200  
     recovering from font-not-found 209  
     redirecting message data set 202  
     requesting an acknowledgement 204  
     restartable abends 204  
     setting a format resolution value 207  
     suppressing interrupt message page 202  
     user libraries 208  
   resource deletion call (RDC) at DSE 219  
   resource load call (RLC) 218  
   resource-access call (RAC) 213  
   termination call (TERM) 221  
 APSUX14 exit 221  
   input 221  
   output 222  
   programming considerations 222  
 APSUX15 exit 222  
 APSUX16 223  
   begin-data-set call (BDSC) 223  
   end-data-set call (EDSC) 225  
   message call 224  
   programming considerations 225  
 APSWPROT procedure, for TCP/IP-attached  
   printers 248  
 APSXUCOM macro 183  
 ARP tables 247  
 ARPAGE statement, in TCP/IP profile 247  
 automatic and auto-resolution modes 127  
 auxiliary, default page and form definitions 217

## B

Basic N\_UP printing 121  
 begin-data-set call (BDSC)  
   APSUX07 212  
   APSUX16 223  
 bibliography 389  
 blank pages after separator pages 81

block size 241  
 block-letter routine 185  
 block-letter tables, APSUBTBL 184  
 bootp  
   configuration worksheet 350  
 bootp daemon 349  
   setting up on AIX 352  
   setting up on OS/2 353  
 bootptab  
   editing the file 350  
 buffer sizes, specifying in TCP/IP profile 245  
 buffer, pool 112  
 BUFNO parameter 41  
 BURST option 57

## C

capture, font 145  
 carrier-strip marking 83  
 carrier-strip, mark form type 83  
 channel paths and switches 6  
 CHARS parameter 41  
 checkpoint 2  
   data 2, 254  
   data set 58, 63  
   deferred-printing mode function 95  
   interval 58, 63, 111  
   JES 254  
   JES3 63  
   microfilm 259  
   microfilm performance 268  
   performance 63  
   PSF 2, 111  
   seconds between 58, 63  
   starting point 58  
   value 58  
 coding the APSXUCOM macro 183  
 COLORMAP parameter 42  
 communication network, establishing 9  
 communication-attached devices (token ring) 13  
 communication-attached printers 10  
 complex image 231  
 complex MO:DCA-P data 231  
 COMPRESS parameter 42  
 COMSETUP parameter 42  
 concatenated  
   libraries 37  
   numbers 219  
   retrieval lists (CRLs) 211, 212, 218  
 configuration considerations for 3130 printer 36  
 configuration considerations for 3160 printer 36  
 configuration considerations for 3935 printer 36  
 CONNINTV parameter 43  
 control block listings, PSF 295  
 control statements for resident fonts 141

Conversion Utility Program, VSE 324  
converting Julian dates 183  
copies of separator pages 80  
copy grouping 117  
copy marking 84  
CRLs (concatenation retrieval lists) 211, 212, 218  
cross-reference, program indicator 319

## D

data integrity 123  
data protection  
    See protecting data  
data separation 241  
data set checkpoint 58  
data set, termination time 212, 213, 218  
DATABUFFERPOOLSIZ statement in TCP/IP  
    profile 245  
DATAACK parameter 43  
dates, converting Julian 183  
default form definition in the PRINTDEV statement 115  
default page definition in the PRINTDEV  
    statement 115  
default, auxiliary page and form definitions 217  
deferred-printing mode  
    PSF startup procedures 40, 67  
    user libraries in 38  
    what to do first 55  
defining communication-attached printers 9  
defining local-attached printers 9  
Defining Printers to OS/390 5  
definition considerations 35  
deleting font mapping  
    raster to outline 154  
deleting resources  
    full control 220  
    partial control 220  
    using PLRs 220  
DEQ or ENQ 199, 212, 218  
DEVICE statement 63  
diagnostic aids  
    trace reports 255  
direct-printing mode  
    PRINTDEV statement 40  
    PSF startup JCL 97  
    using PSF in 95  
disabled mechanisms 86  
DISCINTV parameter 44  
Distributed Print Function (DPF) 9  
Distributed Print Function of PSF/2 80  
    marking resources for 157  
distribution information, separator page 77  
double-byte font 216  
DPAGELBL parameter 44  
DUMP parameter 44

dynamic allocation of printers 55, 96  
dynamic reconfiguration management (DRM) 2  
    3820 channel-attached printer 2, 7  
    AFP1 printers 2, 7  
    channel-attached printer installation 40  
    how to add a 3820 channel-attached printer 7  
    how to add an AFP1 printer 7

