

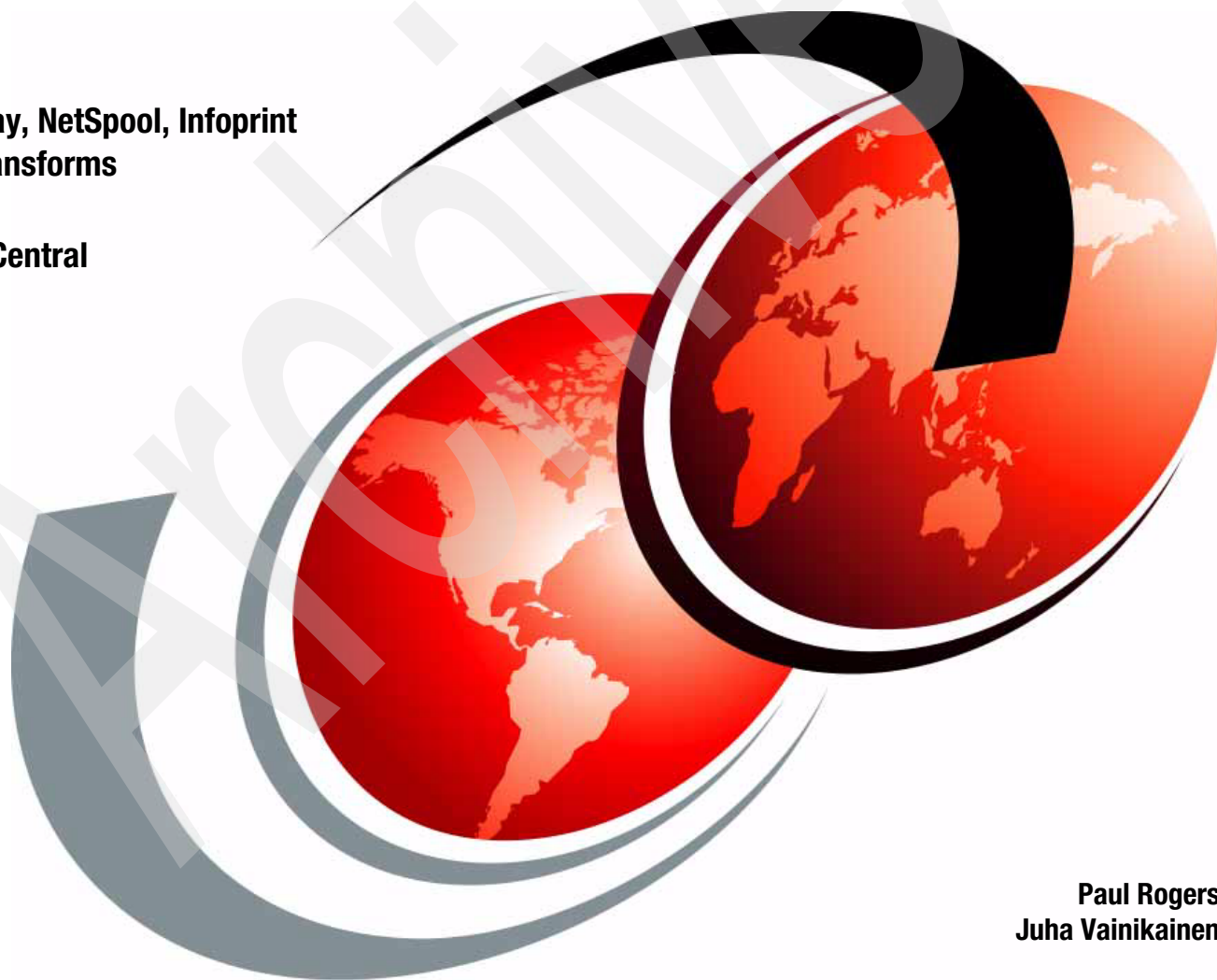
# ABCs of z/OS System Programming

## Volume 7

Infoprint Server

IP PrintWay, NetSpool, Infoprint  
Server Transforms

Infoprint Central



Paul Rogers  
Juha Vainikainen

**Redbooks**





International Technical Support Organization

**ABCs of z/OS System Programming  
Volume 7**

October 2008

Archived

**Note:** Before using this information and the product it supports, read the information in “Notices” on page ix.

## **Second Edition (October 2008)**

This edition applies to Version 1 Release 9 of z/OS (5637-A01), and Version 1, Release 8 of z/OS.e (5655-G52), and to all subsequent releases and modifications until otherwise indicated in new editions.

© Copyright International Business Machines Corporation 2006, 2008. All rights reserved.

Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

# Contents

<b>Notices</b> .....	ix
Trademarks .....	x
 <b>Preface</b> .....	 xi
The team that wrote this book .....	xii
Become a published author .....	xii
Comments welcome .....	xii
 <b>Chapter 1. Infoprint Server</b> .....	 1
1.1 z/OS Infoprint Server .....	3
1.2 Data streams on JES spool .....	5
1.3 Infoprint Server components .....	7
1.4 Print Interface .....	9
1.5 LPR/LPD Protocol - RFC1179 .....	12
1.6 IP PrintWay basic mode support .....	13
1.7 IP PrintWay extended mode support .....	15
1.8 Printer Inventory Manager .....	17
1.9 Infoprint Server Transforms .....	20
1.10 NetSpool .....	22
1.11 SNMP Overview .....	24
1.12 Print work submission to Infoprint Server .....	25
1.13 Printing using LPR command .....	28
1.14 z/OS UNIX shell printing commands .....	29
1.15 Using Internet Printing Protocol (IPP) .....	31
1.16 Printing with Infoprint Port Monitor for Windows .....	32
1.17 Printing SAP R/3 documents .....	34
1.18 Infoprint Central overview .....	35
 <b>Chapter 2. UNIX System Services overview</b> .....	 37
2.1 RACF OMVS segments .....	39
2.2 Infoprint Server and RACF OMVS segments .....	41
2.3 z/OS UNIX users and groups .....	42
2.4 RACF commands to define groups .....	44
2.5 RACF commands to define users .....	45
2.6 Superusers with appropriate privileges .....	47
2.7 BPX.SUPERUSER authority .....	49
2.8 SUPERUSER.FILESYS profile .....	50
2.9 Assigning UIDs .....	52
2.10 Shared UID prevention .....	54
2.11 Initializing z/OS UNIX .....	55
2.12 z/OS UNIX File Security Packet .....	56
2.13 z/OS UNIX interactive interfaces .....	58
2.14 ISPF Option 6 .....	60
2.15 ISHELL ISPF dialog (ISH) .....	61
2.16 Files and directories in /var/Printsrv .....	62
2.17 OMVS command shell session .....	63
2.18 Environment variables .....	64
 <b>Chapter 3. Infoprint Server customization</b> .....	 65

3.1	Enabling the Infoprint Server. . . . .	67
3.2	Basics of the z/OS UNIX file system. . . . .	69
3.3	z/OS UNIX files and Infoprint Server. . . . .	71
3.4	Infoprint Server directories/files. . . . .	73
3.5	Infoprint Server configuration file (aopd.conf). . . . .	74
3.6	Full format of the aopd.conf file. . . . .	77
3.7	Infoprint Server daemons . . . . .	79
3.8	Customizing the Printer Inventory Manager . . . . .	81
3.9	Functions using Printer Inventory . . . . .	83
3.10	Printer Inventory directories and files . . . . .	85
3.11	Printer Inventory and security . . . . .	88
3.12	Create users for administration and operation . . . . .	89
3.13	SYS1.SAMPLIB(AOPRACF) commands . . . . .	91
3.14	Create a user file system . . . . .	93
3.15	Starting Printer Inventory Manager. . . . .	95
3.16	AOPSTART JCL procedure . . . . .	97
3.17	Environment variables in /etc/profile . . . . .	99
3.18	Additional environment variables in /etc/profile. . . . .	101
3.19	Editing aopstart rexx exec. . . . .	103
3.20	Stopping Infoprint Server daemons. . . . .	105
3.21	AOPSTOP JCL procedure . . . . .	107
3.22	Infoprint Server address spaces . . . . .	109
3.23	Working with Printer Inventory Manager (AOPD) . . . . .	111
3.24	Enabling Infoprint Server ISPF panels . . . . .	112
3.25	Infoprint Server configuration panel . . . . .	114
3.26	An alternative REXX EXEC - Start ISPF application. . . . .	116
3.27	Infoprint Server ISPF primary panel . . . . .	117
3.28	Add a printer using ISPF panels . . . . .	118
3.29	Add a printer using ISPF panels . . . . .	120
3.30	Component name list . . . . .	122
3.31	Component definitions . . . . .	124
3.32	Specifying printer definition attributes . . . . .	125
3.33	Components for IP PrintWay printer definitions . . . . .	127
3.34	Add a printer using ISPF panels - Allocation(1/3) . . . . .	129
3.35	Add a printer using ISPF panels - Allocation(2/3) . . . . .	131
3.36	Add a printer using ISPF panels - Allocation(3/3) . . . . .	133
3.37	Add a printer using ISPF panels - Processing (1/3) . . . . .	134
3.38	Add a printer using ISPF panels - Processing(2/3) . . . . .	136
3.39	Add a printer using ISPF panels - Processing(3/3) . . . . .	138
3.40	Add a printer using ISPF panels - IP PrintWay Options (1/2) . . . . .	139
3.41	Add a printer using ISPF panels - IP PrintWay Options(2/2). . . . .	141
3.42	Add a printer using ISPF panels - LPR Protocol . . . . .	142
3.43	Printer Inventory Definition Utility (pidu) . . . . .	144
3.44	pidu commands. . . . .	146
3.45	pidu -c "display printer pokeps ; " . . . . .	148
3.46	Backing up and restoring Printer Inventory . . . . .	149
3.47	Setting Workload Manager goals . . . . .	151
<b>Chapter 4.</b>	<b>Print Interface . . . . .</b>	<b>153</b>
4.1	Print Interface software requirements . . . . .	155
4.2	Functions provided by Print Interface . . . . .	156
4.3	LPR/LPD protocol - RFC 1179 . . . . .	157
4.4	Print Interface customization. . . . .	159

4.5	Print Interface - aoplpd daemon (LPR/LPD)	162
4.6	Print Interface validating requests	164
4.7	Print Interface and z/OS UNIX printing commands	165
4.8	Print requests from z/OS to Print Interface	166
4.9	Print Interface processing for print requests	167
4.10	Printing commands customization	169
4.11	Using attribute files	171
4.12	RACF OMVS segment and aopd.conf file updates	173
4.13	Customizing the AOPPRINT JCL procedure	176
4.14	AOPPRINT examples	178
4.15	Print Interface and IPP protocol	179
4.16	Customization tasks for Print Interface IPP server	181
4.17	AFP authorization for Java libraries	183
4.18	Customizing z/OS Communication Server (TCP/IP)	185
4.19	Other hlq.PROFILE.TCPIP and JES considerations	188
4.20	Print Interface subsystem customization	190
4.21	Transform configuration file	192
4.22	Infoprint Server Transform Manager and transforms	194
4.23	Infoprint Server transforms	196
4.24	Customizing the Transform Manager	198
4.25	Transform classes	201
4.26	Format of a PCL to AFP transform entry	203
4.27	PostScript and PDF to AFP transform	205
4.28	Security for the PostScript and PDF to AFP transform	207
4.29	AFP to PCL, AFP to PDF, and AFP to PostScript transforms	209
4.30	AFP to PCL, AFP to PostScript, or AFP to PDF transform entry	212
4.31	AFP to PCL, AFP to PostScript, or AFP to PDF environment variables	214
4.32	Customization of AFP resources	216
4.33	Specify AFP resource libraries	219
4.34	Mapping AFP raster and AFP outline fonts	221
4.35	Optional environment variables	223
4.36	Example afp2pcl transform class entry	225
4.37	Enhanced PDF documents	226
4.38	To use PDF encryption	228
4.39	Example - Create an encrypted PDF document	231
4.40	Scaling fonts using the AOXCF30 program	232
<b>Chapter 5.</b>	<b>IP PrintWay</b>	<b>235</b>
5.1	IP PrintWay	236
5.2	IP PrintWay modes of operation	239
5.3	SAPI process	241
5.4	SAPI client request to JES	243
5.5	IP PrintWay extended mode SAPI processing	245
5.6	IP PrintWay extended mode printing	246
5.7	IP PrintWay basic mode	247
5.8	IP PrintWay basic mode FSS/FSA	249
5.9	IP PrintWay basic mode printing	252
5.10	One IP PrintWay basic mode FSS/JES and several FSAs	253
5.11	Multiple IP PrintWay basic mode FSS address spaces	254
5.12	TCP/IP customization tasks for IP PrintWay basic mode	255
5.13	IP PrintWay basic mode customization steps	258
5.14	ANFWPROC - IP PrintWay basic mode startup procedure	261
5.15	RACF for IP PrintWay basic mode startup procedure	262

5.16	IP PrintWay transmission-queue and message-log	264
5.17	View IP Printway basic mode transmission-queue	266
5.18	IP PrintWay Transmission Queue Selection panel	267
5.19	IP PrintWay basic mode message log view	268
5.20	VTAM APPL resource definition and LE run-time	269
5.21	Define IP PrintWay basic mode functional subsystem	271
5.22	IP PrintWay basic mode default font and page definitions	273
5.23	JES FSS/FSA definitions for IP PrintWay basic mode	275
5.24	Infoprint Server extended mode versus basic mode	277
5.25	IP PrintWay extended mode enhancements	278
5.26	Enhanced functions with extended mode	279
5.27	IP PrintWay extended mode customization	281
5.28	Job selection rules - IP PrintWay extended mode	284
5.29	Create IP PrintWay extended mode job selection rules	286
5.30	Add IP PrintWay extended mode job selection rule	287
5.31	Job selection rules in the Printer Inventory	288
5.32	Job selection rule security profiles	289
5.33	Authorizing JES spool and SMF for Infoprint Server	290
5.34	Customizing z/OS UNIX sendmail	292
<b>Chapter 6. NetSpool</b>		295
6.1	NetSpool overview	297
6.2	NetSpool features	299
6.3	NetSpool owner definitions	301
6.4	Embedding job attributes in the NetSpool print data	302
6.5	Customizing NetSpool	304
6.6	Customizing the NetSpool startup procedure	306
6.7	Defining NetSpool user ID and LUs	308
6.8	Defining NetSpool printer LUs to VTAM	310
6.9	Defining NetSpool printers	313
6.10	Defining printer pools	315
6.11	Writing NetSpool exits	317
6.12	Starting and stopping NetSpool	319
6.13	NetSpool operator commands	321
<b>Chapter 7. Infoprint Central</b>		325
7.1	Infoprint Central and Infoprint Server	327
7.2	Infoprint Central	328
7.3	Customizing the z/OS HTTP Server configuration file	330
7.4	Protecting Infoprint Central Web pages	332
7.5	z/OS HTTP Server environment variables file	333
7.6	z/OS HTTP Server JCL procedure	337
7.7	Customization tasks for Infoprint Central	338
7.8	Customizing aopd.conf file and RACF for Infoprint Central	340
7.9	Infoprint Central Security - Printer Inventory	342
7.10	Infoprint Central Security - Protecting IP PrintWay printers	343
7.11	Infoprint Central and operator command security	344
7.12	Infoprint Central and NetSpool security	346
7.13	Infoprint Central and job selection rules	347
7.14	Infoprint Central security checks	348
7.15	Logging on to Infoprint Central from the Web	350
7.16	Infoprint Central action access	352
7.17	PIDU command to create printer security profiles	354



7.18	Work with Printers panel	355
7.19	Working with printers.	357
7.20	IP PrintWay Information window	359
7.21	Ping a printer.	361
7.22	Redirect a printer.	363
7.23	Using the stop button	365
7.24	Stop a printer options	366
7.25	View log for a printer	367
7.26	Viewing printer log.	368
7.27	Infoprint Server messages in the OPERLOG	369
7.28	Attributes for the message configuration file	371
7.29	Infoprint Central work with Print Jobs	372
7.30	Infoprint Server print jobs	374
7.31	Print job information	376
7.32	Change print output priority.	377
7.33	Work with NetSpool Logical Units	378
7.34	System Status - System Daemons and Tasks	379
7.35	System Status - IP PrintWay Job Selection Rules panel	380
7.36	Find Printer Definitions panel	381
7.37	Printer Definitions	383
<b>Chapter 8. User interfaces to Infoprint Server</b>		<b>385</b>
8.1	Printing from z/OS UNIX System Services	386
8.2	z/OS UNIX user prints a file	387
8.3	lpstat command options	389
8.4	z/OS UNIX transform commands	390
8.5	Infoprint Server and transforms.	392
8.6	Infoprint Server job attributes	393
8.7	Job attributes and JCL equivalents.	394
8.8	Using job attributes	395
8.9	Job attributes and NetSpool	396
8.10	Printing from batch applications using IP PrintWay	397
8.11	OUTPUT JCL for IP PrintWay.	398
8.12	Infoprint Server sendmail support	400
8.13	Processing options for e-mail printers.	401
8.14	Infoprint Server ISPF e-mail protocol panel	402
8.15	Flow for batch job output.	403
8.16	AOPBATCH Program for Data Transforms.	404
8.17	AOPPRINT JCL procedure	406
8.18	Creating an alias file	407
8.19	Infoprint Central e-mail display	408
8.20	Printing from workstations overview	409
8.21	Adding a z/OS printer to a Windows environment(1 of 5)	411
8.22	Adding a z/OS printer to a Windows environment (2 of 5)	413
8.23	Adding a z/OS printer to a Windows environment (3 of 5)	414
8.24	Adding a z/OS printer to a Windows environment (4 of 5)	415
8.25	Adding a z/OS printer to a Windows environment (5 of 5)	416
8.26	Printing from Windows with the z/OS printer.	417
8.27	Using LPR command to submit print requests	418
8.28	z/OS LPR command syntax	419
8.29	SDSF view of IP PrintWay data sets.	421
8.30	JES2 filter to display PS output.	422
8.31	Viewing Infoprint Server messages.	423

8.32 Viewing messages with aoplogu command .....	425
8.33 Format of common message log messages .....	427
8.34 Viewing IP PrintWay basic mode messages .....	428
8.35 IP PrintWay and the JSPA .....	429
<b>Related publications</b> .....	431
IBM Redbooks documents .....	431
Other publications .....	431
Online resources .....	431
How to get IBM Redbooks publications .....	432
Help from IBM .....	432

# Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.*

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:** INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.


## COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

# Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AIX®	Language Environment®	Redbooks (logo)  ®
AS/400®	MVS™	REXX™
CICS®	MVS/ESA™	RMF™
CUA®	NetSpool™	S/370™
DFSMS™	OS/390®	VTAM®
GDPS®	OS/400®	z/Architecture®
Geographically Dispersed Parallel Sysplex™	Parallel Sysplex®	z/OS®
IBM®	Print Services Facility™	z/VM®
IMS™	PrintWay™	zSeries®
IP PrintWay™	RACF®	
	Redbooks®	

The following terms are trademarks of other companies:

Acrobat, Adobe, PostScript, and Portable Document Format (PDF) are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

ABAP, SAP R/3, SAP, and SAP logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries.

Java, Solaris, SunOS, ZFS, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Internet Explorer, Microsoft, Windows NT, Windows Server, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

# Preface

The ABCs of z/OS® System Programming is an eleven volume collection that provides an introduction to the z/OS operating system and the hardware architecture. Whether you are a beginner or an experienced system programmer, the ABCs collection provides the information that you need to start your research into z/OS and related subjects. If you would like to become more familiar with z/OS in your current environment, or if you are evaluating platforms to consolidate your e-business applications, the ABCs collection will serve as a powerful technical tool.

This IBM® Redbooks® publication describes the functions of the Infoprint Server. It will help you install, tailor, configure, and use the z/OS Version 1 Release 7 version of Infoprint Server. Topics covered in this volume are the following:

- ▶ Chapter 1. Infoprint Server
- ▶ Chapter 2. UNIX® System Services overview
- ▶ Chapter 3. Infoprint Server customization
- ▶ Chapter 4. Print Interface
- ▶ Chapter 5. IP PrintWay™
- ▶ Chapter 6. NetSpool™
- ▶ Chapter 7. Infoprint Central
- ▶ Chapter 8. User interfaces to Infoprint Server

The contents of the volumes are as follows:

- ▶ Volume 1: Introduction to z/OS and storage concepts, TSO/E, ISPF, JCL, SDSF, and z/OS delivery and installation
- ▶ Volume 2: z/OS implementation and daily maintenance, defining subsystems, JES2 and JES3, LPA, LNKST, authorized libraries, SMP/E, Language Environment®
- ▶ Volume 3: Introduction to DFSMS™, data set basics storage management hardware and software, catalogs, and DFSMSHVS
- ▶ Volume 4: Communication Server, TCP/IP, and VTAM®
- ▶ Volume 5: Base and Parallel Sysplex®, System Logger, Resource Recovery Services (RRS), global resource serialization (GRS), z/OS system operations, automatic restart management (ARM), Geographically Dispersed Parallel Sysplex™ (GDPS®)
- ▶ Volume 6: Introduction to security, RACF®, Digital certificates and PKI, Kerberos, cryptography, zSeries® firewall technologies, LDAP, and Enterprise identity mapping (EIM)
- ▶ Volume 7: Printing in a z/OS environment, Infoprint Server and Infoprint Central
- ▶ Volume 8: An introduction to z/OS problem diagnosis
- ▶ Volume 9: z/OS UNIX System Services
- ▶ Volume 10: Introduction to z/Architecture®, zSeries processor design, zSeries connectivity, LPAR concepts, HCD, and HMC
- ▶ Volume 11: Capacity planning, performance management, WLM, RMF™, and SMF

## The team that wrote this book

This book was produced by a team of specialists from around the world working at the International Technical Support Organization, Poughkeepsie Center.

**Paul Rogers** is a Consulting IT Specialist at the International Technical Support Organization, Poughkeepsie Center and has worked in IBM for 41 years. He writes extensively and teaches IBM classes worldwide on various aspects of z/OS JES3, Infoprint Server, and z/OS UNIX. Before joining the ITSO 20 years ago, Paul worked in the IBM Installation Support Center (ISC) in Greenford, England, providing OS/390® and JES support for IBM EMEA, and in the Washington Systems Center in Gaithersburg, Maryland.

**Juha Vainikainen** was a Senior I/T Specialist in IBM Finland until his recent retirement. He has extensive experience with all aspects of OS/390 and z/OS and all related subsystems, and he has worked with Infoprint Server since OS/390 Release 5. Juha continues to share his vast knowledge about these topics as a consultant for various ITSO documentation projects.

## Become a published author

Join us for a two- to six-week residency program! Help write a Redbooks publication dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners and/or customers.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

[ibm.com/redbooks/residencies.html](http://ibm.com/redbooks/residencies.html)

## Comments welcome

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this or other Redbooks publications in one of the following ways:

- Use the online **Contact us** review book form found at:

[ibm.com/redbooks](http://ibm.com/redbooks)

- Send your comments in an email to:

[redbook@us.ibm.com](mailto:redbook@us.ibm.com)

- Mail your comments to:

IBM Corporation, International Technical Support Organization  
Dept. HYTD Mail Station P099  
2455 South Road  
Poughkeepsie, NY 12601-5400

# Infoprint Server

Infoprint Server is an optional feature of z/OS that uses z/OS UNIX System Services. This feature is the basis for a total print serving solution for the z/OS environment. It lets you consolidate your print workload from many servers onto a central z/OS print server.

Infoprint Server delivers improved efficiency and lower overall printing cost with the flexibility for high-volume, high-speed printing from anywhere in the network. With Infoprint Server, you can reduce the overall cost of printing while improving manageability, data retrievability, and usability.

Some highlights of Infoprint Server are:

- ▶ **IP PrintWay** can give you fast access to TCP/IP-connected printers and to Virtual Telecommunications Access Method (VTAM)-controlled printers.
- ▶ **NetSpool** automatically directs VTAM application data to the job entry subsystem (JES) spool without requiring application changes.
- ▶ **Print Interface** receives print requests from clients that run on remote systems, such as Windows® and UNIX systems, and directs data to the JES spool. It accepts data in a variety of formats, including: Advanced Function Presentation (AFP), plain text, Printer Control Language (PCL), Portable Document Format (PDF), and PostScript® formats. It also provides commands that let you print from z/OS UNIX System Services.
- ▶ **Infoprint Central** lets help desk operators and other authorized users or job submitters work with print jobs, printers, and NetSpool logical units (LUs); display printer definitions; and check system status. Infoprint Central is a Web-based print management system.
- ▶ **Infoprint Server Transforms** for z/OS provide a set of data transforms that let you convert data to and from the AFP data format. Some transforms (provided at no additional cost to Infoprint Server customers) let you print PCL, PDF, PostScript, and SAP® data on IBM AFP printers. Other transforms (available as priced features) let you print AFP and line data on PCL and PostScript printers, convert AFP data to PDF format for posting on the Web, and print line data on VTAM-controlled printers.

In addition:

- IBM Infoprint XML Extender for z/OS, a separate product (5655-J66), lets you transform Extensible Markup Language (XML) files to AFP or PDF format for printing or e-mailing.

- IBM Infoprint XT Extender for z/OS, a separate product (5655-J65), lets you transform Xerox files to AFP format for printing or e-mailing. The Xerox files can be line-conditioned data streams (LCDS) or metacode data streams. XT is the IBM Xerox Transform technology.

An administrator can set up the transforms to automatically convert data when printing. A user can transform documents to and from the AFP data format using the z/OS UNIX System Services. The transformed documents can be saved in the converted format and printed later or be sent to other users.

Examples of printing situations in which you can use Infoprint Server:

- ▶ Printing CICS® or IMS™ output data on network printers (ASCII printers in a TCP/IP network and VTAM-controlled printers in an SNA network)
- ▶ Printing UNIX data on AFP printers
- ▶ Printing Web documents data using the Windows SMB printing protocol
- ▶ Printing Web documents using the Infoprint Port Monitor for Windows
- ▶ Printing documents over the Internet using the Internet Printing Protocol (IPP)
- ▶ Printing data from batch applications to network printers
- ▶ Printing AFP data from batch applications on LAN-based Printer Control Language (PCL) printers and to e-mail destinations
- ▶ Printing AFP documents from Windows environments
- ▶ Printing SAP R/3® Output Text Format (OTF) and Advanced Business Application Programming (ABAP™) data stream files on AFP printers
- ▶ Monitoring printer status



## 1.1 z/OS Infoprint Server

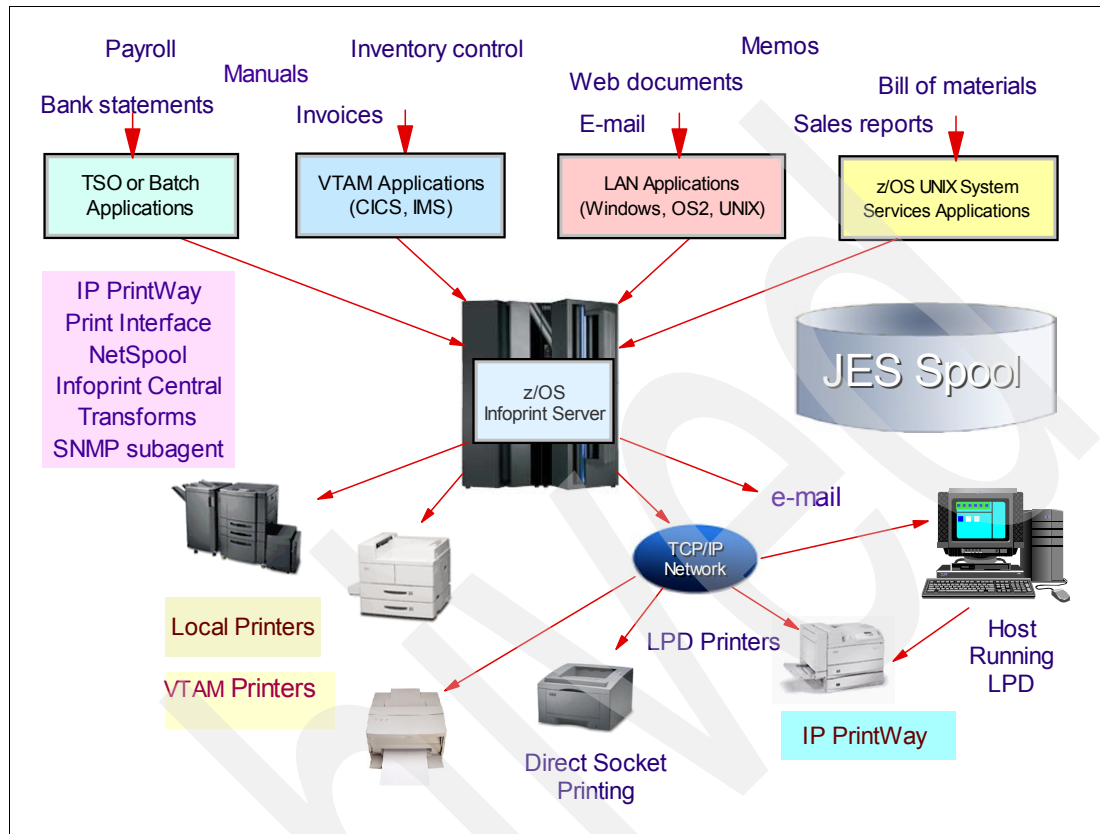


Figure 1-1 z/OS Infoprint Server overview

### Infoprint Server overview

Infoprint Server is an optional feature of z/OS. Infoprint Server is a UNIX application that uses z/OS UNIX System Services. This feature is the basis for a total print serving solution for the z/OS environment in a TCP/IP network. Infoprint Server lets users submit print requests from remote workstations in a TCP/IP network, from UNIX System Services applications, from batch applications, and from VTAM applications, such as CICS or IMS applications. It allows you to consolidate your print workload from the servers onto a central z/OS print server.

Infoprint Server delivers improved efficiency and lower overall printing cost with the flexibility for high-volume, high-speed printing from anywhere in the network. With Infoprint Server, you can reduce the overall cost of printing while improving manageability, data retrievability, and usability.

#### IP PrintWay

IP PrintWay is the component of Infoprint Server that transmits output data sets from the JES2 or JES3 spool to network printers, or to other host systems in your TCP/IP network. The remote printer or host system must support either the LPR/LPD protocol, the IPP protocol, or direct socket printing. IP PrintWay can give you fast access to TCP/IP-connected printers and to Virtual Telecommunications Access Method (VTAM)-controlled printers.

#### Print Interface

Print Interface is the component of Infoprint Server that processes print requests received from both remote clients and local users. When the

Print Interface receives a print request, it allocates an output data set on the JES spool.

#### **NetSpool**

NetSpool intercepts print data from VTAM applications, such as CICS and IMS; transforms the data streams to EBCDIC line data, PCL, PDF, or other formats that the target printer accepts; and writes the output data set to the JES spool. JES or PSF can print the output data sets or JES can transmit them to other locations for printing. IP PrintWay can transmit the output data sets to remote printers in your TCP/IP network.

#### **Infoprint Central**

Infoprint Central is a Web-based print management system primarily for help desk operators. However, other authorized users or job submitters can also use it. Infoprint Central works with IP PrintWay extended mode. With Infoprint Central, you can:

- ▶Work with print jobs.
- ▶Work with printers.
- ▶Work with NetSpool logical units (LUs).
- ▶Display printer definitions.
- ▶Check system status.

#### **Transforms**

Transform Interface communicates with transforms that IBM transform products provide. Transform Interface manages transforms that run on the z/OS system. In addition, it can send data to remote systems, such as Linux®, to be transformed. The transforms are all separately priced products. The available transforms and order numbers are:

*Infoprint Transforms to AFP for z/OS, 5655-N60*

*Infoprint Transform for AFP to HP PCL for z/OS, 5655-P19*

*Infoprint Transform for AFP to Adobe® PDF for z/OS, 5655-P20*

*Infoprint Transform for AFP to Adobe PostScript for z/OS, 5655-P21*

*Infoprint Coaxial Printer Support for z/OS, 5655-N62*

*Infoprint XML Extender for z/OS, 5655-J66*

*Infoprint XT Extender for z/OS, 5655-J65*

*Infoprint Transform Manager for Linux, 5639-P51*

#### **SNMP subagent**

The Infoprint Server Simple Network Management Protocol (SNMP) subagent, with support that PSF and the z/OS SNMP agent provide, lets administrators monitor printer characteristics (such as the printer resolution) and printer status (such as paper jams) for any printer that PSF controls. Also, administrators can be notified as soon as an intervention situation (such as a paper jam) occurs on the printer. This support does not let administrators change any printer characteristics.

## 1.2 Data streams on JES spool

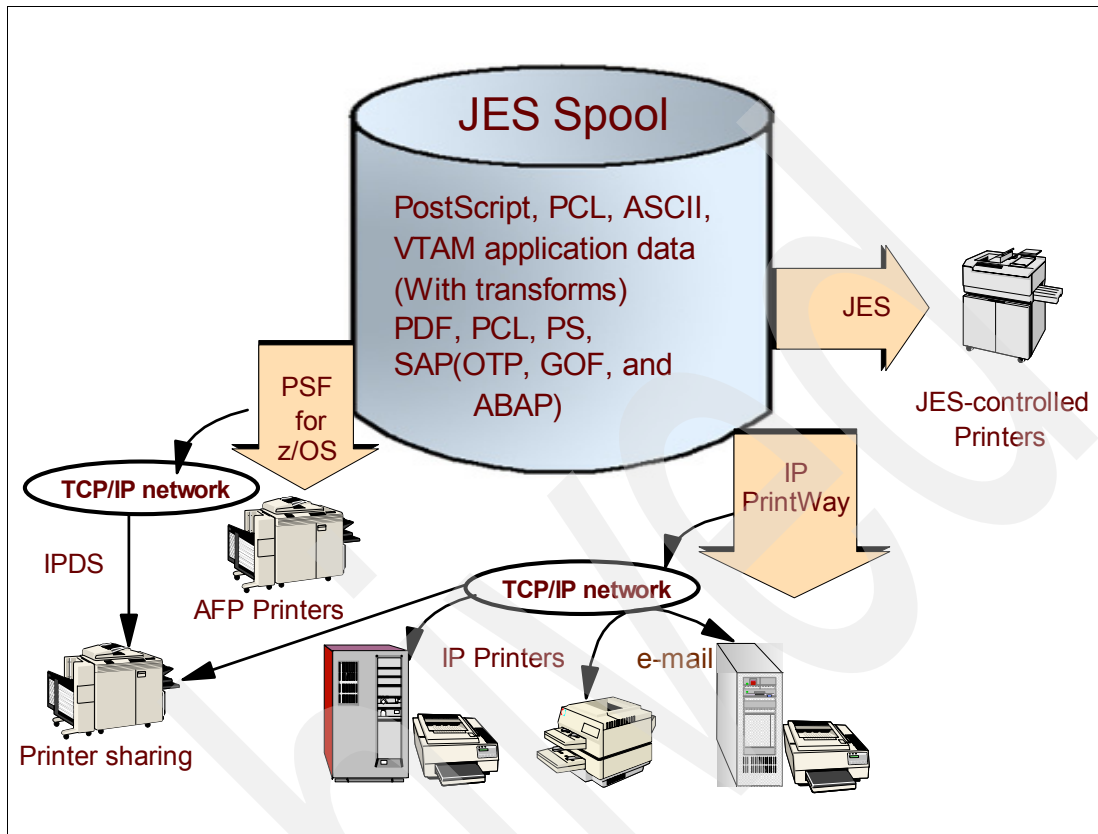


Figure 1-2 Data streams supported on the JES spool

### JES spool data

Infoprint Server provides support for the most commonly used printer languages in the industry, including Postscript, PCL, and AFP or Mixed Object Document Content Architecture-Presentation (MO:DCA-P). With the Infoprint Server Transforms product, Infoprint Server supports also printing PDF, SAP OTF, and SAP ABAP data streams on IBM AFP printers and printing line and AFP data streams on PCL and PostScript printers.

### IP PrintWay

The IP PrintWay component of Infoprint Server transmits output data sets from the JES spool to remote printers or print servers and to e-mail destinations. You can run either IP PrintWay basic mode or IP PrintWay extended mode:

- ▶ IP PrintWay basic mode, the original mode of operation, uses the z/OS Functional Subsystem Interface (FSI) to obtain output data sets from the JES spool. IBM does not plan additional enhancements to IP PrintWay basic mode.
- ▶ IP PrintWay extended mode uses the z/OS Sysout Application Programming Interface (SAPI) to obtain output data sets from the JES spool. It provides better performance, improved usability, and more function than IP PrintWay basic mode.

## **JES spool print streams**

The following print data streams are supported from the JES spool:

- ▶ PostScript (PS)
- ▶ Printer Control Language (PCL)
- ▶ American National Standard Code for Information Interchange (ASCII)
- ▶ Data from VTAM applications (such as CICS and IMS)
- ▶ Portable Document Format (PDF)
- ▶ SAP Output Text Format (OTF)
- ▶ SAP Advanced Business Application Programming (ABAP)
- ▶ eXtensible Markup Language (XML)

IP PrintWay can, in most cases, automatically detect the input data format and transform data into the format required by the printer or e-mail destination. It uses information in printer definitions in the Printer Inventory to process data sets, select the transmission protocol, and obtain printer addresses.

IP PrintWay, in many cases, automatically detects the format of the data stream in print jobs. It can then invoke the transforms that the selected printer needs to print the data stream.

## 1.3 Infoprint Server components

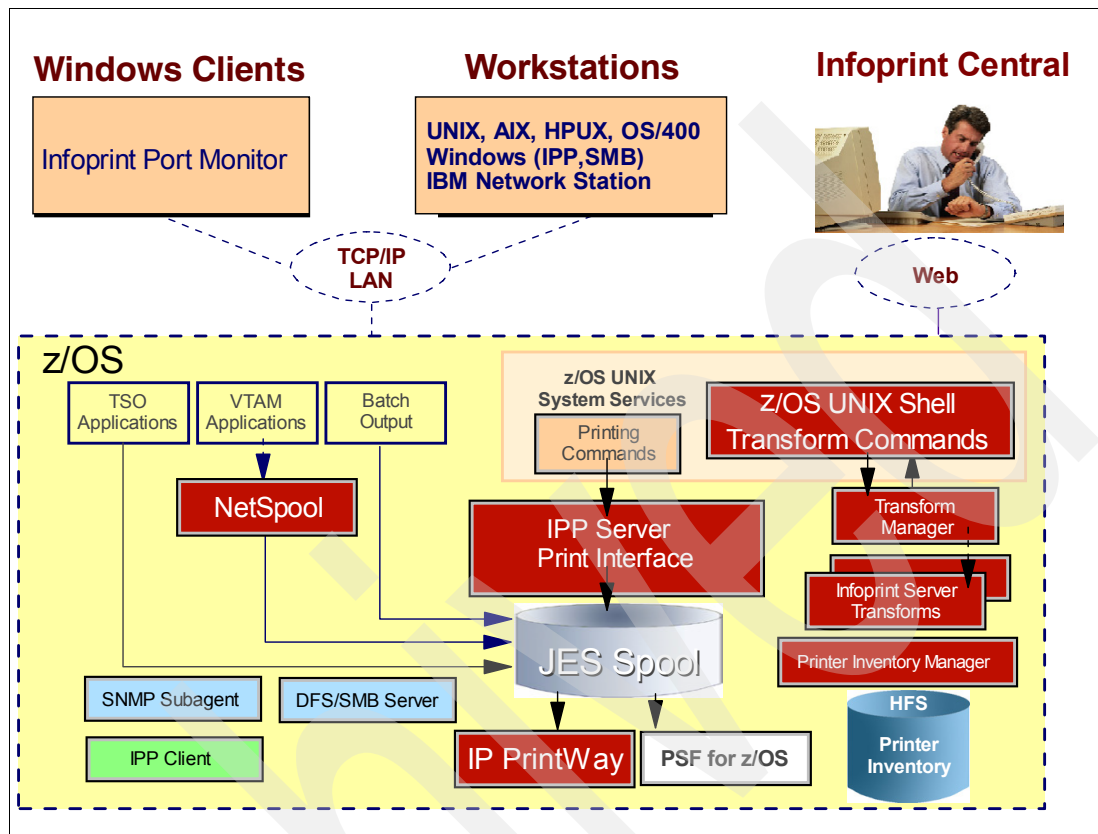


Figure 1-3 Infoprint Server components

### Infoprint Server components

- ▶ **Print Interface** - The Print Interface component of Infoprint Server processes print requests received from both remote clients and local users.
- ▶ **Printer Inventory** - The Printer Inventory consists of HFS or zFS files that contain information about the printing environment. The Printer Inventory Manager component of Infoprint Server controls the Printer Inventory.
- ▶ **IP PrintWay** - The IP PrintWay component of Infoprint Server transmits output data sets from the JES spool to remote printers or print servers and to e-mail destinations. You can run either IP PrintWay basic mode or IP PrintWay extended mode.
  - IP PrintWay basic mode uses the z/OS Functional Subsystem Interface (FSI) to obtain output data sets from the JES spool. IBM does not plan additional enhancements to IP PrintWay basic mode.
  - IP PrintWay extended mode uses the z/OS Sysout Application Programming Interface (SAPI) to obtain output data sets from the JES spool. It provides better performance, improved usability, and more function than IP PrintWay basic mode.