## E

ECAMFCS 84  
    carrier-strip field 84  
    example code 173  
    for separator page exits 84  
    mark form carrier strip field 173  
    mark form carrier-strip field 84  
    to disable marking 84  
    to initiate 84  
    where to set this field 84  
    which exits 84  
end-data-set call (EDSC), APSUX16 225  
enhanced N\_UP printing 122  
ENQ or DEQ 199, 212, 218  
Enterprise Systems Connection  
    See ESCON  
error messages, PSF 253  
error-handling for font conversion 328  
ESCON  
    channel-attached printers, sharing 53  
    using channel-attached printers 6  
establishing a communication network 9  
establishing network nodes 9  
estimates, minimum storage 112  
exit processing 169  
exit table, APSUCOM 182  
exit-put routine 186  
exits, PSF  
    for formatting separator pages (APSUX01,  
        APSUX02, APSUX03) 179  
    for message processing (APSUX06) 195  
    for processing logical records, APSUX04 187  
    for resource control (APSUX07) 198  
    for SMF type 6 record processing, APSUX05 193  
    put routine interface, APSUPUT 186

## F

FAILURE parameter 44  
FDEFDD parameter 45  
field descriptions for APSUX07 295  
FLU  
    microfilm utility 265  
font capture 145, 233  
    single-byte raster font 145  
font conversion program (APSRFCF30) 327  
    description 327

- font conversion program (APSRCF30) *(continued)*
  - error handling 328
- Font Fidelity, specifying 129
- Font Library Update utility
  - See FLU
- font mapping
  - adding new 151, 152
  - considerations, outline 209
  - considerations, recovering from font-not-found 209
  - creating new 148
  - disabling 154
  - outline to raster, adding 152
  - raster to outline, adding 151
  - raster to outline, disabling 154
  - tables 150
- font mapping tables 150
- font member names to grid parts (APSRNTGM) 343
- font pruning 232
  - deactivate 232
  - double-byte fonts 304
  - non-3800 double-byte font fonts 304
  - performance considerations 232
  - processing time 232
  - single-byte fonts 304
  - when useful 232
- font resource table 134
- FONT240 parameter 45
- FONT300 parameter 45
- FONTDD parameter 45
- fonts
  - adding outline to raster mapping 152
  - adding raster to outline mapping 151
  - APSRMARK 135
  - calling out 216
  - converting from 240-pel to 300-pel 327
  - creating new 148
  - disabling mapping, raster to outline 154
  - double-byte 216
  - FONT240 parameter on PRINTDEV statement 45
  - FONT300 parameter on PRINTDEV statement 45
  - FONTDD parameter on PRINTDEV statement 45
  - in resource access call 216
  - in resource list control block (RLST) 216
  - libraries 111
  - library 37
  - mapping considerations, outline 209
  - mapping tables 150
  - marking 135
  - MICR 123
  - outline, performance 233
  - printer-resident 138
  - recovery considerations, font-not-found 209
  - resident font utility 135
  - resource management 111
  - resource selection table, deferred environment 120
  - resource selection table, direct printing
    - environment 121

- fonts *(continued)*
  - restriction when substituting 216
  - section ID 219
  - specifying in startup procedures 248
  - supporting resident 133
- FORCE parameter on MODIFY command 252, 254
- form definitions
  - default and auxiliary 217, 218
  - in the resource access call 217
  - in the resource load call 218
  - inline 218
  - modified-default 206
  - resource management 111
  - resource selection table, deferred environment 118
  - resource selection table, direct-printing
    - environment 118
    - user-specified 218
    - with RLST chain 217
- format resolution
  - order 130
  - setting with installation exits 130
  - specifying 129
- formatted line data 231
- FORMDEF parameter 45
- forms flash 116, 117
- forms marking 81
- FSI above the line 3
- FSI below the line 3
- FSSDEF statement 62
- full control of resource deletion 220
- Functional Subsystem (FSS)
  - numbers of in operating system 55
- Functional Subsystem Application (FSA)
  - numbers of in operating system 55
- functional subsystem interface (FSI) 97, 241
  - deferred-printing mode 241
  - how PSF is marked 241
  - performance 241

## G

- GATEWAY statement in TCP/IP profile 246
- GATEWAY statement, in TCP/IP profile 247
- gateway, token-ring 13
- grid parts to font member names (APSRGTM) 340
- GROUP statement 27, 28, 34

## H

- hard page segment 215
- hardware configuration definition (HCD) 5, 244
  - for channel-attached 3800 page printers 7
  - for channel-attached 3820 page printers 7
  - microfilm device 7
  - microfilm printer 7

hardware problem, recovering from 254  
HCD 5  
*See also* hardware configuration definition (HCD)  
header pages 85  
hierarchy for resource selection 115  
  fonts, deferred environment 120  
  fonts, direct-printing environment 121  
  form definitions, deferred environment 118  
  form definitions, direct-printing environment 118  
  interface attachment feature, direct-printing environment 123  
  introduction to 115  
  MICR, direct-printing environment 123  
  page definitions, deferred environment 119  
  page definitions, direct-printing environment 119  
  searching for resource specified for print job 121  
host name, specifying 45

**I**

i-data 7913 IPDS Printer LAN Attachment 347  
  attaching 356  
  checking 348  
  configuration 347  
  configuring 349  
  installing 360  
  operating 356  
  pinging 359  
  setting the controls 354  
  Setting up the controls for the Ethernet LAN model 355  
  shutting down 357  
  starting 356  
  testing 358  
  troubleshooting 360  
I/O subsystem capacity 237  
identifying all output with an overlay 75  
indicator cross-reference, program 319  
initialization call (INIT) 211  
inline resources  
  copy grouping 115, 117  
  integrity for 123  
Input, PAGEDEFs and FORMDEFs 240  
installation exits  
  for formatting separator pages (APSUX01, APSUX02, APSUX03) 179  
  for message processing (APSUX06) 195  
  for processing logical records, APSUX04 187  
  for resource control (APSUX07) 198  
  for SMF type 6 record processing, APSUX05 193  
  put routine interface, APSUPUT 186  
  specifying format resolution for 130  
installation tasks 5  
installation verification program (IVP) 39, 335  
  240-pel resolution printer sample 335  
  example of output 335

installation-exit processing 169  
intermediate device 4  
INTERVAL parameter for keep-alive transmissions 246  
intervention timer, local 44, 254  
IPADDR parameter 45, 248  
IPDS  
  installation 347  
issue an optional message 212  
IVP 335  
  *See also* installation verification program (IVP)

**J**

JCL for startup procedures 73, 249  
JCL parameters  
  description of 111  
  examples of 249  
  FLASH 117  
  for starting PSF 40, 134  
  on the PRINTDEV statement 40, 53  
  PIMSG 114  
  used to generate a long report 142  
  used to generate a short report 142  
  used to mark an MVS member 141  
  used with the APSRMARK utility 137  
JES (Job Entry System)  
  initialization 57  
  JES2 initialization 61  
  JES2 PRINTDEF 58, 61  
  JES3 initialization 62  
JES2 55  
  FSS statement 57  
  initialization statements 56  
  numbers of FSS and FSA 55  
  PRTnnnn statement 57  
JES3 55  
  numbers of FSS and FSA 55  
JES3 initialization statement and the PRINTDEV statement 217  
job separator pages 75  
job-header page 75  
job-trailer page 75  
JOBHDR parameter 45  
jobs formatted at different resolutions, printing 125  
JOBTRLR parameter 45  
Julian dates, converting 183