**Note:** The SYSOUT Application Program Interface (SSI function code 79) allows JES to function as a server for applications needing to process SYSOUT data sets residing on JES spool. Use of the SAPI SSI call allows a program to access JES SYSOUT data sets independently from the normal JES-provided functions (such as print or network).

Protocols used by IP PrintWay to transmit output data sets to printers:

- LPR: The LPR protocol is a TCP/IP protocol defined by RFC 1179. An LPD that adheres to RFC 1179 must be running in the remote printer or system.
- Direct-sockets: The direct sockets printing protocol is a TCP/IP protocol in which data is transmitted directly to a designated port. The remote printer or print server must support direct sockets printing.
- Internet Printing Protocol (IPP): IPP is a standard TCP/IP protocol for printing over the Internet. An IPP Server must be running in the remote printer or system.
- Virtual Telecommunications Access Method (VTAM): IP PrintWay can print on printers that are defined to VTAM as LU type 0, LU type 1, or LU type 3. Supported output data streams are SNA character string (SCS) and Data Stream Compatible/Data Stream Extended (DSC/DSE).

Infoprint Coaxial Printer Support for z/OS (5655-N62) is required to print on VTAM-controlled printers.

- E-mail: IP PrintWay can use the z/OS UNIX sendmail function to send print output to one or more e-mail addresses. IP PrintWay sends the output, which can be in any data format, as an e-mail attachment.
- ▶ Infoprint Server Transform Interface - The Transform Interface component of Infoprint Server communicates with transforms that Infoprint transform products provide. Transforms convert data from one format to another, for example, from PCL to AFP format and from AFP to PDF format.  
  
Transform Interface can communicate with transforms that run on the z/OS system and on non-z/OS systems.
- ▶ NetSpool - The NetSpool component of Infoprint Server intercepts print data from VTAM applications, such as CICS and IMS; transforms the data streams to EBCDIC line data, PCL, PDF, or other formats that the target printer accepts; and creates output data sets on the JES spool. Existing VTAM applications can continue to send print requests to NetSpool in the same manner as they currently send print requests to SNA network printers.
- ▶ z/OS UNIX printing commands - Print Interface provides enhanced z/OS UNIX shell commands for printing: lp, lpstat, and cancel.
- ▶ Simple Network Management Protocol (SNMP) Subagent - The Infoprint Server SNMP subagent lets administrators monitor printer characteristics and printer status. This support does not let administrators change any printer characteristics.

To monitor PSF printers, the z/OS SNMP agent must be configured and an SNMP manager, such as IBM Network Printer Manager (NPM) for the Web, must be installed.

- ▶ Windows Workstation support - The Infoprint Server Windows client program:
  - Infoprint Port Monitor for Windows - The software that runs on a Windows workstation. It lets you print files on z/OS printers just like you print files on local Windows printers. You can submit the files from any Windows application that has a printing function.
- ▶ Other related products that also run on Windows systems. Although they are not part of the Infoprint Server Windows client, you might want to use them if your installation has IBM AFP printers or AFP documents:
  - AFP Printer Driver for Windows - Creates output files in AFP format. Files in AFP format can be printed on IBM AFP printers.
  - AFP Viewer Plug-in for Windows - Lets you view documents in AFP format. The Infoprint Port Monitor for Windows is shipped with Infoprint Server.

## 1.4 Print Interface

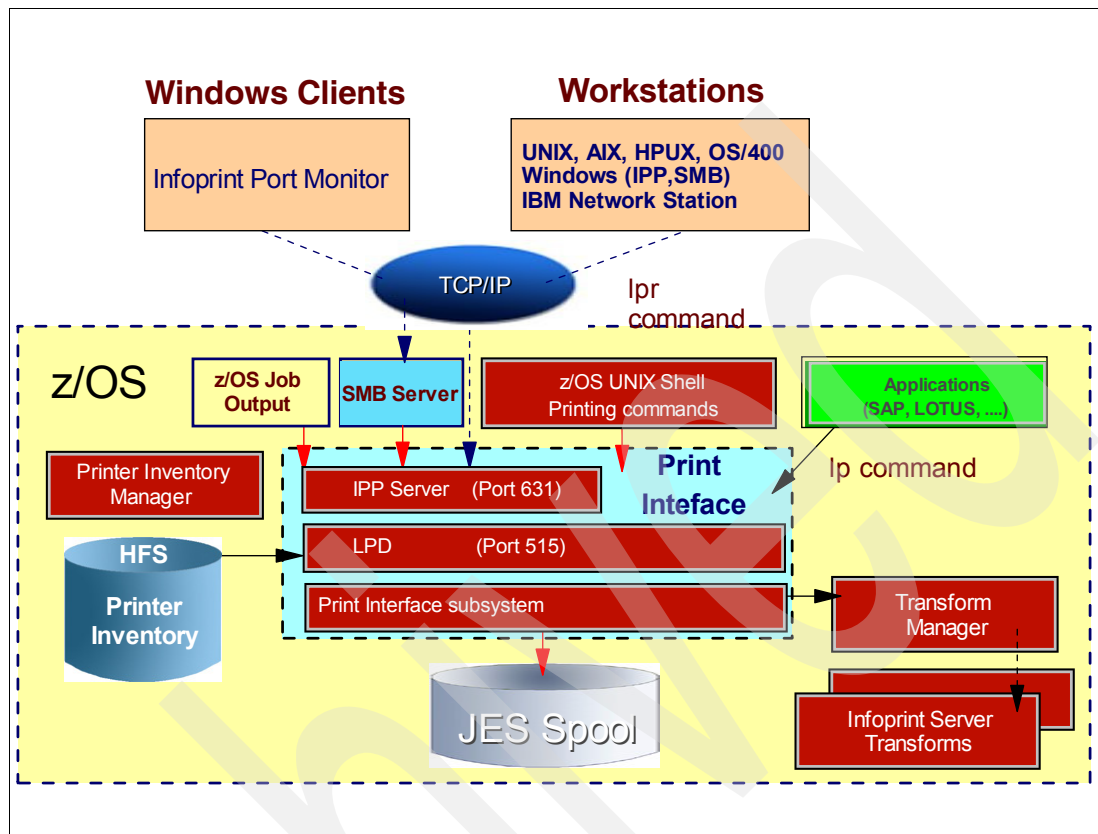


Figure 1-4 Print Interface component

### Print Interface component

The Print Interface component of Infoprint Server provides z/OS UNIX System Services printing commands and a subsystem that let you print from the local z/OS system. Print Interface accepts different data formats; transforms the data streams to EBCDIC line data, AFP, PCL, PostScript, PDF, or other data formats that the printer accepts; and allocates output data sets on the JES spool.

Print Interface also provides a line print daemon (LPD) that lets you submit print requests from remote workstations that have TCP/IP access. In addition, Print Interface can receive print requests from remote workstations that use the Internet Printing Protocol (IPP) or the Server Message Block (SMB) printing protocol, which is standard on Windows systems. Print Interface also provides an SAP R/3 Output Management System that can receive SAP R/3 print requests.

Print Interface does the following:

- It creates an output data set on the JES spool for each document to be printed. Print Interface maps the printing options specified on **lp** commands and some of the printing options specified on **lpr** commands to JES output parameters. These parameters are the same parameters that you can specify on JCL statements when you submit batch jobs.
- It responds to query requests with the status of the output data set on the JES spool or a list of the printers defined in the Printer Inventory.

- ▶ It responds to cancel requests by removing data sets that are on the JES spool. The data sets must not yet have been selected for printing. For security reasons, Print Interface does not let unauthenticated users on remote systems cancel print requests.
- ▶ Print Interface can automatically transform data to the format specified for the target printer.

After Print Interface creates output data sets on the JES spool, the data sets are available for printing:

- ▶ JES or PSF can print the data sets.
- ▶ JES can transmit the data sets to another location for printing.
- ▶ IP PrintWay can transmit the data sets to a remote printer in your TCP/IP or SNA network.

Some of the functions of Print Interface are:

- ▶ Printing of any data format that the printer supports: This lets users submit print requests with any data format that the printer supports. These formats include, but are not limited to, PCL, PostScript, MO:DCA-P, line data, XML, and text. Print Interface automatically detects the data format.
- ▶ Automatic data transform: Print Interface can automatically transform data when Infoprint transform products are installed.

Print Interface can also automatically transform data to the AFP format remotely on an AIX® or Windows system using transforms that Infoprint Manager for AIX (5785-E42) and Infoprint Manager for Windows (5639-127) provide.

Print Interface can use the PostScript to AFP and the PDF to AFP color transform that Infoprint Manager for AIX provides so that you can print color PostScript and color PDF files on the IBM Infoprint Color 130 Plus printer.

- ▶ Validation of print requests: Before accepting print requests, Print Interface validates that a document can print as requested on the selected printer. For example, Print Interface rejects a document with a data format that the printer does not support.
- ▶ Notification of completion: Print Interface notifies users on the local z/OS system when processing of a document is complete.
- ▶ Identification of printed output - Print Interface maintains the user ID of the job submitter for printing on separator pages. Both PSF and IP PrintWay allow installations to write an exit to print separator pages.
- ▶ Double-byte character set (DBCS) support: Print Interface converts DBCS data from one code page to another before writing the data to the JES spool.
- ▶ IPP server: Print Interface includes an IPP server to support printing from Windows 2000 and Windows XP. The IPP server lets Windows users print on z/OS printers without installing the Infoprint Port Monitor for Windows.
- ▶ SMB support: Print Interface includes support for SMB printing, which is used by Windows systems. SMB support lets Windows users print on z/OS printers without installing the Infoprint Port Monitor for Windows.
- ▶ Print Interface subsystem: The Print Interface subsystem can transform and print output from z/OS applications with minimal changes to the applications' job control language (JCL). The subsystem processes the application's output, transforms it to the format required by the printer or e-mail destination, and writes it to the JES spool. From the JES spool, the transformed data can be printed on any printer or be sent to e-mail destinations.

Print Interface includes enhanced z/OS UNIX System Services shell printing commands, which provide more function than the printing commands available without Infoprint Server.



These enhanced commands adhere to the XPG4.2 standard to facilitate the porting of UNIX applications to z/OS UNIX System Services:

- ▶ The **lp** command sends files for printing to Print Interface running on the local z/OS system. The files can be UNIX files or traditional MVS™ data sets, such as sequential data sets and partitioned data sets.
- ▶ The **lpstat** command queries the status of print jobs. It also queries the names, locations, and descriptions of printers that the administrator has defined in the Printer Inventory.
- ▶ The **cancel** command cancels print requests, provided that the data set allocated on the JES spool has not yet been selected for printing.

## Starting Infoprint Server

When you start Infoprint Server, a set of address spaces are created for the Print Interface functions. The functions in these UNIX address spaces include:

- ▶ Support for the z/OS UNIX System Services printing commands **lp**, **lpstat**, and **cancel**.
- ▶ The LPD function that lets you submit print requests to Infoprint Server with commands that use the LPR to LPD protocol, such as:
  - **LPR** and **LPQ** commands from MVS, z/OS, VM, z/VM®, AS/400R, OS/2R, UNIX, and Windows systems
  - **lpr**, **enq** and **qstat** commands from AIX systems
- ▶ Support for Internet Printing Protocol (IPP) standard protocol for printing over the Internet. An IPP client runs in the remote system.
- ▶ Receive print requests from remote workstations that use the SMB printing protocol, (standard on Windows systems).
- ▶ Provide an SAP R/3 Output Management System that can receive SAP R/3 print requests.

## 1.5 LPR/LPD Protocol - RFC1179

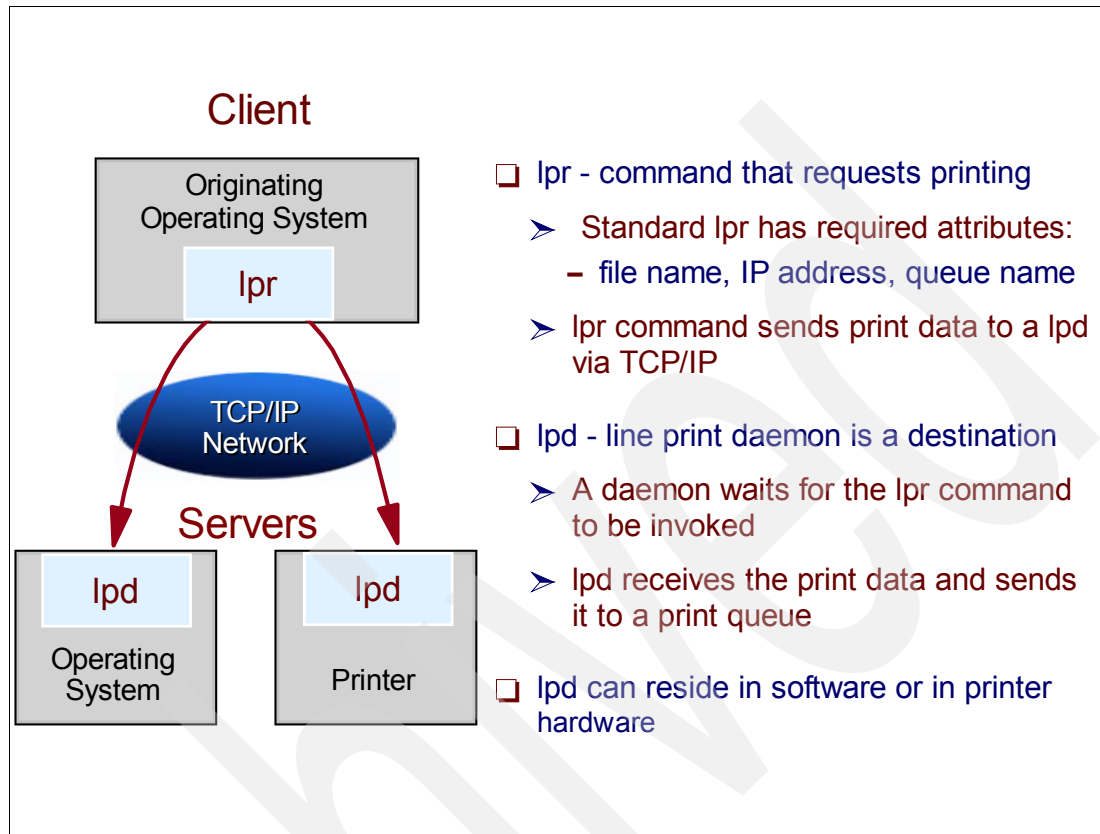


Figure 1-5 LPR/LPD protocol - RFC1179

### LPR to LPD protocol

LPR/LPD protocol is a platform-independent printing protocol for remote printing that allows multiple platforms to print to the same LPD.

It is one of the protocols that Infoprint Server uses to accept print requests and to send print data.

The Network Printing Working Group defined the line printer daemon (LPD) protocol in RFC1179. This RFC describes the existing print server protocol, which is widely used on the Internet for communicating between line printer daemons (both clients and servers). The protocol describes in detail how a line printer daemon client requests and controls printing.

LPR is a TCP/IP based command. The default port on which an line printer daemon (LPD) listens is 515. The source port must be in the range 721 to 731, inclusive. An LPD (the server "daemon") responds to commands sent to its port and is the part that receives and processes the request.

**Note:** The Infoprint Server's Print Interface LPD does not require that the LPR use a source port in the range of 721 to 731. Generally, performance on the system where the LPR runs is improved if you do not restrict the range for the source port.

## 1.6 IP PrintWay basic mode support

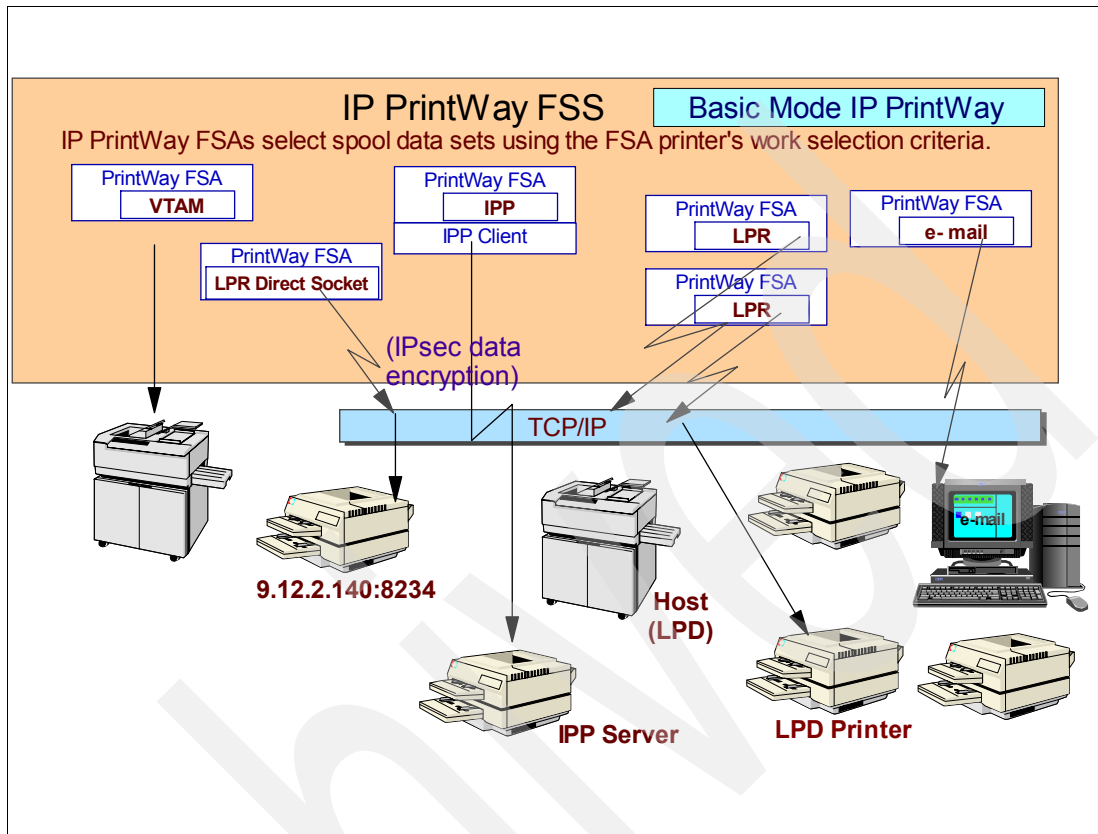


Figure 1-6 IP PrintWay component using basic mode

### IP PrintWay basic mode

IP PrintWay transmits output data sets from the JES spool to remote printers or print servers and to e-mail destinations using either IP PrintWay basic mode or IP PrintWay extended mode.

The IP PrintWay basic mode is implemented as a functional subsystem (FSS) and uses the z/OS Functional Subsystem Interface (FSI) to obtain output data sets from the JES spool.

In an IP PrintWay basic mode FSS definition you specify configuration information that applies to an IP PrintWay FSS and to all functional subsystem applications (FSA) in that FSS. An IP PrintWay FSS definition is required:

- ▶ If any of the FSAs in the FSS will transmit output to VTAM-controlled printers. In this case, you must specify the name of the VTAM APPL statement you defined for the FSS.
- ▶ If you need to change the IP PrintWay configuration defaults. If an IP PrintWay FSS definition does not exist when IP PrintWay starts, IP PrintWay uses default values.

IP PrintWay transmits print data sets to remote printers or to host systems in your TCP/IP network. The remote printer or host system must support either the LPR/LPD protocol, the IPP protocol, or direct socket printing.

**Note:** A functional subsystem (FSS) is a collection of programs residing in an address space separate from JES that communicates with JES to provide a JES-related function, such as print processing. Because an FSS operates in its own address space, it functions independently of JES in several areas.

A functional subsystem application (FSA) is a collection of programs residing in the FSS address space that control one device. There can be multiple FSAs per FSS.

The Functional Subsystem Interface (FSI) services are actually JES and FSS/FSA supplied routines that allow interaction between JES and the FSS/FSA.

## Starting basic mode

IP PrintWay basic mode starts when an operator enters the IP PrintWay FSA printer start command. The JES FSA printer start command starts the FSS address space automatically when the first FSA under the control of that FSS is started.

In the IP Printway basic mode printer work selection criteria is defined to JES. JES performs the work selection for IP Printway basic mode.

IP PrintWay basic mode can also transmit output data sets from the JES spool to VTAM-controlled printers that are defined to VTAM as LU0, LU1, or LU3 printers. The supported output data streams are:

- ▶ Data Stream Compatibility/Data Stream Extended (DSC/DSE)
- ▶ SNA Character String (SCS)

**Note:** This support in IP PrintWay requires that you install the Coax Printer Support feature of Infoprint Server Transforms.

## IPsec data encryption

The IP security function of z/OS Communications Server can encrypt data that IP PrintWay sends to printers and print servers that support Internet Protocol security (IPsec). Encrypting data ensures the confidentiality of the data that you send to the printer.

To encrypt data sent from z/OS to a printer that supports IPsec, no changes are required to Infoprint Server or to your printer definitions. However, the network administrator must customize the IP security function that z/OS Communications Server provides, and you must enable IPsec in your printers. Both IP PrintWay basic mode and IP PrintWay extended mode can send encrypted data to printers that support IPsec, using any of the TCP/IP transmission protocols that IP PrintWay supports (LPR, direct sockets, and IPP).

Many newer printers, including Hewlett Packard (HP), Lexmark, and IBM printers, support IPsec.

**Note:** **IPSec** is a suite of protocols and standards defined by the Internet Engineering Task Force (IETF) for secure communication over an existing IP network.

## 1.7 IP PrintWay extended mode support

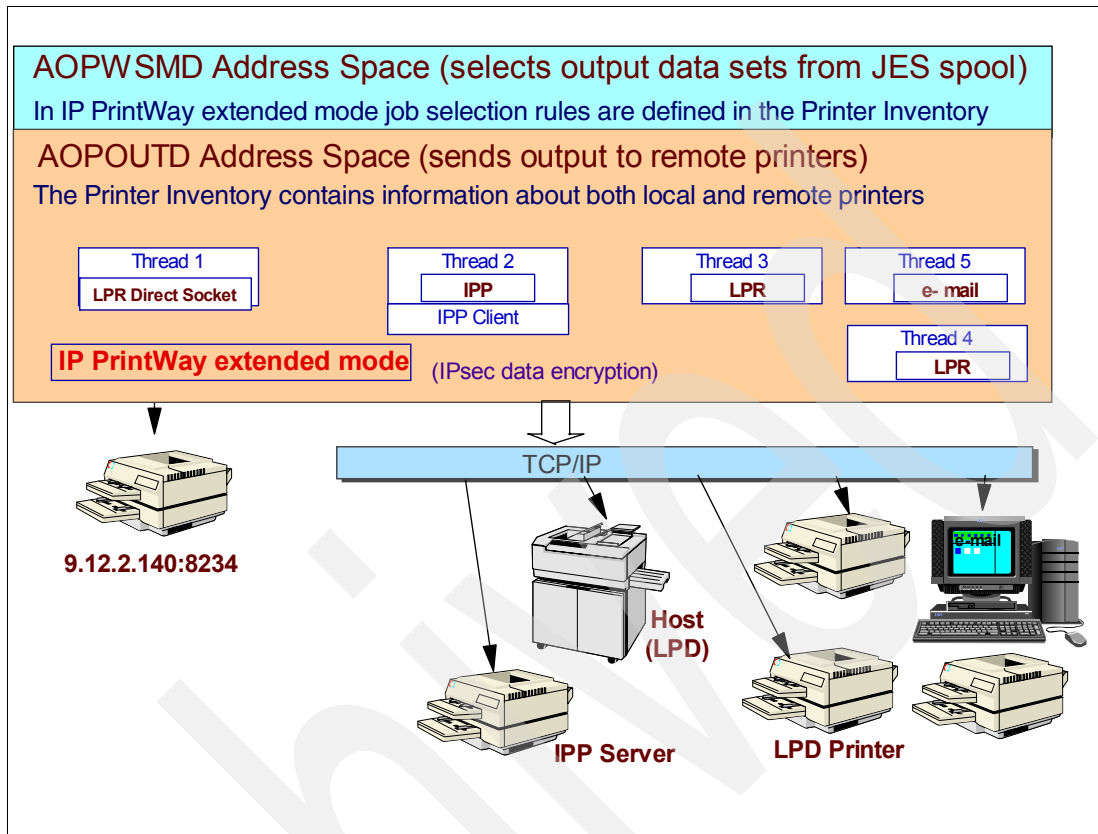


Figure 1-7 IP PrintWay extended mode support

### IP PrintWay extended mode

IP PrintWay extended mode uses the z/OS Sysout Application Programming Interface (SAPI) to obtain output data sets from the JES spool. This implementation results in better performance and improved usability compared with IP PrintWay basic mode:

- ▶ Better performance:
  - Most jobs start printing sooner.
  - A printer problem or large job on one printer does not delay printing to another printer.
  - IP PrintWay extended mode calls data stream transforms and other filters directly without resubmitting them to Print Interface for filtering. Calling transforms directly is more efficient because data is not written to the JES spool a second time.
- ▶ Improved usability:
  - Operators can use Infoprint Central, instead of Infoprint Server ISPF panels, to work with IP PrintWay extended mode printers and print jobs.
  - Operators can use JES commands to work with print jobs that IP PrintWay extended mode has selected to process, is waiting to retry, or has retained on the JES spool. However, to work with print jobs that IP PrintWay is currently processing, operators must use Infoprint Central.
  - IP PrintWay extended mode can print data sets larger than 2 gigabytes if space is available in the file system.

- IP PrintWay extended mode prints output data sets in priority order.
  - IP PrintWay extended mode can restart printing after an error from the last page that the printer reported printed successfully.
  - IP PrintWay extended mode can print on printers that have IPv6 addresses. However, you must use the host name (instead of the colon-hexadecimal address) in the DEST=IP: JCL parameter, in Infoprint Server job attributes, and in printer definitions.
  - IP PrintWay extended mode writes the printer address for all protocol types in the System Management Facilities (SMF) type 6 record.
  - IP PrintWay extended mode has enhanced how it selects print jobs from the JES spool:
    - It can select all print jobs that specify the DEST=IP: parameter on the OUTPUT JCL statement. You specify this job-selection criterion in a job-selection rule in the Printer Inventory.
    - If print jobs request a printer that is not defined in the Printer Inventory, IP PrintWay extended mode releases the print job back to JES so that another program can select the print job for processing.
  - Authorized users can use Infoprint Central, a Web-based application, to do additional printer functions and print job functions.
  - IP PrintWay extended mode always uses the z/OS iconv utility to convert between EBCDIC and ASCII code pages. It does not use standard or customized TCP/IP translate tables as IP PrintWay basic mode can.
- Limitation
- **Sysplex:** IP PrintWay extended mode cannot share printing information across multiple systems in a sysplex as IP PrintWay basic mode can.
    - IP PrintWay basic mode keeps queue information in its transmission-queue data set and keeps messages in its message-log data set. Both data sets can be shared across multiple systems.
    - IP PrintWay extended mode keeps queue information in the Printer Inventory and keeps messages in the common message log. The Printer Inventory and common message log cannot be shared across multiple systems.

IP PrintWay extended mode and IP PrintWay basic mode use the same printer definitions in the Printer Inventory. However, some fields apply to either IP PrintWay basic mode or extended mode, but not both. Fields that do not apply are ignored.

When you run IP PrintWay extended mode, the administrator defines job selection rules in the Printer Inventory to specify which print jobs to select. The operator can use Infoprint Central to start and stop these job selection rules, or the administrator can automate the starting and stopping of job selection rules.

When Infoprint Server is started and the Infoprint Server configuration file (aopd.conf) *start-daemons* specification includes *outd*, address spaces for the *aopoutd* and *aopwsmd* daemons are created. These IP PrintWay extended mode daemons select output data sets from the JES spool and send them to remote printers in a TCP/IP network or to e-mail destinations.

Infoprint Server environment variables APOOUTD\_MAXTHREADTASKS and AOPWSMD\_MAXTHREADTASKS set the limits on the number of threads for the aopoutd and aopwsmd daemons. If an environment variable is not specified, the limit is 200.

## 1.8 Printer Inventory Manager

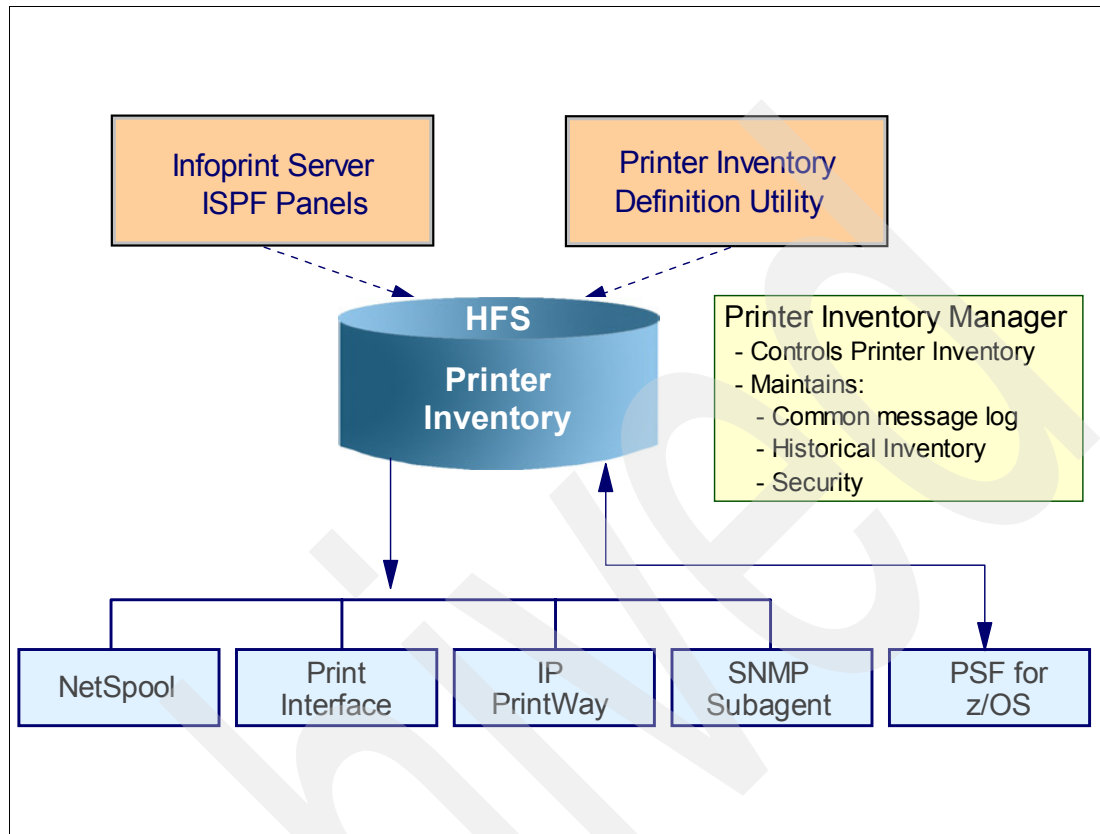


Figure 1-8 Printer Inventory Manager component

### Printer Inventory

The Printer Inventory, a set of HFS files maintained by the system administrator, contains information about the z/OS printers, including both local and remote printers.

The Printer Inventory is common to the Print Interface, IP PrintWay, and NetSpool components of Infoprint Server. This means that the system administrator can define a printer once for all components of Infoprint Server. The Printer Inventory, however, cannot be shared by Infoprint Server components running on other z/OS systems.

The Printer Inventory in one central location contains all the information needed to:

- ▶ Validate print requests
- ▶ Transform data to a format supported by a printer
- ▶ Assign processing attributes to the data sets allocated on the JES spool
- ▶ Direct data sets to printers

### Printer Inventory Manager

The Printer Inventory Manager controls the Printer Inventory. The Printer Inventory is maintained by the system administrator using an ISPF application or a Print Inventory Definition Utility (pidu) and contains the following types of objects (definitions):

**Printer definitions**      Information about printers and e-mail destinations.

<b>Printer pool definitions</b>	Information about groups of printer definitions that you want to broadcast data to (applies only to NetSpool).
<b>FSS definitions</b>	Configuration information for IP PrintWay basic mode functional subsystems (FSSs) and for PSF FSSs.
<b>FSA definitions</b>	Configuration information for IP PrintWay basic mode functional subsystem applications (FSAs) and for PSF FSAs.
<b>Job selection rules</b>	Rules that IP PrintWay extended mode uses to determine which print jobs to select from the JES spool for printing.

## Printer Inventory and NetSpool

Each NetSpool started task runs in its own system address space and processes data sets for different classes of NetSpool printer LUs. When a NetSpool task starts, it attempts to start any printer LUs defined in the Printer Inventory and assigned to one of the started LU classes specified in the NetSpool startup procedure. If the administrator defines a new printer LU in the Printer Inventory after you start the NetSpool task, NetSpool automatically starts that printer LU if it is in one of the started LU classes.

## Printer Inventory and Print Interface

When Print Interface allocates output data sets on the JES spool it extracts the attributes defined for the target printer in the Printer Inventory, assigns the attributes to the spool data sets, and invokes transforms.

## Printer Inventory and IP PrintWay

IP PrintWay uses information in printer definitions in the Printer Inventory to process data sets, select the transmission protocol, and obtain printer addresses.

IP PrintWay extended mode's job selection rules are stored in the Printer Inventory and determine which print jobs IP PrintWay selects from the JES spool.

## Printer Inventory and Infoprint Central

Infoprint Central is a Web-based print management system primarily for desk operators help. Infoprint Central works with IP PrintWay extended mode. With Infoprint Central **Work with printers** you can work with printers that are defined in the Printer Inventory - including IBM AFP printers controlled by PSF (called PSF printers), and TCP/IP-attached printers to which IP PrintWay extended mode sends print jobs (called IP PrintWay printers). You can find and work with IP PrintWay printers only when you run IP PrintWay extended mode.

The selection **Work with NetSpool logical units** allows you to work with any NetSpool logical units (LUs) that are defined to NetSpool in the Printer Inventory and also defined to VTAM.

## Printer Inventory and SNMP

The Simple Network Management Protocol (SNMP) reporting for a PSF-controlled printer is enabled in the FSA definition for the PSF printers in the Printer Inventory.

## Printer Inventory and PSF

The Print Service Facility (PSF) must be customized to use the Printer Inventory. If PSF is to use the Printer Inventory, an FSS definition must exist in the Printer Inventory. When PSF uses the Printer Inventory, the Infoprint Server SNMP subagent can display printer status for PSF printers. The SNMP reporting can be requested for PSF for channel, SNA, and TCP/IP attached PSF printers (FSAs).



Any SNMP manager that displays the status of printers defined to a printing system, such as IBM Network Printer Manager for the Web (NPM), can manage PSF printers through the z/OS SNMP subagent.

### Additional Printer Inventory Manager functions

- **Common message log** - The common message log lets you see messages from most Infoprint Server components in one place. The log contains messages from all components of Infoprint Server except for IP PrintWay basic mode. Also, the log contains error messages from Infoprint transforms that fail. (If a transform completes successfully, messages related to problems in the input data stream are written at the end of the output instead of to the common message log.)

IP PrintWay extended mode writes its messages only to the common message log. Other components, such as NetSpool and Print Interface, write their messages to the common message log and also to other locations such as the NetSpool message-log data set and the system console log.

Messages in the common message log can be viewed and processed by:

- Authorized Infoprint Central users can view messages for selected print jobs and IP PrintWay printers.
  - Infoprint Server administrators can use the **aoplogu** command to select messages in a particular time range and copy them to a file or view them on the terminal.
  - Infoprint Server administrators can send all or selected messages to the z/OS system log (SYSLOG) or operations log (OPERLOG) for message automation.
- **Historical Inventory** - The Historical Inventory contains information about data sets that Infoprint Server has processed but that are no longer on the JES spool because they finished processing or were deleted. Infoprint Central lets authorized users display information about data sets (called print jobs) in the Historical Inventory.
  - **Migration program** - The Infoprint Server migration program helps the administrator move printer information located in PSF startup procedures to the Printer Inventory. It automatically creates PSF FSS and FSA definitions in the Printer Inventory.

The migration program can also help the administrator migrate from the IP PrintWay and NetSpool features of PSF V3R2 and earlier releases.

- **Security** - You can use RACF to restrict access to the Printer Inventory, the common message log, and operator commands that start and stop Infoprint Server daemons.

## 1.9 Infoprint Server Transforms

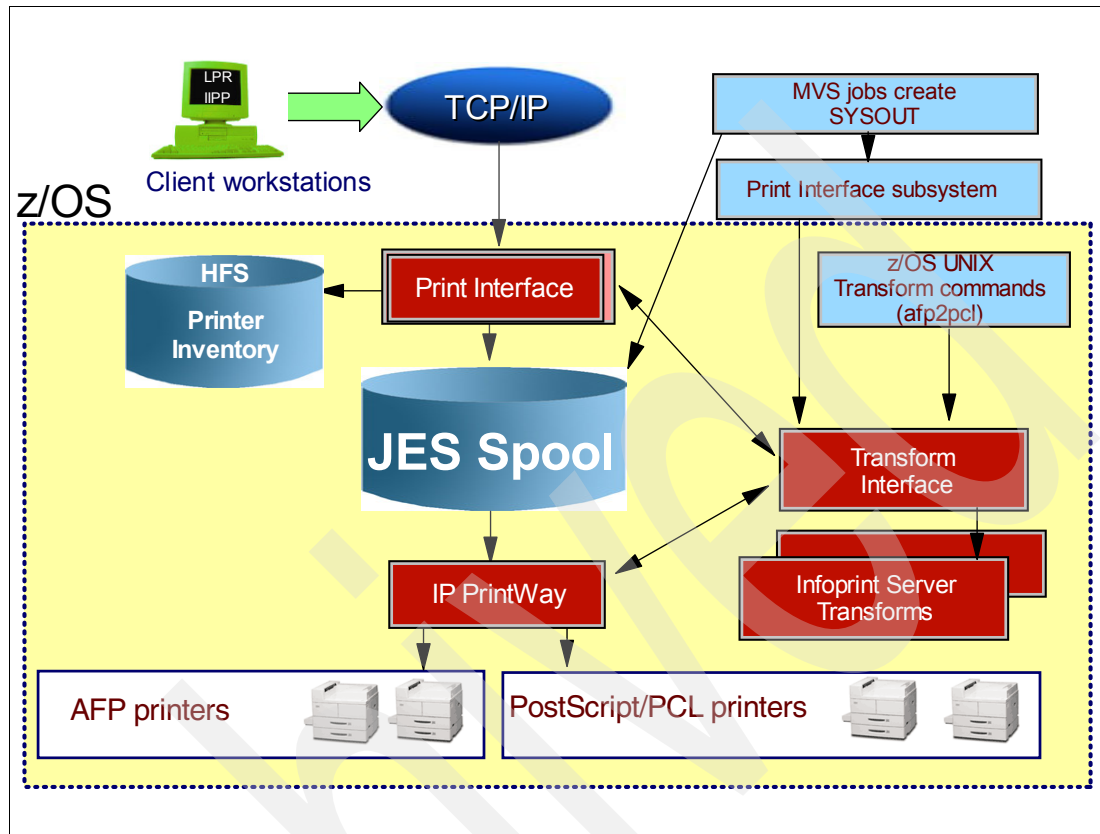


Figure 1-9 Infoprint Server Transforms component

### Infoprint Server Transforms

Transforms are programs that convert a data stream from one format to another, for example, from PCL to AFP, PDF to AFP, and so on. The IBM-provided transforms are implemented as DLL filters. These data stream transforms give you the flexibility to print a variety of output on a wide range of printers. The transform runs automatically when associated as a filter for an IP PrintWay printer definition supported data format.