**K**

keep-alive transmissions 245  
KEEPALIVEOPTIONS statement in TCP/IP profile 245

**L**

least-recently-used algorithm 219

- libraries
  - containing AFP resources 37
  - fonts 37
  - form definitions 37
  - format of 37
  - overlays 37
  - page definitions 37
  - page segments 37
  - working with 37
- library, resources 157, 158
- line mode
  - APSUX04 106
  - enable 106
  - IBM 3800 printers 105
  - limitations, conversion function 108
  - overview 105
  - programming considerations 108
  - programming considerations, Exit 4 106
  - requesting 189
- LINE statement 27, 29, 32, 34
- listings, PSF control block 295
- lists (CRLs), concatenation retrieval 211, 212, 218
- local intervention timer 44
- local major nodes 19
- local-attached printers 9
- local-intervention timer 254
- LOGMODE parameter 45
- logon-mode table 16
- LU statement 24, 30, 33
- LUNAME parameter 46

## M

- macros (ENQ or DEQ) 199, 212, 218
- macros for programming interfaces 337
- magnetic ink character recognition printing 123
- major nodes, local 19
- mapping (fonts)
  - adding new 151, 152
  - considerations, outline 209
  - considerations, recovering from font-not-found 209
  - creating new 148
  - disabling 154
  - outline to raster, adding 152
  - raster to outline, adding 151
  - raster to outline, disabling 154
  - tables 150
- mapping tables 150
- mark form carrier-strip (MFCS) 83
- marking a resource 158
- marking fonts 135
- marking resources 135
- maximum transmission unit in TCP/IP profile 246
- message call, APSUX16 224
- message data set redirection 113

- MESSAGE parameter 46
- message, optionally issued 212
- messages, PSF 253
- MFCS, mark form carrier-strip 83
- MGMTMODE parameter 46
- MICR Postprocessor 123
  - direct-printing environment 123
  - interface attachment feature 123
- microfiche
  - See microfilm
- microfilm
  - microfilm 255
- microfilm device
  - ESCON-attached 6
- MODEENT statement 16
- MTU in TCP/IP profile 246
- MTU, in TCP/IP profile 247
- multiple resolution modes 126
- multipoint telecommunication line 26
- MVS/370
  - numbers of in operating system 55
- MVS/ESA
  - numbers of FSS and FSA 55

## N

- N\_UP printing 121
- NCP
  - major node 25
  - MAXOUT value 236
  - performance 236
- nested resource 203, 220, 221
- network printers, PRINTDEV parameters for 51
- network, communication 9
- nodes, local major 19
- NORESP parameter 47
- NOTIFY JCL keyword 132
  - controlling print complete notification 132
  - creating profiles 132
  - giving access authority 132
- NPRO 131
  - JES2 parameter 59
  - JES3 parameter 65
  - PRINTDEV parameter 47
- number, concatenation 219

## O

- OBJCOND parameter 48
- objects, resource
  - See resource
- operating a TCP/IP-attached printer 251
- optionally issue a message 212
- outline fonts
  - adding mapping of raster 151
  - adding mapping to raster 152



- outline fonts (*continued*)
  - creating new 148
  - disabling mapping of raster 154
  - mapping considerations 209
  - mapping tables 150
  - performance 233
  - recovering from font-not-found considerations 209
- OVERLAY parameter 48
- overlays
  - considerations with APSUX07 203
  - identifying output with 75
  - in the resource access exit 215
  - limits in RRLVs 219
  - maximum supported in direct-printing mode 96
- OVLY240 parameter 48
- OVLY300 parameter 48
- OVLYDD parameter 48