In Figure 1-9 an LPR print request of a client data set from the workstation comes through the Infoprint Server's Print Interface (aoplpd) to JES spool. The Printer Inventory entry for the target AFP printer is specified such that the client data set will be transformed to AFP format for printing on the AFP printer.

IP PrintWay extended mode calls transforms directly, while IP PrintWay basic mode transforms data by resubmitting it to Print Interface. Calling transforms directly is more efficient because data is not written to the JES spool a second time.

The Print Interface subsystem (also shown in Figure 1-9) can transform and print output from z/OS applications. The subsystem processes the application's output, transforms it to the format required by the printer or e-mail destination, and writes it to the JES spool. From the JES spool, the transformed data can be printed on any printer or be sent to e-mail destinations.

You can also run the transforms as standalone commands from z/OS UNIX System Services. The transforms are all separately priced products. You should purchase the transforms only if you need to use them.

## IBM Infoprint transform products

The following IBM Infoprint transform products work with Infoprint Server:

- ▶ **Infoprint Transforms to AFP for z/OS, 5655-N60** transforms documents to Advanced Function Presentation (AFP) format from HP Printer Control Language (PCL), Adobe Portable Document Format (PDF), Adobe PostScript, or SAP R/3 System Generic Output Format (SAPGOF).
- ▶ **Infoprint Transform for AFP to HP PCL for z/OS, 5655-P19** transforms documents from AFP format to PCL.
- ▶ **Infoprint Transform for AFP to Adobe PDF for z/OS, 5655-P20** transforms documents from AFP format to PDF.
- ▶ **Infoprint Transform for AFP to Adobe PostScript for z/OS, 5655-P21** transforms documents from AFP format to PostScript.
- ▶ **Infoprint Coaxial Printer Support for z/OS, 5655-N62** transforms line data to Data Stream Compatibility/Data Stream Extended (DSC/DSE) or SNA Character String (SCS). IP PrintWay requires this transform to print on VTAM-controlled printers in an SNA network.
- ▶ **Infoprint XML Extender for z/OS, 5655-J66** transforms Extensible Markup Language (XML) files to AFP or PDF format for printing or e-mailing. To transform XML files to another format, you can transform XML to AFP, and then transform AFP to PCL or PostScript.
- ▶ **Infoprint XT Extender for z/OS, 5655-J65** transforms Xerox files to AFP format for printing or e-mailing. The Xerox files can be line-conditioned data streams (LCDS) or metacode data streams. XT is the IBM Xerox Transform technology.

## z/OS UNIX shell transform commands

z/OS UNIX Services users can run the transforms with the following z/OS UNIX System Services shell commands:

- |                |   |
|----------------|---|
| <b>pcl2afp</b> | Transforms a PCL file to an AFP file.   |
| <b>pdf2afp</b> | Transforms a PDF file to an AFP file.   |
| <b>ps2afp</b>  | Transforms a PostScript file to an AFP file.                                  |
| <b>sap2afp</b> | Transforms an SAP ABAP or SAP OTF Version 1 or Version 2 file to an AFP file. |
| <b>afp2pdf</b> | Transforms an AFP or line data file to a PDF file.                            |
| <b>afp2pcl</b> | Transforms an AFP or line data file to a PCL file.                            |
| <b>afp2ps</b>  | Transforms an AFP or line data file to a PostScript file.                     |
| <b>xml2afp</b> | Transforms an XML file to an AFP file.  |
| <b>xml2pdf</b> | Transforms an XML file to a PDF file.   |

Infoprint Server Transforms also provide support for printing Japanese DBCS data streams on IBM AFP printers and for printing line data on coaxially-attached VTAM-controlled printers.

## 1.10 NetSpool

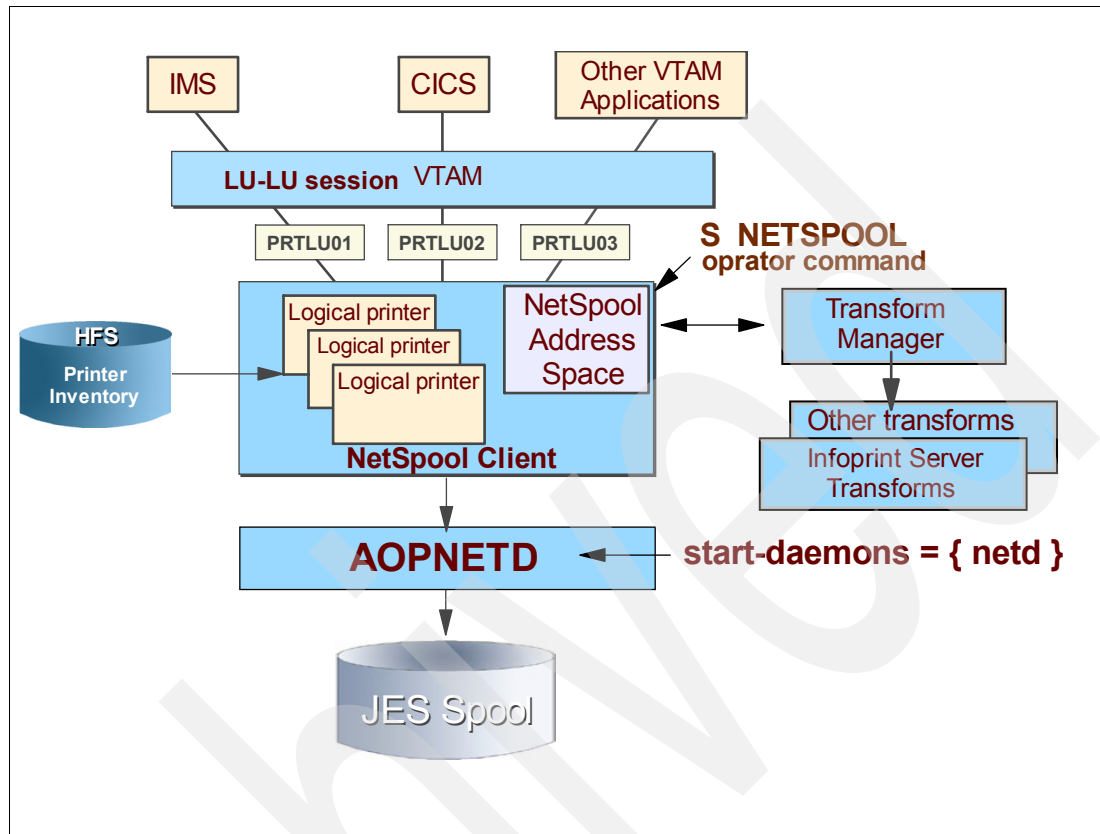


Figure 1-10 NetSpool component

### NetSpool component

NetSpool intercepts print data from VTAM applications, such as CICS and IMS; transforms the data streams to EBCDIC line data, PCL, PDF, or other formats that the printer accepts; and allocates output data sets on the JES spool. JES or PSF can print the output data sets or JES can transmit them to other locations for printing. IP PrintWay can transmit the output data sets to remote printers in your TCP/IP network.

You can configure NetSpool so that you do not need to change existing VTAM applications. That is, existing VTAM applications can send print requests to NetSpool in the same manner as they currently send print requests to SNA network printers.

By placing VTAM application output on the JES spool, NetSpool lets you take advantage of the security, checkpoint and restart, and reprint capabilities that JES and IP PrintWay provides.

VTAM data streams accepted by NetSpool:

- ▶ SNA character string (SCS) data over a logical unit (LU) type 1 session
- ▶ 3270 data over an LU type 0 or type 3 session
- ▶ Binary data over an LU type 0, type 1, or type 3 session

VTAM applications, such as CICS or IMS, establish communication sessions with NetSpool printer logical units (LUs) instead of with SNA-network printers. Each NetSpool printer LU must be defined to VTAM as an *application* logical-unit (LU).

## Using NetSpool

NetSpool runs as a VTAM application on the same or a different z/OS system. NetSpool can process VTAM print requests sent to different NetSpool printer LUs. Each NetSpool printer LU is defined in a printer definition or printer pool definition in the Printer Inventory. NetSpool uses information in the printer definition to format data into lines and pages, transform data to a format that the printer accepts, and group the data into output data sets.

NetSpool can transform SCS and 3270 data streams to EBCDIC line data or ASCII PCL data streams. It can also use other Infoprint transform products to convert line data to other formats for printing or e-mailing.

NetSpool dynamically allocates output data sets on the JES spool using JES allocation parameters specified in the printer definition, including:

- ▶ JES work-selection parameters, such as class, forms name, and destination. These parameters cause JES to direct the output data sets to the correct program, such as IP PrintWay or PSF.
- ▶ Advanced Function Presentation (AFP) parameters, such as the name of a form definition and page definition. PSF can use these parameters when printing data on IBM AFP printers.
- ▶ Distribution information, such as name and address, which can be printed on output header pages.

Additional functions that NetSpool provides:

- ▶ **Transforms** - NetSpool can transform data to line data or PCL format without using a separate transform product. It can:
  - Transform SCS and 3270 data streams into line data streams.
  - Transform SCS and 3270 data streams into PCL data streams. PCL data can be printed on network printers.

NetSpool transforms data to PDF or PostScript format using an Infoprint transform product by first transforming SCS and 3270 data to line data, and then using the transform.

- ▶ **Validation of print requests** - NetSpool validates, with some exceptions, that the document can print as requested on the selected printer.
- ▶ **Operator control** - Operators can control NetSpool LUs from Infoprint Central, from the system console, or from extended MCS consoles. For example, the operator can display the status of NetSpool LUs, stop them, and start them.
- ▶ **Broadcasting output** - NetSpool supports printer pool definitions in the Printer Inventory. When VTAM application data is submitted to a printer pool definition, NetSpool creates multiple output data sets on the JES spool.
- ▶ **Exits and filters** - System programmers can write and install exits to customize NetSpool processing. NetSpool exits can add data to the beginning of output data sets, map graphic escape characters to other printable characters, and modify or delete transparent data in SCS data streams.
- ▶ **Binary data support** - The administrator can request in the printer definition that NetSpool treat the data stream as binary data. NetSpool writes binary data to the output data set as variable length records without formatting the data and without rejecting unsupported commands, orders, or data. This function is useful if you want to pass through all data without change and without including transparent (TRN) controls.
- ▶ **Owner and job attribute support** - VTAM application programmers can embed the owner job attribute and other Infoprint Server job attributes in the application's print data that is sent to NetSpool.

## 1.11 SNMP Overview

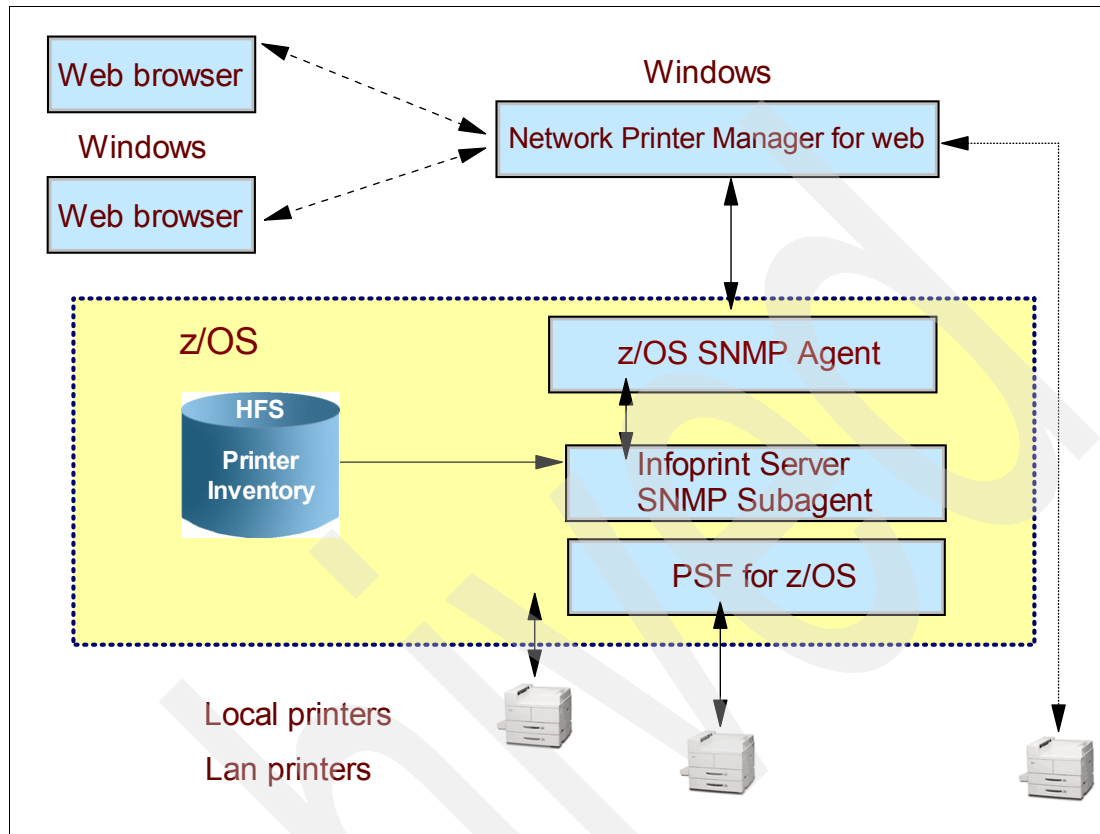


Figure 1-11 SNMP subagent component

### SNMP subagent

Simple Network Management Protocol (SNMP) is a set of protocols that describes management data and the protocols for exchanging that data between heterogeneous systems. The protocols include both the description of the management data, defined in the Management Information Base (MIB), and the operations for exchanging or changing that information. By implementing common protocols, management data can be exchanged between different platforms with relative ease.

### Infoprint Server SNMP subagent

The Infoprint Server SNMP subagent works with PSF, the z/OS Communications Server SNMP agent, and an SNMP manager application, such as IBM Network Printer Manager for the Web.

The Infoprint Server SNMP subagent reports the status of printers that do not have their own SNMP subagent to a printer management program, for example, the IBM Network Printer Manager (NPM) for the Web, which you can download at no charge from the Printing Systems Division Web site at <http://www.ibm.com/printers/download.html>

**Note:** The IBM Network Printer Manager for the Web (NPM), which works with the SNMP subagent, is no longer in service. However, you can still use the SNMP subagent.

## 1.12 Print work submission to Infoprint Server

- ❑ Workstation users - (lpr command) - (IPP)
- ❑ Windows workstation users
  - lpr command - Native Windows SMB - Infoprint Port Monitor - IPP
- ❑ TSO/E - VM users - LPR command)
- ❑ Users submitting batch JCL
  - AOPPRINT JCL EXEC - Standard JCL
- ❑ z/OS UNIX System Services users
  - lp, lpstat, and cancel commands
- ❑ VTAM application users
  - NetSpool
- ❑ SAP R/3

Figure 1-12 Users who can access Infoprint Server

### Print work submission to Infoprint Server

Users can submit print requests and query job status from *remote* clients in the TCP/IP network using one of these TCP/IP protocols.

### Clients that use LPR protocol

Workstation users can use the following methods to send output data for processing at a z/OS host:

- ▶ The Infoprint Port Monitor for Windows. Infoprint Server provides this client.
- ▶ TCP/IP commands such as **lpr**, **enq**, and **lpq**.
- ▶ Internet Printing Protocol (IPP): IPP is a standard protocol for printing over the Internet. An IPP client must run in the remote system.
- ▶ SMB, which stands for Server Message Block, is a protocol for sharing files, printers, serial ports, and communications abstractions such as named pipes and mail slots between computers.
- ▶ Clients connect to servers using TCP/IP (actually NetBIOS over TCP/IP as specified in RFC1001 and RFC1002), NetBEUI or IPX/SPX. Once they have established a connection, clients can then send commands (Simps) to the server that allow them to access shares, open files, read and write files, and generally do all the sort of things that you want to do with a file system. However, in the case of SMB, these things are done over the network.

- SMB is the standard printing protocol that Windows systems use. The z/OS SMB server must be installed on the z/OS system to receive print requests. The z/OS SMB server uses Print Interface callable services to allocate output data sets on the JES spool and return print job status to the client.

## z/OS users

Users can submit print requests from a z/OS system using one of these methods:

- To print from a z/OS system, use the **LPR** command from a TSO session:  
`LPR 'filename' (P printername AT hostaddress TITLE title`
- With the AOPPRINT JCL procedure, provided in SYS1.PROCLIB, you can print MVS data sets and UNIX files and specify job attributes.
  - If Infoprint Server transforms or another optional transform product is installed, you can automatically transform jobs from one data format to another.
  - Infoprint Server validates that data can print on the selected printer.

## z/OS UNIX users

To print from z/OS UNIX shell, use printing commands (**lp**, **lpstat**, and **cancel**) that Infoprint Server provides. Using these commands, you can print MVS data sets and UNIX files, query the status of a print job, and cancel a print job. You can run these commands from the z/OS UNIX shell command line or from a UNIX application.

```
lp [-csw] [-ddestination] [-ncopies] [-ooption] [-ttitle] [filename...]
```

The **lp** command prints one or more files, or sends the files to an e-mail destination. The address of the printer is specified in the printer definition in the Infoprint Server Printer Inventory. The e-mail addresses are specified in the printer definition or in job attributes. The files can be:

- MVS data sets, such as partitioned data sets or sequential data sets
- UNIX files, such as files in a Hierarchical File System (HFS), a z/Series File System (zFS), a Network File System (NFS), or a temporary file system (TFS)
- Lists of printable files

The **lpstat** command shows printer names and locations and status of print jobs, as follows:

```
lpstat [-dt] [-a [printername ...]] ... [-o [printername ...]] ...  
[-p [printername ...]] ... [-u [userid ...]] ... [jobid ...]
```

**lpstat** returns printer definition names, location information specified in the printer definitions, and the status of jobs to standard output.

For jobs that Infoprint Server has processed (including jobs submitted in any of these ways: from a VTAM application through NetSpool; from a remote system or with the **lp** command through Print Interface; from batch JCL printed by IP PrintWay extended mode; using the Print Interface subsystem), the **lpstat** command returns status information.

The **cancel** command cancels a print job.

```
cancel jobid ...
```

The **cancel** command cancels one or more print jobs that you submitted, with these restrictions:

- You can only cancel your own jobs.



- ▶ You cannot cancel a job after it has started processing.
- ▶ In a JES3 environment, you might not be able to cancel a job that is held on the spool.

To print data sets created by batch jobs or TSO/E sessions using IP PrintWay or the Print Interface subsystem you can use JCL parameters that have special considerations.

- ▶ You can direct print output data sets using IP PrintWay to a printer or print server in your TCP/IP network or to a VTAM-controlled printer. IP PrintWay can also send an output data set to an e-mail destination.
- ▶ You can transform data from one data format to another format and print it on any printer.

IP PrintWay transmits output data sets from the JES spool to remote printers or print servers and to e-mail destinations. You define work-selection criteria for each IP PrintWay basic mode through JES initialization statements for the FSA. IP PrintWay job selection rules determine which print jobs IP PrintWay extended mode selects from the JES spool. The job selection rules are defined in the Printer Inventory.

The Print Interface subsystem transforms and prints output from z/OS applications with minimal changes to the applications' job control language (JCL). The subsystem processes the application's output, transforms it to the format required by the printer or e-mail destination, and writes it to the JES spool. From the JES spool, the transformed data can be printed on any printer or be sent to e-mail destinations.

## NetSpool users

The NetSpool component of Infoprint Server is configured such that it intercepts print data from VTAM applications, such as CICS and IMS; transforms the data streams to EBCDIC line data, PCL, PDF, or other formats that the printer accepts; and creates output data sets on the JES spool.