## P

- packet size in TCP/IP profile 246
- page definitions
  - default and auxiliary 217
  - in the resource access call 217
  - in the resource load call 218
  - inline 218
  - modified-default 206
  - resource management 111
  - resource selection table, deferred environment 119
  - resource selection table, direct-printing environment 119
  - user-specified 218
  - with RLST chain 217
- page segments
  - as a nested resource 220
  - defining in an overlay 203
  - deleting auxiliary 220
  - hard 215
  - maximum supported in direct-printing mode 96
  - retaining 200
  - soft 215
  - when RRLV equals 0 220
  - with reload option 216
  - with selective deletion 220
- parameter areas
  - APSLCRL 308
  - APSURLST 311
  - RLSTFEXT 317
  - XTP7 295
- parameters
  - ACQINTV 41
  - APPLID 41
  - BUFNO 41
  - CHARS 41
  - COLORMAP 42
  - COMPRESS 42

- parameters (*continued*)
  - COMSETUP 42
  - CONNINTV 43
  - DATAACK 43
  - DISCINTV 44
  - DPAGELBL 44
  - DPF 163
  - DSHDR 44
  - DUMP 44
  - FAILURE 44
  - FDEFDD 45
  - FONT240 45
  - FONT300 45
  - FONTDD 45
  - FORMDEF 45
  - IPADDR 45
  - JES2 PRINTDEF 58, 61
  - JOBHDR 45
  - JOBTRLR 45
  - LOGMODE 45
  - LUNAME 46
  - MESSAGE 46
  - MGMTMODE 46
  - NORESP 47
  - NPRO 47, 259
  - OBJCOND 48
  - OVERLAY 48
  - OVLY240 48
  - OVLY300 48
  - OVLYDD 48
  - PAGEDEF 48
  - PDEFDD 48
  - PIMSG 49
  - PORTNO 49
  - PRINTDEV 40
  - PRTERORR 49
  - PSEG240 49
  - PSEG300 49
  - PSEGDD 49
  - RELINTV 50
  - RELMODE 50
  - RESPTIME 50
  - SDEFDD 50
  - SETUP 50
  - SFONTDD 50
  - SOVLYDD 50
  - SPAGELBL 50
  - SPSEGDD 50
  - SYSAREA 51
  - TIMEOUT 51
  - TRACE 51
- partial control of resource deletion 219
- PATH statement 24
- paths, channel 6
- PDEFDD parameter 48

- performance factors
  - attachment hardware 234
  - blank compression 238
  - block size 241
  - buffers 239
  - BUFNO parameter 240
  - channel-attached printer 234
  - checkpointing 241
  - checkpointing, microfilm 268
  - coat-tailing 236
  - communication-attached printer 235
  - complex image 231
  - complex MO:DCA-P data 231
  - data checks 239
  - data compression 233
  - data density and complexity 230
  - data rates 235
  - data separation 241
  - data transmission 233
  - Distributed Print Function (DPF) of PSF/2 235
  - DPF 235
  - DPF Resource Saving 235
  - duplex printing 230
  - enhancements and RPQs 230
  - formatted line data 231
  - FSI 241
  - host system 237
  - I/O capacity 237
  - internal processing speed 229
  - local-attached printer 234
  - MAXOUT value, NCP 236
  - microfilm 268
  - NCP MAXOUT value 236
  - outline fonts 233
  - paper size 230
  - print job 230
  - print job characteristics 231
  - print-job size 233
  - printer data checks 239
  - printer features 230
  - printer throughput 229
  - printer-data requirements 239
  - processing capability 237
  - pruning of raster fonts 232
  - PSF Direct 235
  - raster storage 230
  - rate of printing 229
  - real storage 237
  - region sizes 112
  - resident fonts 232
  - resource buffer pool 112
  - resources used 232
  - RU size 235
  - simple image 231
  - simple MO:DCA-P data 231
  - SPBUFFER value 240
- performance factors (*continued*)
  - storage estimates 112
  - system environment 237
  - system load 237
  - system requirements 239
  - system tuning 237
  - tracing 242
  - TRKCELL parameter 241
  - understanding 229
  - unformatted line data 231
  - VTAM DELAY parameter 236
  - VTAM PSNDPAC parameter 236
  - VTAM request unit (RU) size 235
  - PIMSG parameter 49
  - PLR (priority level resource) 220
  - port number, specifying for TCP/IP-attachment 49
  - port switching 52
  - PORTNO parameter 49
  - Postprocessor 123
    - direct-printing environment 123
    - interface attachment feature 123
    - MICR 123
  - PPT entry, default 40
  - prefix mapping 150, 209, 305
  - print-job size 233
  - PRINTDEF, parameter in JES2 58, 61
  - PRINTDEV parameters for network printers 51
  - PRINTDEV statement 40, 53
  - printer sharing in OS/390 51
  - printers
    - disabled mechanisms 86
    - dynamic allocation 55, 96
    - restriction for selective deletions 220
  - printing at multiple resolutions 125
  - printing error messages 114
  - priority level resource (PLR) 220
  - procedures, PSF startup 67, 248
  - profile, TCP/IP 244
  - Program Directory, PSF 5, 67
  - program indicator cross-reference 319
  - program properties table (PPT) entry, default 40
  - programming considerations
    - APSUX04 189
    - APSUX05 194
    - APSUX06 196
    - APSUX07 200
    - APSUX14 222
    - APSUX16 225
  - programming interfaces 337
  - protecting data 131
  - PRTEROR parameter 49
  - pruning of raster fonts 232
  - PSEG240 parameter 49
  - PSEG300 parameter 49
  - PSEGDD parameter 49

- PSF
  - internal processing speed 229
  - performance 229
  - rate of printing 229
  - timers, recommended values 54
- PSF control block listings 295
- PSF Direct function of PSF for AIX 9
- PSF Direct function of PSF/2 9
- PSF errors, responding to 253
- PSF messages 253
- PSF startup procedures 67, 248
- PSF support for resident fonts 133
- PSNDPAC
  - coat-tailing 236
  - pacing value 236
  - VTAM parameter 236
- PU statement 32
- publications, related 389
- put routine interface, APSUPUT 186

**R**

- RACF 131, 132
- raster font, single-byte, capture 145
- raster fonts
  - disabling mapping to outline 154
  - mapping of outline 152
  - mapping tables 150
  - mapping to outline 151
- RDC (resource deletion at DSE) 219
- re-size 327
- real storage 237
- reasonable resource loading values (RRLV) 219, 325
  - coded fonts 325
  - form definitions 325
  - overlays 325
  - page definitions 325
  - page segments 325
- redirecting message data set 113
- region sizes 112
- related publications 389
- RELMODE parameter 50
- Remote PrintManager
  - managing PSF resources 157
  - using APSRMARK for resource identification 157
- resident fonts
  - APSRFTBL table 134
  - APSRMARK utility 135
  - control statements for 141
  - diagnosing problems with 143
  - font pruning 232, 304
  - general parameters 140
  - performance considerations 232
  - specific parameters 138
  - supporting 133
- resolution modes, multiple 126
- resolutions, printing at multiple 125
- resource 111
  - access call (RAC) 213
  - buffers 112
  - definition of 111
  - delete at DSE 219
  - deletion at DSE (RDC) 219
  - exit (APSUX07) 198
  - font pruning 232, 304
  - full control of deletion 220
  - hard page segment 215
  - hierarchy of 115
  - integrity considerations 123
  - libraries for multiple resolution 127
  - library for DPF 157
  - library for PSF 158
  - list control block (RLST) 215, 216
  - load call (RLC) 218
  - management 111
  - marking 135
  - Microfilm setup resource 264
  - nested 203, 220, 221
  - partial control of deletion 219
  - performance considerations 232
  - PLR (priority level resource) 220
  - resident fonts 232
  - selection tables and hierarchy 115
  - soft page segment 215
  - types managed by PSF 111
- Resource access control facility (RACF) 131, 132
- Resource Conversion Utility Program 55
- resource-access call (RAC) 213
  - double-byte font in, 216
  - fonts in, 216
  - form definitions in, 217
  - overlays in, 215
  - page definitions in, 217
  - page segments in, 215
  - restrictions with, 216
- responding to PSF errors 253
- RESPTIME parameter 50
- restartable abends
  - controlling in APSUX07 204
  - understanding 255
- restrictions
  - exception to resource list control block 217
  - for APSUX07 311
  - for printers not supporting selective deletion 220
  - for system macros ENQ or DEQ 218
  - with RAC 216
- retrieval lists (CRLs), concatenation 211, 212, 218
- return codes for font conversion 328
- RLC (resource load call) 218
- RLST (resource list control block) 215, 216