VTAM applications establish communication sessions with NetSpool printer logical units (LUs) instead of with SNA-network printers. Each NetSpool printer LU must be defined to VTAM as an application logical-unit (LU). NetSpool accepts the following VTAM data streams:

- ▶ SNA character string (SCS) data over an LU type 1 session
- ▶ 3270 data over an LU type 3 or LU type 0 session
- ▶ A binary data stream over an LU type 0, type 1, or type 3 session

## 1.13 Printing using LPR command

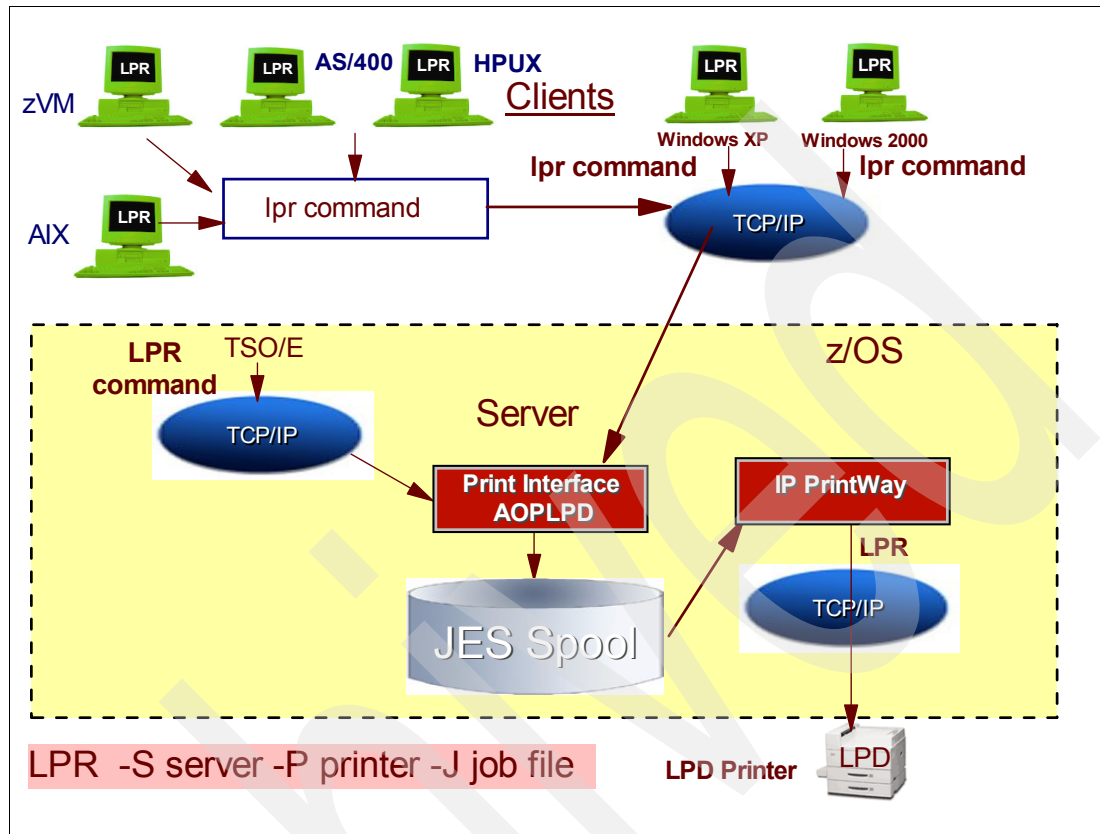


Figure 1-13 Printing using the LPR command

### Infoprint Server LPD daemon (aoplpd)

The Infoprint Server implementation uses this protocol in the following two ways:

1. Workstation client users use the **LPR** command to send print requests to an Infoprint Server. The Print Interface address space (shown in the figure as AOPLPD) acts as the LPD.

To use the **LPR** command, the administrator must configure the Infoprint Server LPD to listen at port 515. If the Infoprint Server LPD listens at a different port, use other print commands instead.

You can specify this command and these options when you print from a Windows system using Infoprint Server:

```
LPR -S server -P printer -J job file
```

The TSO/E LPR command syntax:

```
LPR data_set_name (Optional parameters
Optional parameters include: Host; Printer; Job and many others
```

2. IP PrintWay uses the line printer requester/line printer daemon (LPR/LPD) protocol to transmit data to printers in a TCP/IP network.

## 1.14 z/OS UNIX shell printing commands

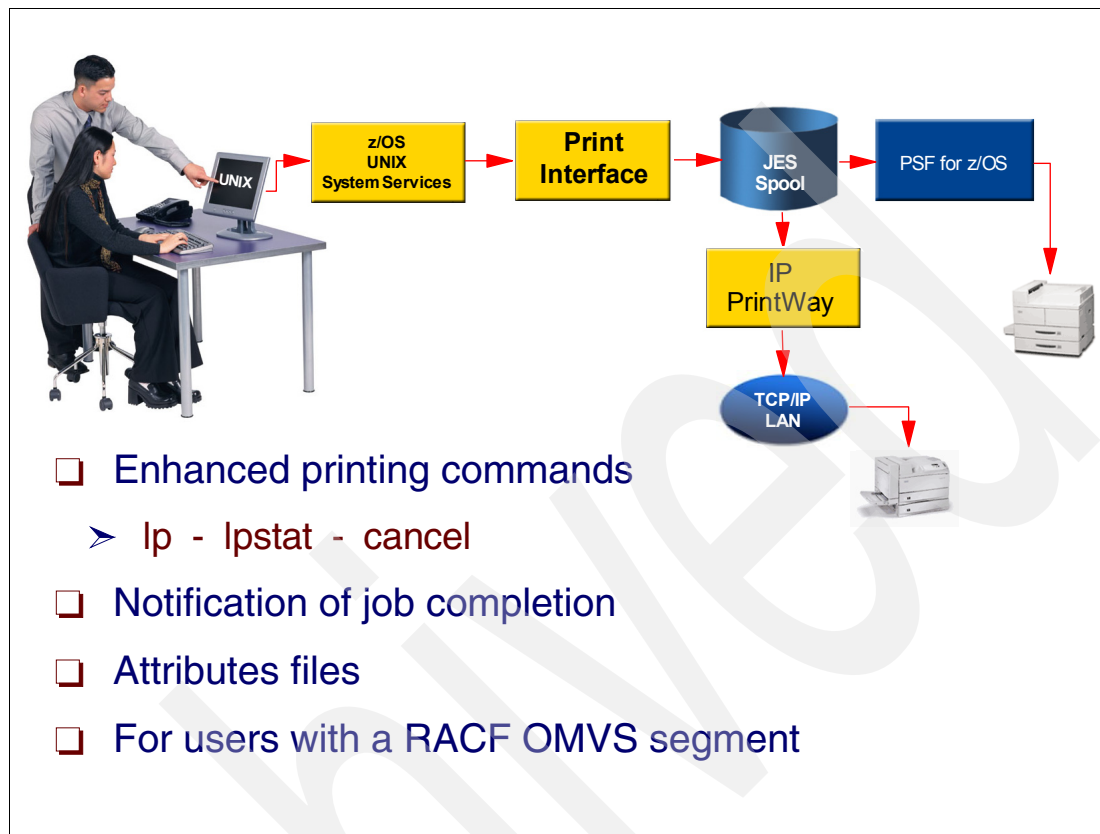


Figure 1-14 z/OS UNIX enhanced shell printing commands

### z/OS UNIX print commands

A z/OS UNIX user can print files using the Print Interface services. Print Interface provides enhanced versions of the z/OS UNIX System Services shell printing commands. These commands have more functions than the standard UNIX shell printing commands.

**Note:** These enhanced commands adhere to the XPG4.2 standard, which makes it possible to port UNIX applications to z/OS UNIX.

The z/OS UNIX shell commands available for processing of files to be printed are:

- lp** The **lp** command sends files for printing to Print Interface running on the local z/OS system. The files that a UNIX user can print are:
- HFS files that include PDF, PCL, SAP, and PS
  - MVS data sets
  - Sequential data sets
  - Partitioned data set members
- lpstat** The **lpstat** command queries the status of print jobs. It also queries the names, locations, and descriptions of printers that the administrator has defined in the Printer Inventory.
- cancel** The **cancel** command cancels print requests, provided that the data set allocated on the JES spool has not yet been selected.

## Attributes files

You can predefine attribute and value pairs in permanent files. You can use the permanent attribute files when you need those attribute values with the **lp**, **afp2pc1**, **afp2pdf**, and **afp2ps** commands. You can also use the permanent attribute files when you need to specify those values with the AOPPRINT JCL procedure and when you use the Print Interface subsystem.

For example, you could create an attributes file called **myatts** to request 5 copies of a job, simple duplex printing, and a specific output bin. Your file contains these lines:

```
# These are myatts
copies      = 5
duplex      = yes
output-bin  = collator # Collate the job
```

Use the **-o** flag to read an attributes file into the **lp** command. Use the **-j** flag to read an attributes file into the **afp2pc1**, **afp2pdf**, and **afp2ps** commands.

For example, to print a file called **myfile**, using the attributes in the **myatts** file, enter:

```
lp -o "attributes=myatts" myfile
```

This command is equivalent to this command:

```
lp -o "copies=5 duplex=yes output-bin=collator" myfile
```

## 1.15 Using Internet Printing Protocol (IPP)

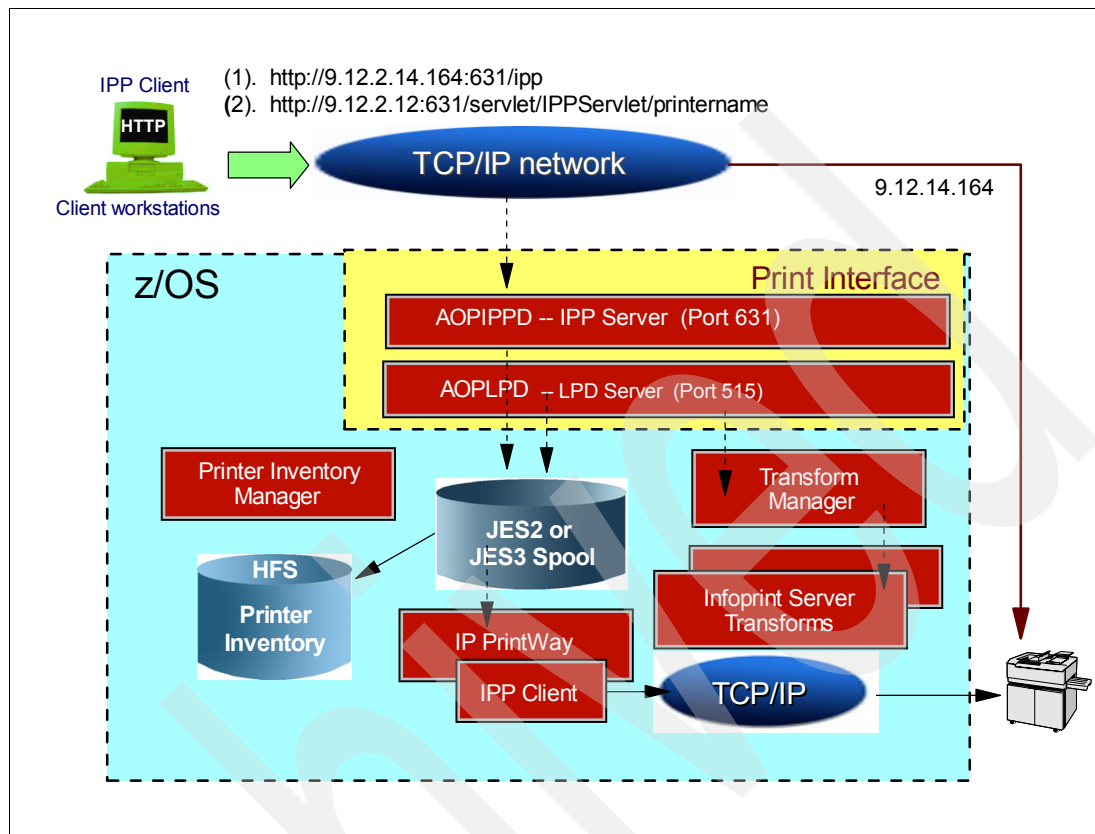


Figure 1-15 Internet Printing Protocol (IPP) support in Infoprint Server

### Internet Printing Protocol

Internet Printing Protocol (IPP) is a standard protocol for printing over the Internet. An IPP client must run in the remote system.

Infoprint Server Print Interface provides an Internet Printing Protocol Server, which accepts print requests from any client that uses the IPP protocol. The Print Interface IPP Server supports the IPP job attributes sent with the print requests. The IPP Server does not, however, support canceling a print request.

From a Windows workstation, a user submits a document for printing using the standard print-submission methods once an Internet printer has been added. You use the standard Windows procedure to add an Internet printer:

- ▶ When the Add Printer Wizard asks you whether to install the printer as a local printer or a network printer, select either **Network printer** or **A network printer**, or a **printer attached to another computer**.
- ▶ When the Add Printer Wizard asks you how to connect, select **Connect to a printer on the Internet**. Enter the URI of the printer in the URL field.
- ▶ When the Add Printer Wizard asks you whether to install the printer driver, select **OK**.

The Uniform Resource Identifier (URI) of a printer defined in the Printer Inventory has the format:

`http://host:port/servlet/IPPServlet/printername`

## 1.16 Printing with Infoprint Port Monitor for Windows

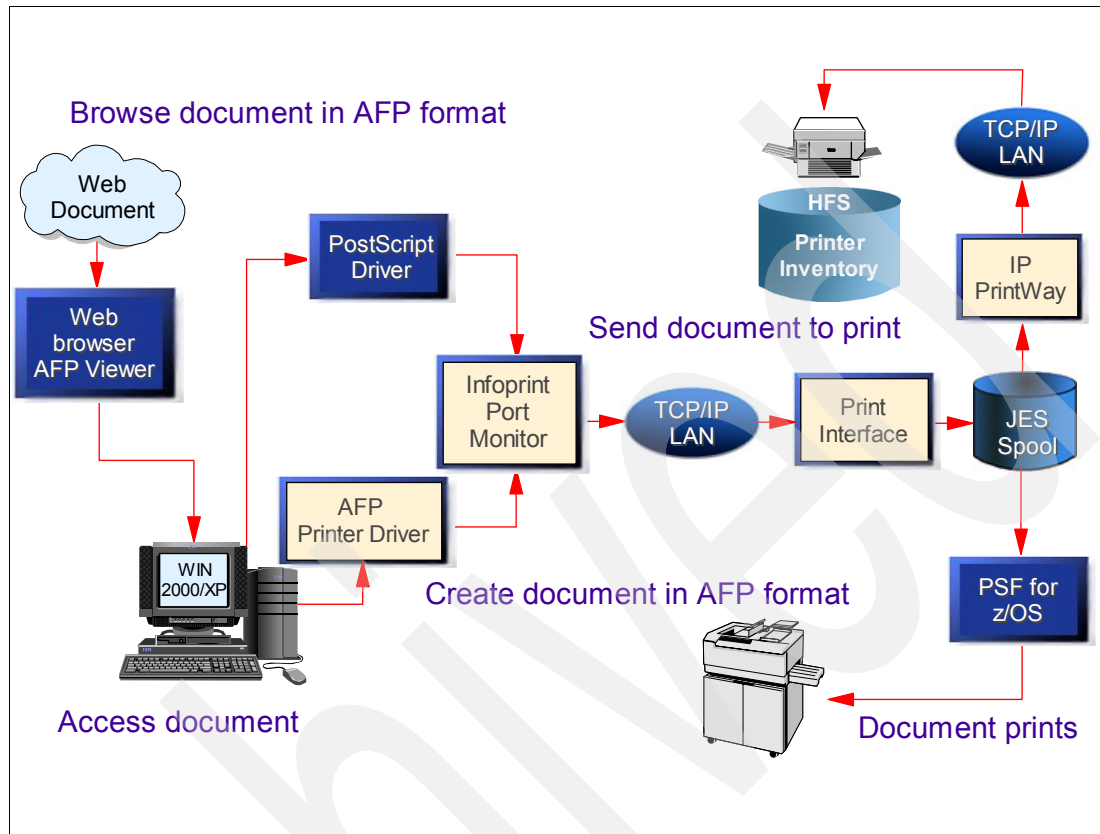


Figure 1-16 Infoprint Port Monitor for Windows component

### Infoprint Port Monitor

The Infoprint Port Monitor for Windows sends files submitted for printing from Windows 2000, Windows Me, and Windows XP systems to the Print Interface component of Infoprint Server. The Infoprint Port Monitor lets users print from Windows applications that support printing and from the Windows print command on any printer that the administrator has defined in the Printer Inventory.

**Note:** Infoprint Server also supports printing from a Windows system with the SMB protocol and the IPP protocol. To use these protocols, Windows users do not need to install the Infoprint Port Monitor for Windows.

Infoprint Server supports three client programs for Windows systems that allows Windows users to print documents on AFP printers and other z/OS printers. The three client programs for Windows workstations are:

#### 1. Infoprint Port Monitor

The Infoprint Port Monitor for Windows 95/98 and Windows NT® sends files submitted for printing from Windows applications to the Print Interface component of Infoprint Server. The Infoprint Port Monitor lets users print from Windows applications that support printing to any printer that the administrator has defined in the Printer Inventory.

The other two related programs for Windows are not part of Infoprint Server, but can be made available at no charge from the Web site.

## 2. AFP Printer Driver

The AFP Printer Driver for Windows NT and Windows 2000 creates output files in AFP format, so that Windows users can print on AFP printers. The AFP Printer Driver can create output files that contain documents, overlays, or page segments. It can also create inline form definitions for printing documents with special options, such as printing on both sides of the paper.

## 3. AFP Viewer plug-in

The AFP Viewer plug-in for Windows NT and Windows 2000 allows Windows users to view files in AFP format using an Internet browser. Users can also print from the AFP Viewer plug-in.

**Note:** The Infoprint Port Monitor for Windows is shipped with Infoprint Server. You can also download it, the AFP Printer Driver for Windows, or the AFP Viewer Plug-in for Windows directly to your Windows system from the IBM Printing Systems Web site:

<http://www.ibm.com/printers/download.html>

## LPRAFP command

You can use the **lp AFP** command to submit print requests from a Windows, HP-UX, or SunOS™ system and specify Infoprint Server job attributes. IBM provides the **lp AFP** command on an “as is” basis. You can download the **lp AFP** command at:

[www.ibm.com/printers/downloads.html](http://www.ibm.com/printers/downloads.html)

## 1.17 Printing SAP R/3 documents

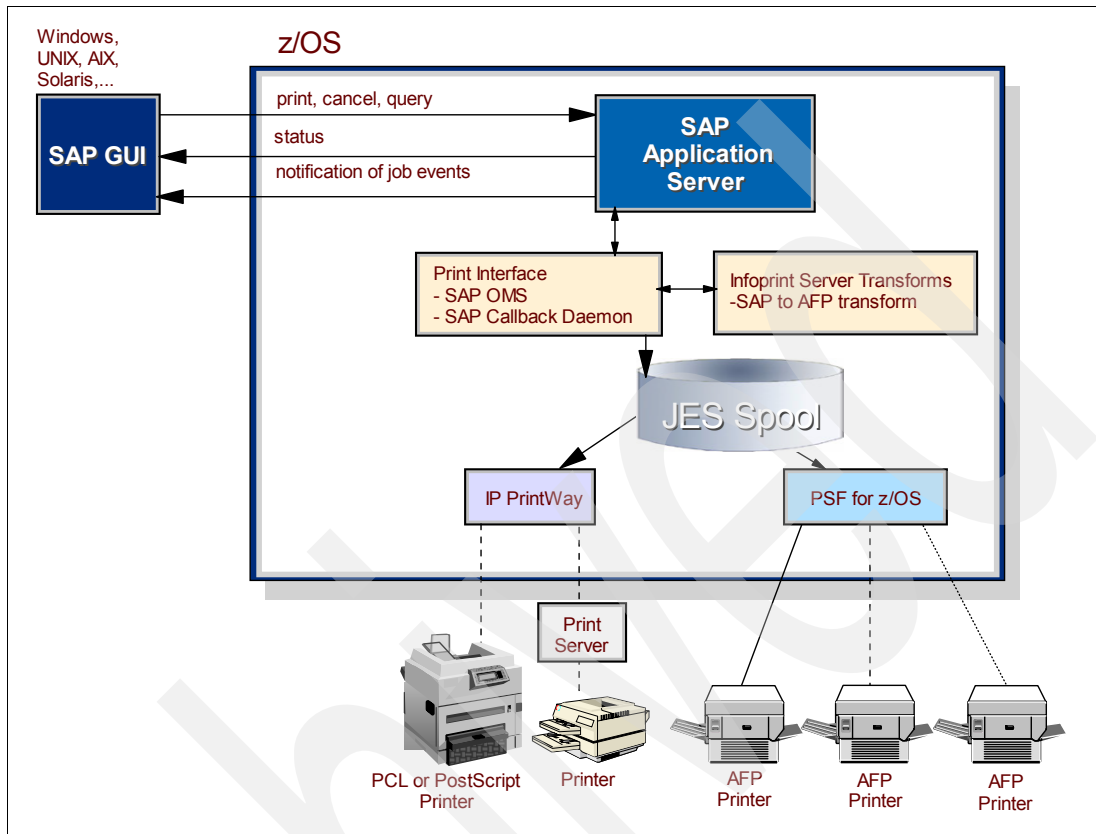


Figure 1-17 Printing SAP R/3 documents

### SAP R/3 support

Using the Infoprint Server SAP Output Management System SAP OMS and the SAP R/3 Application Server for z/OS, SAP R/3 users can submit a print job and receive immediate notification about job events:

- ▶ Using the SAP R/3 GUI, a user submits an SAP ABAP or SAP OTF document for printing on an SAP R/3 output device, which the SAP R/3 administrator has associated with a printer defined to Infoprint Server. The SAP R/3 application server sends the document to the spool work process of the SAP R/3 Application Server for z/OS. The spool work process must run on the same system as Infoprint Server.
- ▶ The SAP R/3 Application Server's spool work process submits the print request to the Infoprint Server SAP Output Management System (OMS), which is part of the Print Interface component.