RLSTFEXT parameter area 317  
routines, service 185  
RRLV (reasonable resource loading values) 219, 325

## S

sample jobs, APSRMARK 271  
SDEFDD parameter 50  
search for specified resource 121  
section ID 219  
security  
    See protecting data  
Security Labeling 131  
    how enabled 131  
    protect printed output 131  
selective deletion restriction 220  
SENDGARBAGE parameter for keep-alive  
    transmissions 246  
separating jobs or copies of jobs 75  
    carrier-strip marking 83  
    copy marking 84  
    edge marking 84  
    header pages 85  
separating users' jobs 75  
    blank pages after separator pages 81  
    forms marking 81  
    separator pages 75  
    using MFCS 83  
separator pages 75  
    DPF considerations 80  
    for job header and trailer 75  
    printing multiple copies of 80  
service routines 185  
SERVICE statement 29  
SETUP parameter 50  
sharing printers, methods for 51  
simple image 231  
simple MO:DCA-P data 231  
sizes, region 112  
SMALLDATABUFFERPOOLSIZESIZE statement in TCP/IP  
    profile 245  
SMF type 6 records 86, 193  
soft page segment 215  
SOVLYD parameter 50  
SPAGELBL parameter 50  
specifying format resolution 129  
SPSEGDD parameter 50  
startup procedures, PSF 67, 248  
storage estimates, minimum 112  
supporting resident fonts 133  
switchable resolution mode 126  
switched major node 21  
switches, channel 6  
syntax 14  
syntax for APSXUCOM macro 183

SYSAREA parameter 51  
SYSPARM 78  
    omitted 78  
    value 78  
system environment  
    storage considerations 112  
system environment, changing 39  
system macros 199, 212, 218  
System Management Facility type 6 records 86  
system tuning 237

## T

tables for resource selection 115  
tables, font mapping 150  
TCP/IP profile, modifying 244  
TCP/IP-attached printers  
    connecting 243  
    defining MVS communications control unit 244  
    defining to JES 248  
    defining to PSF 248  
    interface with PSF 251  
    operating 251  
    pinging 246  
    port number for 49  
    software requirements for 243  
    starting 251  
    startup procedure for 248  
    stopping 251  
    TCP/IP profile, modifying 244  
terminating a printer FSA with the MODIFY  
    command 254  
termination call (TERM) 221  
termination time for data set 212  
termination times 199, 218  
TIMEOUT parameter 51  
timer, local intervention 44  
timer, local-intervention 254  
timers, recommended values 54  
TINYDATABUFFERPOOLSIZESIZE statement in TCP/IP  
    profile 245  
Token ring, communication-attached devices 13  
token-ring subsystem connection 34  
trace description 255  
TRACE parameter 51  
tracing 242  
tuning, system 237

## U

unformatted line data 231  
user job control language (JCL) parameters  
    See JCL parameters  
user libraries 38  
Utilities  
    FLU 265

Utilities (*continued*)

- Font Library Update, microfilm 265
- microfilm FLU 265
- microfilm utility 265
- utility 265

utility

- microfilm utility 265

Utility Program, VSE 324

## V

VBUILD statement 22

VSE Resource Conversion 324

VTAM (Virtual Telecommunications Access Method)

- 37X5 communication controller 40
- application-program node name 41
- coat-tailing algorithm 236
- communication controller 19, 22
- communication lines 26
- communication-attached devices 11
- defining bits in the station ID 23
- defining local major nodes 19
- defining NCP major nodes 19
- definitions for PSF network nodes 11
- DELAY parameter 236
- dialing the phone 27
- disconnecting the physical unit 23
- DLOGMODE 35
- establishing a communication network 9
- establishing connections 34
- IDBLK value 23
- IDNUM value 23
- IOBUF buffer pool 20
- local-attached devices 19
- logical unit 33
- logical unit (LU) 14
- LOGMODE 35
- logon-mode table entry 16
- logon-mode table entry name 20, 24, 33, 45
- logon-mode table name 30
- MODEENT statement 16
- NCP generation 11
- NCP major node definition 25
- network configurations 26
- network considerations 35
- network definition 45, 46
- OPNDST command 36
- pacing the flow of data 21, 25
- pacing value 236
- performance 236
- PSNDPAC parameter 236
- PSNDPAC value 17
- RU size 20, 235
- RUSIZES value 17
- SIMLOGON command 36
- SNA 11

VTAM (Virtual Telecommunications Access Method)  
(*continued*)

- SNA set up for ACF/VTAM 39
- SNA set up for VTAM/NCP 39
- SRCVPAC value 17
- syntax rules 14
- when line is active 27
- XID exchange 23

## X

XTP7 parameter area 295

XTP7nnnn field descriptions

See APSUX07 exit

---

# Communicating Your Comments to IBM

Print Services Facility for OS/390  
Customization  
Version 3, Release 1.0  
Publication No. S544-5622-00

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM. Whichever method you choose, make sure you send your name, address, and telephone number if you would like a reply.

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

If you are mailing a readers' comment form (RCF) from a country other than the United States, you can give the RCF to the local IBM branch office or IBM representative for postage-paid mailing.

- If you prefer to send comments by mail, use the RCF at the back of this book.
- If you prefer to send comments by FAX, use this number:  
1-800-524-1519
- If you prefer to send comments electronically, use this network ID:  
printpub@us.ibm.com

Make sure to include the following in your note:

- Title and publication number of this book
- Page number or topic to which your comment applies.

---

# Readers' Comments — We'd Like to Hear from You

Print Services Facility for OS/390  
Customization  
Version 3, Release 1.0  
Publication No. S544-5622-00

Overall, how satisfied are you with the information in this book?

|                      | Very Satisfied           | Satisfied                | Neutral                  | Dissatisfied             | Very Dissatisfied        |
|----------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| Overall satisfaction | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

How satisfied are you that the information in this book is:

|                          | Very Satisfied           | Satisfied                | Neutral                  | Dissatisfied             | Very Dissatisfied        |
|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| Accurate                 | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Complete                 | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Easy to find             | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Easy to understand       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Well organized           | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Applicable to your tasks | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

Please tell us how we can improve this book:

Thank you for your responses. May we contact you?  Yes  No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

\_\_\_\_\_  
Name

\_\_\_\_\_  
Address

\_\_\_\_\_  
Company or Organization

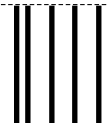
\_\_\_\_\_  
Phone No.



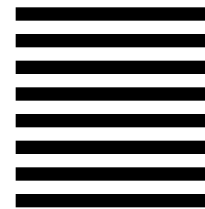
Fold and Tape

Please do not staple

Fold and Tape



NO POSTAGE  
NECESSARY  
IF MAILED IN THE  
UNITED STATES



# BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Corporation  
Information Development  
IBM Printing Systems Company  
Department H7FE Building 003G  
Boulder, CO 80301-9817



Fold and Tape

Please do not staple

Fold and Tape







Program Number: 5655-B17



Printed in the United States of America  
on recycled paper containing 10%  
recovered post-consumer fiber.

S544-5622-00



*Spine information:*



Print Services Facility for OS/390

Customization

*Version 3, Release 1.0*