**Note:** Except for JCL print submissions, the target printer is selected by specifying the printer definition name on the request. The printer definition specifies the work selection attributes that will be assigned to the spooled print data set and subsequently will be used by the IP PrintWay component.

The JCL print submission requires that all IP Printway work selection attributes must be defined correctly through JCL keyword parameters in order to get the print data selected by the IP PrintWay.



## 1.18 Infoprint Central overview

- ❑ Infoprint Central primarily for:
  - Help desk operators
  - Other authorized users or job submitters
- ❑ Only with IP PrintWay extended mode
- ❑ Single point of control for print management on z/OS
- ❑ No installation required on client workstation
- ❑ Easy to use - "Point and click"
- ❑ Reduces training required for help desk operators
- ❑ Allows end users to check status of their print jobs



Figure 1-18 Infoprint Central overview

### Infoprint Central

Infoprint Central is a Web-based print management system primarily for help desk operators that is available beginning with z/OS V1R5. However, other authorized users or job submitters can also use it. Infoprint Central works with IP PrintWay extended mode. With Infoprint Central, you can:

- ▶ Work with print jobs
- ▶ Work with printers
- ▶ Work with NetSpool logical units (LUs)
- ▶ Display printer definitions
- ▶ Check system status

### Infoprint Central users

Infoprint Central provides functions for the following types of users:

**Help desk operators** Infoprint Central helps you respond to callers' questions about where print jobs have printed and why printers are not working. For example, you can find the status of print jobs, cancel print jobs, and move print jobs. Also, you can find the status of printers, restart printers, and redirect printers to alternate printers.

<b>Printer operators</b>	Infoprint Central lets you work with printers. For example, you can start and stop printers, change the forms loaded in printers, and redirect all print jobs to a different printer.
<b>Job submitters</b>	Infoprint Central lets you find out where and when your print job printed, see if a printer is busy, and find the name of a printer in your building.

### **IP PrintWay extended mode**

When you run IP PrintWay extended mode Infoprint Central lets operators do functions that cannot be done with other tools such as Infoprint Server ISPF panels, SDSF, or JES commands. For example, you can redirect printers to alternate printers, cancel print jobs that are currently printing on IP PrintWay printers, and see all messages in the common message log for print jobs and printers. Infoprint Central does not let you work with IP PrintWay basic mode printers or print jobs.

If you run PSF for z/OS, Infoprint Central lets operators see detailed information about print jobs submitted through Infoprint Server. For example, you can see information about print jobs that are no longer on the JES spool and all messages in the common message log for print jobs. If you customize PSF to use the Printer Inventory, you can also use Infoprint Central to work with PSF printers.

You must customize and start the z/OS HTTP Server to display Infoprint Central Web pages.

## UNIX System Services overview

Infoprint Server is a UNIX application. The Infoprint Server installation and customization requires an extensive knowledge of UNIX System Services (z/OS UNIX). This chapter describes many of the basic concepts of z/OS UNIX to assist in the installation and customization.

UNIX is a highly portable operating system originally developed by Bell Laboratories that features multiprogramming in a multiuser environment. UNIX is implemented in the C language. UNIX was originally developed for use on minicomputers but has been adapted on mainframes and microcomputers. It is especially suitable for multiprocessor, graphics, and vector-processing systems. Many of the commands in the z/OS shell are based on similar commands available with UNIX System V.

z/OS UNIX System Services (z/OS UNIX) support an environment within which operating systems, servers, distributed systems, and workstations share common interfaces. z/OS UNIX supports standard application development across multivendor systems. It is required if you want to create and use applications that conform to the POSIX standard. z/OS UNIX combines the personal power of the workstation, the flexibility of open systems, and the strength of MVS. It supports and fosters a super environment of larger operating systems or servers and of distributed systems and workstations that share common interfaces. Users can switch back and forth between the traditional TSO/E interface and the shell interface. UNIX-skilled users can interact with the system, using a familiar set of standard commands and utilities. MVS-skilled users can interact with the system, using familiar TSO/E commands and interactive menus to create and manage hierarchical file system files and to copy data back and forth between MVS data sets and files. Application programmers and users have both sets of interfaces to choose from and, by making appropriate trade-offs, can choose to mix these interfaces.

z/OS UNIX (a component of the BCP FMID) provides:

- ▶ XPG4 UNIX 1995 conformance
- ▶ Assembler callable services
- ▶ TSO/E commands to manage the file system
- ▶ ISPF shell environment

z/OS UNIX interacts with the following elements and features of z/OS:

- ▶ BCP (WLM and SMF components)
- ▶ C/C++ Compiler, to compile programs
- ▶ Language Environment, to execute the shell and utilities or any other XPG4-compliant shell application
- ▶ Data Facility Storage Management Subsystem/MVS (DFSMS). HFS is a component of DFSMS.
- ▶ Security Server for z/OS. (RACF is a component of the Security Server.)
- ▶ Resource Measurement Facility (RMF)
- ▶ System Display and Search Facility (SDSF)
- ▶ Time Sharing Option Extensions (TSO/E)
- ▶ z/OS Communications Server (TCP/IP)
- ▶ ISPF, to use the dialogs for OEDIT, or ISPF/PDF for the ISPF shell
- ▶ Network File System (NFS)
- ▶ z/OS Distributed File Service zSeries File System (zFS)

## 2.1 RACF OMVS segments

User profile

Userid	Default Group	Connect Groups		TSO	DFP	OMVS		
SMITH	PROG1	PROG1	PROG2	...	...	UID	Home	Program
						15	/u/smith	/bin/sh

Group profile

Groupid	Superior Group	Connected Users				OMVS
PROG1	PROGR	SMITH	BROWN	...	...	GID
						25

Group profile (no OMVS segment)

Groupid	Superior Group	Connected Users			
PROG2	PROGR	SMITH	WHITE	...	...

OMVS segment = The portion of a RACF profile that contains OMVS logon information.

Figure 2-1 Defining user and group profiles

### OMVS segments in RACF profiles

The user profile, in computer security, contains a description of a user that includes such information as user ID, user name, password, access authority, and other attributes that are obtained when the user logs on. A group is a collection of users who can share access authorities for protected resources. A user must belong to a group.

The RACF component of the Security Server authenticates users and verifies whether they are allowed to access certain resources. An OMVS user is identified by a UID, which is kept in the RACF user profile, and a GID, which is kept in the RACF group profile. OMVS users should belong to RACF groups that have GID defined.

The RACF user profile and group profile definitions for z/OS UNIX are called OMVS segments. All users and programs that need access to z/OS UNIX must have a RACF user profile defined with an OMVS segment which has, as a minimum, a UID specified. A user without a UID cannot access z/OS UNIX.

### RACF user profile

Within this profile is an OMVS segment that defines the user as a z/OS UNIX user. The OMVS segment includes the following fields:

**UID** A number from 0 to 2147483647 that identifies a z/OS UNIX user. A z/OS UNIX user must have a UID defined.

- HOME** The name of a directory in the file system. This directory is called the home directory and becomes the current directory when the user accesses z/OS UNIX. This field is optional.
- The home directory is the current directory when a user invokes z/OS UNIX. During z/OS UNIX processing, this can be changed temporarily by using the **cd** (change directory) shell command. The command will not change the value in the RACF profile. The directory specified as home directory in the RACF profile must exist (be pre-allocated) before a user can invoke z/OS UNIX. If a home directory is not specified in RACF, the root (/) directory will be used as default.
- PROGRAM** The name of a program. This is the program that will be started for the user when the user begins a z/OS UNIX session. Usually this is the program name for the z/OS UNIX shell. This field is optional.

In addition, the user OMVS segment can contain user limit specifications for address space size, files per process, processes per UID, threads per process, memory map area size, non-shared memory size, and shared memory size.

### RACF group profile

The RACF group also has a segment called OMVS to define z/OS UNIX groups. It contains only one field:

- GID** A number from 0 to 2147483647 that identifies a z/OS UNIX group.

### Segment example

The example in Figure 2-1 on page 39 shows a user profile for TSO/E user ID SMITH, which is connected to two groups, PROG1 and PROG2. SMITH is defined as a z/OS UNIX user and has a UID. His home directory is /u/smith and he will get into the shell when he issues the OMVS command because the name of the shell, /bin/sh is specified as the program name.

A program that will access z/OS UNIX and run as a started task (for example, RMFGAT) or a daemon (for example, the *inetd* daemon, which is used for remote login (*rlogin*) to the shell via TCP/IP) must also be defined to RACF with a user profile and a UID specified. This type of user does not require a home directory or a program specified in the OMVS segment. The home directory and program are important for people's user IDs.

The RACF profile for a group is also extended with an OMVS segment. A z/OS UNIX group is a RACF group with a GID specified in the OMVS segment. The figure shows that group PROG1 is also a z/OS UNIX group with a GID value of 25. The group PROG2 does not have an OMVS segment and therefore is not a z/OS UNIX group.

## 2.2 Infoprint Server and RACF OMVS segments

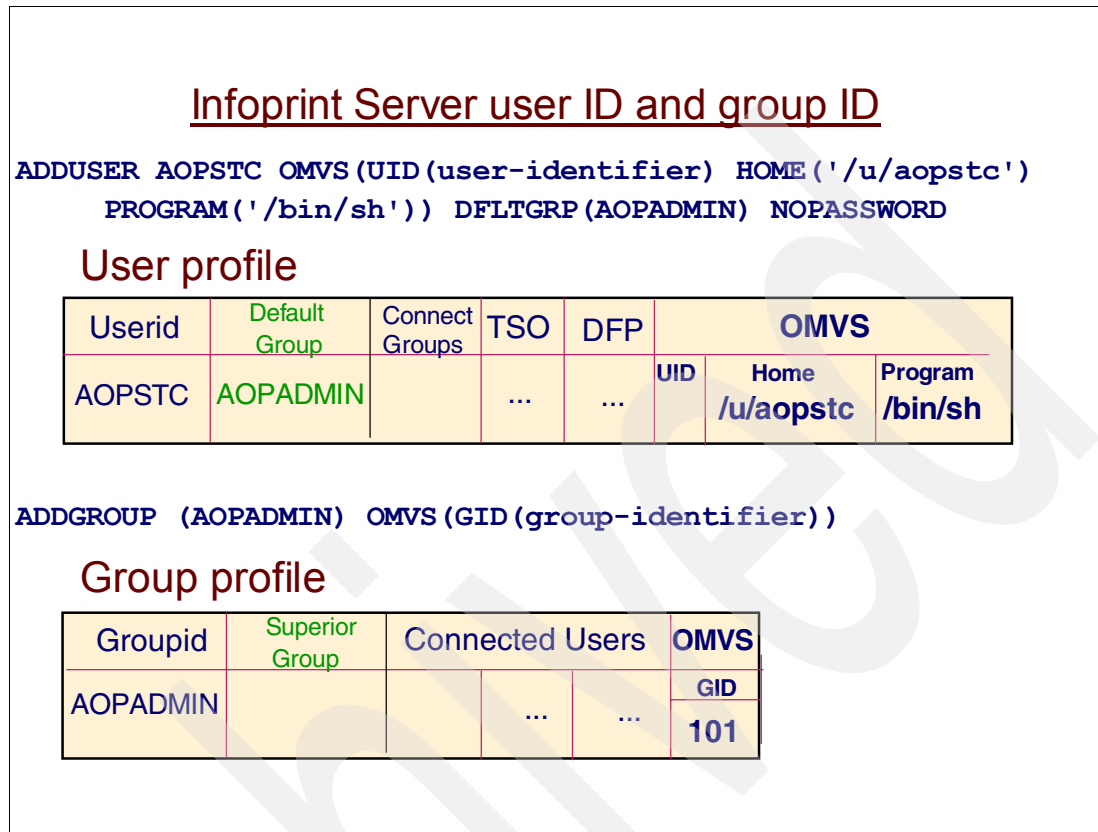


Figure 2-2 RACF user and group profiles with OMVS segments

### RACF OMVS segments

RACF OMVS segments defines a UID (z/OS UNIX user ID) for user and a GID (group ID) for each group that needs access to z/OS UNIX functions and resources.

Create a user ID with an OMVS segment, a home directory, and a default group that has an OMVS segment with a group identifier (GID). You can select any name for the user ID. Use the RACF ADDUSER command to create a new user ID. Use the RACF ALTUSER command to alter an existing user ID to give it access to z/OS UNIX. For the default group, you can select any group that has an OMVS segment and a GID. The user ID must be connected to the default group.

An Infoprint Server administrator can view and update the Printer Inventory and can view all messages in the common message log. The suggested group name for administrators is AOPADMIN. However, you can use any name. The group profile must have an OMVS segment and an OMVS group identifier (GID), as shown in Figure 2-2.

For example, this RACF command defines user ID AOPSTC with a default group of AOPADMIN. For user-identifier, specify an integer that is different from other UIDs in your installation. The NOPASSWORD operand defines the user ID as a protected user ID. Protected user IDs cannot log on to the z/OS system, and they cannot be revoked due to incorrect password attempts.

```
ADDUSER AOPSTC OMVS(UID(user-identifier) HOME('/u/aopstc')  
PROGRAM('/bin/sh')) DFLTGRP(AOPADMIN) NOPASSWORD
```

## 2.3 z/OS UNIX users and groups

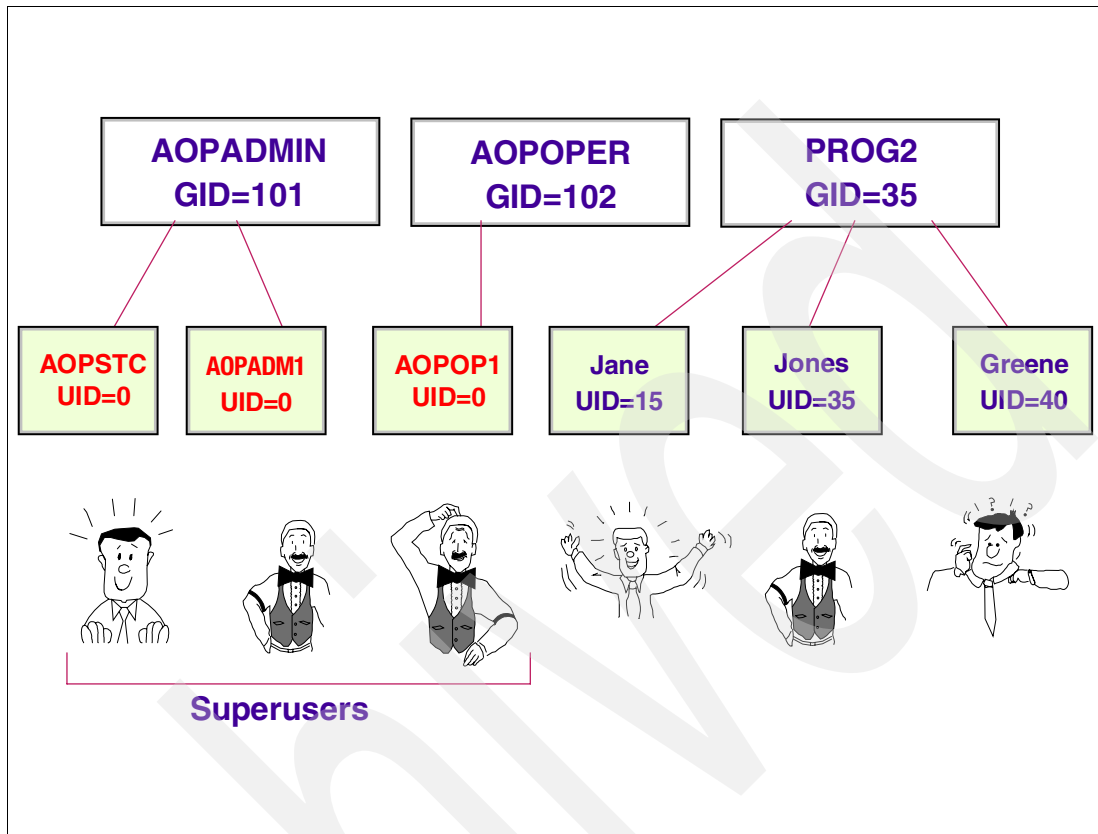


Figure 2-3 z/OS UNIX users and groups

### z/OS UNIX users and groups

z/OS UNIX users must have RACF user profiles defined with an OMVS segment.

Similar to users in a UNIX system, z/OS UNIX users are identified by a UID (user identification). The UID has a numerical value.

Types of users by UID:

- ▶ User (regular user)
  - Identified by a non-zero UID.
- ▶ Superuser (authorized/privileged user). A superuser can be any of the following:
  - A z/OS UNIX user with a UID=0.
  - A started procedure with a trusted or privileged attribute in the RACF started procedures table.
  - A z/OS UNIX user that has READ authority to the BPX.SUPERUSER profile in the RACF FACILITY class.

The concept of *superusers* comes from UNIX. Sometimes it is also referred to as *root* authority.



## z/OS UNIX superusers

A superuser can:

- ▶ Pass all z/OS UNIX security checks, so that the superuser can access any file in the hierarchical file system. A superuser does not get any additional authorities to access MVS/ESA™ resources. The authority is limited to the z/OS UNIX component.
- ▶ Manage z/OS UNIX processes and files.
- ▶ Have an unlimited number of processes running concurrently. For a started procedure, this is true only if it has a UID of 0. It is not true for a trusted or privileged process with a different UID.
- ▶ Change identity from one UID to another.
- ▶ Use `setrlimit` to increase any of the system limits for a process.

A superuser is usually a system administrator, or it can be a started procedure which is authorized by the RACF started procedures table or the RACF STARTED class.

z/OS UNIX users belong to one or more groups in the same way as TSO/E users belong to groups. An z/OS UNIX group is a RACF group with a GID (group ID) defined. The GID has a numerical value.

**Note:** Multiple users may have the same UID.

## Infoprint Server AOPADMIN and AOOPER groups

You should define these groups to RACF to protect access to the Printer Inventory, the common message log, and Infoprint Server operator commands:

- ▶ **AOPADMIN:** This group is for Infoprint Server administrators, who can view and update the Printer Inventory and view all messages in the common message log.
- ▶ **AOOPER:** This group is for Infoprint Server operators, who can start and stop Infoprint Server daemons.

**Note:** You can choose any names for these groups. If you do not want to distinguish between administrators and operators, you need to define only one group.

You must run the **aopsetup** shell script to set up the correct z/OS UNIX permissions for Infoprint Server directories and executable files before you start Infoprint Server daemons for the first time. Also, you must rerun **aopsetup** whenever you move to a new z/OS release.

The **aopsetup** shell script requires two positional arguments:

```
aopsetup operator-group administrator-group
```

*operator-group* is the name of the RACF group you created for Infoprint Server operators. The suggested RACF group name is AOOPER. However, you might have used a different name for this group. This argument is required.

*administrator-group* is the name of the RACF group you created for Infoprint Server administrators. The suggested RACF group name is AOPADMIN. However, you might have used a different name for this group. This argument is required.

## 2.4 RACF commands to define groups

Add OMVS segment to existing group AOPADMIN:

```
ALTGROUP AOPADMIN OMVS (GID (101) )
```

Define a new group PROG1:

```
ADDGROUP PROG1 OMVS (GID (25) )
```

List the OMVS segment of group AOPADMIN:

```
LG AOPADMIN NORACF OMVS  
  
INFORMATION FOR GROUP AOPADMIN  
  
OMVS INFORMATION  
-----  
GID= 0000000101
```

Figure 2-4 Defining group profiles with RACF commands

### RACF commands for group profiles

The **ALTGROUP**, **ADDGROUP**, and **LISTGRP** commands have keywords for administering the OMVS segment.

- ▶ Use **ALTGROUP** (**ALG**) to modify an existing RACF group with or without an OMVS segment. The OMVS segment can be added, modified or deleted.
- ▶ Use **ADDGROUP** (**AG**) to define a new RACF group with or without an OMVS segment. The OMVS keyword can be used to define this group as a z/OS UNIX group.
- ▶ The **LISTGRP** (**LG**) command will display the OMVS segment if the OMVS keyword is specified.

A z/OS UNIX group is a RACF group with an OMVS segment and a GID defined. Figure 2-4 shows examples of how to use the RACF commands to add and change the OMVS segment for a group.

**ALTGROUP NOOMVS** requests RACF to delete the OMVS segment from the specified group's profile.

A user can belong to (or be connected to) multiple groups. A z/OS UNIX user must belong to at least one z/OS UNIX group. It is not necessary that all the groups a z/OS UNIX user belongs to are defined as z/OS UNIX groups. Only the groups that are defined as z/OS UNIX groups will be used for authorization checking in z/OS UNIX.

## 2.5 RACF commands to define users

Add OMVS segment to existing user JANE:

```
ALTUSER JANE +  
    OMVS (UID(15) HOME('/') PROGRAM('/bin/sh'))
```

Define a new user AOPADM1:

```
ADDUSER AOPADM1 DFLTGRP(AOPADMIN) +  
    OMVS (UID(0) HOME('/u/aopadm1') PROGRAM('/bin/sh'))
```

List the OMVS segment of user AOOPER:

```
LU AOPADM1 OMVS NORACF  
USER=AOPADM1  
OMVS INFORMATION  
-----  
UID= 0000000000  
HOME= /u/aopadm1  
PROGRAM= /bin/sh  
CPUTIMEMAX= NONE  
ASSIZEMAX= NONE  
FILEPROCMAx= NONE  
PROCUSERMAX= NONE  
THREADSMAX= NONE  
MMAPAREAMAX= NONE
```

Note: UID=0 is a UNIX superuser

Connect user AOPADM1 to group SYS1:

```
CO AOPADM1 GROUP(SYS1)
```

Figure 2-5 Defining user profiles with RACF commands

### RACF commands for user profiles

The RACF commands **ALTUSER**, **ADDUSER** and **LISTUSER** have keywords for administering the OMVS segment, as follows:

- ▶ Use the **ALTUSER (ALU)** command to change the definitions for an existing TSO/E user ID. An OMVS segment can be added, modified, or deleted.
- ▶ Use the **ADDUSER (AU)** command to define a new TSO/E user ID with or without an OMVS segment.
- ▶ The **LISTUSER (LU)** command is used for listing the definitions for a TSO/E user ID. When the OMVS keyword is specified, the values specified in the OMVS segment will also be listed.

### ADDUSER (AU) command

The **ADDUSER** command sets up a new user, whereas **ALTUSER** modifies an existing user. To remove a specification in the OMVS segment, use the keywords **NOUID** (to remove UID), **NOHOME** (to remove home directory definition), or **NOPROGRAM** (to remove program definition). The **ALU NOOMVS** keyword will remove all the definitions in the OMVS segment.

### DFLTGRP parameter

The **DFLTGRP** parameter on the **ALU** and **AU** command specifies the name of a RACF-defined group to be used as the default group for the user. If you do not specify a group, RACF uses your current connect group as the default.

## **CONNECT command**

The **CONNECT** RACF command connects a user to a group, modify a user's connection to a group, or assign the group-related user attributes. If you are creating a connection, defaults are available for each operand. If you are modifying an existing connection, no defaults apply.

## **OMVS keyword**

The OMVS keyword must be used on the **LISTUSER** command if you want the OMVS segment definitions to be displayed.

## **ISHELL command**

The z/OS UNIX ISPF shell (ISHELL) provides some menus for administering user IDs. However, before a user can use the **ISHELL** command, he must have a user ID with a UID defined (and a group with a GID defined). This must be done using the RACF commands or the ISPF RACF dialog.

## 2.6 Superusers with appropriate privileges

- ❑ 3 ways to assign superuser authority
  - Assigning a UID of 0, which is the least desirable way
    - Okay for special administrators
  - Using the BPX.SUPERUSER resource in the RACF FACILITY class
  - Using the UNIXPRIV class profiles
    - The preferred way

Figure 2-6 Assigning superuser authority to users

### Defining superuser authority

When you are defining z/OS UNIX users, you might want to define some of them with appropriate superuser privileges. There are three ways of assigning superuser authority:

- ▶ Assigning a UID of 0, which is not the recommended way
- ▶ Using the BPX.SUPERUSER resource in the FACILITY class.
- ▶ Using the UNIXPRIV class profiles, the preferred way.

### UID of 0

While some functions require a UID of 0, in most cases you can choose among the three ways. When choosing among them, try to minimize the number of user IDs (as opposed to started procedures) with a UID(0) superuser authority.

To summarize the choices, UID(0) gives you access to all UNIX functions and resources, as is true for all UNIX systems. However, in z/OS, RACF allows certain users to perform specific privileged functions without being defined as UID(0).

### BPX.SUPERUSER

BPX.SUPERUSER allows you give each user a unique UID and have them use the **su** command to obtain the authority they need. You can give them the ability to use the **su** command by giving them READ authority to the BPX.SUPERUSER resource in the FACILITY class.

The **su** command changes the user ID associated with a session. **su** starts a new shell and lets you operate in it with the privileges of a superuser or another user.

If you do not specify a user ID, **su** changes your authorization to that of the superuser. If you specify a user ID, **su** changes your authorization to that of the specified user ID. The new environment is built and then a new session is initiated. The new session is run as a child shell of the shell issuing the **su** command.

## UNIXPRIV class

You can define profiles in the UNIXPRIV class to grant RACF authorization for certain z/OS UNIX privileges. These privileges are automatically granted to all users with z/OS UNIX superuser authority. By defining profiles in the UNIXPRIV class, you may specifically grant certain superuser privileges with a high degree of granularity to users who do not have superuser authority. This allows you to minimize the number of assignments of superuser authority at your installation and reduces your security risk.

Resource names in the UNIXPRIV class are associated with z/OS UNIX privileges. You must define profiles in the UNIXPRIV class protecting these resources in order to use RACF authorization to grant z/OS UNIX privileges. The UNIXPRIV class must be active and SETROPTS RACLIST must be in effect for the UNIXPRIV class. Global access checking is not used for authorization checking to UNIXPRIV resources.

The following lists UNIXPRIV class resource names:

- ▶ CHOWN.UNRESTRICTED
- ▶ FILE.GROUPOWNER.SETGID
- ▶ RESTRICTED.FILESYS.ACCESS
- ▶ SHARED.IDS
- ▶ SUPERUSER.FILESYS.ACLOVERRIDE
- ▶ SUPERUSER.FILESYS
- ▶ SUPERUSER.FILESYS.CHANGEPERMS
- ▶ SUPERUSER.FILESYS.CHOWN
- ▶ SUPERUSER.FILESYS.MOUNT
- ▶ SUPERUSER.FILESYS.QUIESCE
- ▶ SUPERUSER.FILESYS.PFSCTL
- ▶ SUPERUSER.FILESYS.VREGISTER
- ▶ SUPERUSER.IPC.RMID
- ▶ SUPERUSER.PROCESS.GETPSENT
- ▶ SUPERUSER.PROCESS.KILL
- ▶ SUPERUSER.PROCESS.PTRACE
- ▶ SUPERUSER.SETPRIORITY

See *z/OS UNIX System Services Planning*, GA22-7800 for the details of the resource names available in the UNIXPRIV class, the z/OS UNIX privilege associated with each resource, and the level of access required to grant the privilege.

## 2.7 BPX.SUPERUSER authority

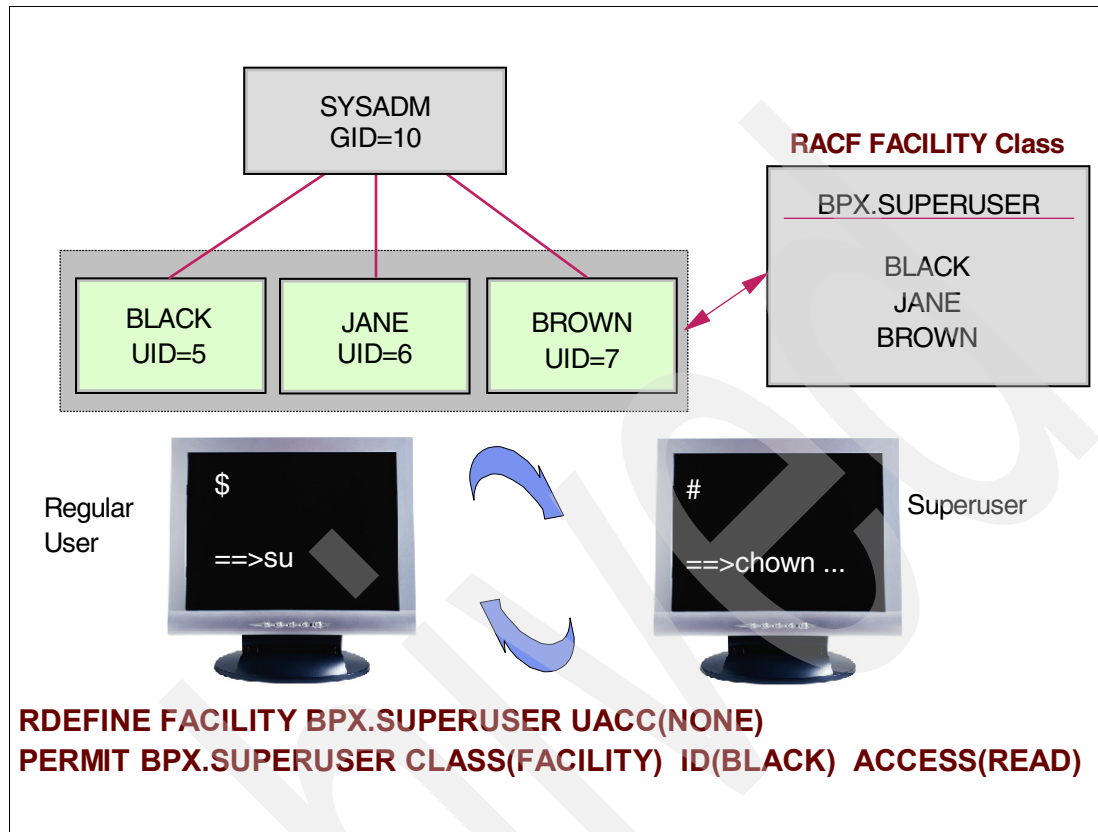


Figure 2-7 Assigning superuser authority with BPX.SUPERUSER

### BPX.SUPERUSER authority

A user with a non-zero UID can be granted superuser authority through the RACF FACILITY class **BPX.SUPERUSER** profile. Users with this authority are able to temporarily switch to become superuser when this authority is required for administrative tasks. These users can use any of the following methods to switch to superuser:

- ▶ In the z/OS UNIX shell, use the command **su** (switch user). This command will create a subshell where the user will have superuser authority and authorized commands can be executed. When the subshell session is ended, the user will return to the first shell session as a regular user.
- ▶ Use the **ISHELL** command to enter the z/OS UNIX ISPF Shell. Select the option (from action bar **Setup** option **7 Enable superuser mode(SU)**) to switch to superuser state. The user will have superuser authority until he or she exits the ISHELL.
- ▶ After gaining superuser authority in the ISHELL, the user can do a split screen in ISPF and enter the OMVS command. The z/OS UNIX shell that is started will inherit the superuser authority set up in the ISHELL.

You can also define a user called **BPXROOT** with an OMVS segment. Specify UID=0, a home directory of / (root), and the program /bin/sh. **BPXROOT** should not have any special permission to MVS resources. This user ID will be used in rare cases where a daemon process tries to change the identity of a process to superuser but does not know the MVS identity of the process. **BPXROOT** is the default name. A different name can be used, but then the installation has to change the specification in the BPXPRMxx parmlib member: **SUPERUSER(userid)**.

## 2.8 SUPERUSER.FILESYS profile

❑ Grant authorization for certain UNIX privileges

```
RDEFINE UNIXPRIV SUPERUSER.FILESYS UACC(NONE)
PERMIT SUPERUSER.FILESYS CLASS(UNIXPRIV) ID(user|group) ACC(READ)
```

- SUPERUSER.FILESYS - ACC(.....)
  - **READ** - Allows a user to read any local file, and to read or search any local directory
  - **UPDATE** - Allows a user to write to any local file, and includes privileges of READ access
  - **CONTROL/ALTER** - Allows a user to write to any local directory, and includes privileges of UPDATE access

Figure 2-8 Granting superuser authority for file system access with UNIXPRIV

### UNIXPRIV class SUPERUSER.FILESYS profile authority

The UNIXPRIV class provides the capability to assign specific superuser functions to a user or group when you give a user or group either a:

- ▶ UID of 0
- ▶ BPX.SUPERUSER profile

Instead of giving a user or group access to all functions as a superuser, the UNIXPRIV class provides profiles that allows access to a specific superuser function.

The **SUPERUSER.FILESYS** profile in the UNIXPRIV class has three access levels that allows access to z/OS UNIX files:

- |                      |  |
|----------------------|--|
| <b>READ</b>          | Allows a user to read any local file, and to read or search any local directory.         |
| <b>UPDATE</b>        | Allows a user to write to any local file, and includes privileges of READ access.        |
| <b>CONTROL/ALTER</b> | Allows a user to write to any local directory, and includes privileges of UPDATE access. |

The **SUPERUSER.FILESYS.CHANGEPERMS** profile allows users to use the **chmod** command to modify the permission bits that are used to control the owner access, group access, and general access to a file and to use the **setfac1** command. You can use it to set flags that modify the user ID (UID) and group ID (GID) of the file when it is executed. You can



also use it to set the sticky bit to indicate from where the file should be fetched. You identify the file by its pathname.

The **SUPERUSER.FILESYS.CHOWN** profile allows a user to use the **chown** command to change the owner or group of a file or directory.

The **SUPERUSER.FILESYS.MOUNT** profile has two access levels:

**READ** Allows user to issue the TSO/E MOUNT command or the mount shell command with the nosetuid option. Also allows users to unmount a file system with the TSO/E UNMOUNT command or the unmount shell command mounted with the nosetuid option.

Users permitted to this profile can use the chmount shell command to change the mount attributes of a specified file system.

**UPDATE** Allows user to issue the TSO/E MOUNT command or the mount shell command with the setuid option. Also allows user to issue the TSO/E UNMOUNT command or the unmount shell command with the setuid option.

Users permitted to this profile can issue the chmount shell command on a file system that is mounted with the setuid option.

The **SUPERUSER.FILESYS.QUIESCE** profile allows user to issue **quiesce** and **unquiesce** commands for a mounted file system. The quiesce service makes the files in a file system unavailable for use. After the file system is quiesced, the system can back up the data in it.

The unquiesce service unquiesces a file system, making the files in it available for use again.

## 2.9 Assigning UIDs

### ❑ **SHARED.IDS** profile UNIXPRIV class

- Acts as a system-wide switch to prevent assignment of an ID which is already in use

### ❑ Enable **shared UID** prevention

- **RDEFINE UNIXPRIV SHARED.IDS UACC(NONE)**
- **SETROPTS RACLIST(UNIXPRIV) REFRESH**

### ❑ **Examples:**

- `ADDUSER MARCY OMVS(UID(12))`  
**IRR52174I Incorrect UID 12. This value is already in use by BRADY.**  
`ADDUSER (HARRY MARY) OMVS(UID(14))`  
**IRR52185I The same UID cannot be assigned to more than one user.**

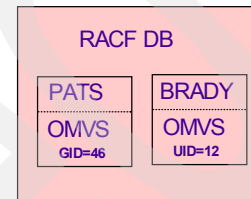


Figure 2-9 Assigning UIDs with shared UID prevention

### Assigning UIDs

z/OS UNIX allows multiple users to have the same UID. Assigning the same UID to multiple user IDs allows each user to access all of the resources associated with the other users of that shared user ID. The shared access includes not only z/OS UNIX resources such as files, but also includes the possibility that one user could access z/OS resources of the other user that are normally considered to be outside the scope of z/OS UNIX.

However, you may want to assign the same UID to multiple user IDs if these user IDs are used by the same person or persons. It may also be necessary to assign multiple users a UID of 0 (superuser authority). When doing this, it is important to remember that a superuser is implicitly a trusted user who has the potential of using UID(0) to access all z/OS resources.

**Note:** If the SHARED.IDS profile is defined in the UNIXPRIV class, in order to assign a UID that is already in use to another user ID you must specify the SHARED keyword with the UID keyword on the RACF ADDUSER or ALTUSER command.

If the SHARED.IDS profile is defined in the UNIXPRIV class, you may need to use the SHARED keyword because UID(0) is likely to be used by several IDs. For example:

```
ALTUSER SMORG OMVS(UID(0) SHARED HOME('/') PROGRAM('/bin/sh'))
```

When assigning a UID to a user, also make sure that the user is connected to at least one group that has an assigned GID. This group should be either the user's default group or one that the user specifies during logon or on the batch job. A user with a UID and a current connect group with a GID can use z/OS UNIX functions and access z/OS UNIX files based on the assigned UID and GID values. If a UID and a GID are not available as described, the user cannot use z/OS UNIX functions.

### Shared UID prevention option

In order to prevent several users from having the same UID number, a new RACF SHARED.IDS profile has been introduced in the UNIXPRIV class. This new profile acts as a system-wide switch to prevent assignment of an UID which is already in use. The use of the SHARED.IDS profile requires AIM stage 2 or 3. To enable shared UID prevention, it is necessary to define a new SHARED.IDS profile in the UNIXPRIV class, as follows:

```
RDEFINE UNIXPRIV SHARED.IDS UACC(NONE)
SETROPTS RACLIST(UNIXPRIV) REFRESH
```

### SHARED.IDS examples

Once the SHARED.IDS profile has been defined and the UNIXPRIV class refreshed, it will not allow a UID to be assigned if the UID is already in use.

As shown in Figure 2-9 on page 52, UID 12 is not assigned to user MARCY because in the RACF database this UID is assigned to user BRADY. Also, users HARRY and MARY cannot be assigned the same UID 14.

The same is true for GIDs; it will not allow a GID to be shared between different groups.

**Note:** The use of this functionality does not affect pre-existing shared UIDs. They remain as shared once you install the new support. If you want to eliminate sharing of the same UID, you must clean them up separately. RACF provides an IRRICE report to find the shared UIDs.

## 2.10 Shared UID prevention

### ❑ **SHARED keyword**

- **OMVS segment of the ADDUSER, ALTUSER, ADDGROUP, and ALTGROUP commands**

### ❑ **Example:**

```
PERMIT SHARED.IDS CLASS(UNIXPRIV) ID(UNIXGUY) ACC(READ)
SETROPTS RACLIST(UNIXPRIV) REFRESH
```



**UNIXGUY**

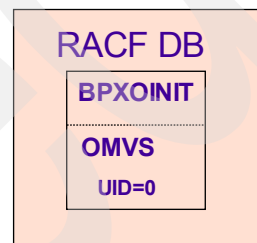
```
AU OMVSKERN OMVS(UID(0) SHARED)
AG (G1 G2 G3) OMVS(GID(9) SHARED)
```



**HARRY**

```
AU MYBUDDY OMVS(UID(0) SHARED)
```

**IRR52175I You are not authorized to specify the SHARED keyword.**



To specify the **SHARED** operand, you must have the **SPECIAL** attribute or at least **READ** authority to the **SHARED.IDS** profile in the **UNIXPRIV**

Figure 2-10 Using the SHARED keyword to allow duplicate UID assignment

### Allowing duplicate user IDs

You may want to assign the same UID to multiple user IDs if these user IDs are used by the same person or persons. It may also be necessary to assign multiple users a UID of 0 (superuser authority). When doing this, it is important to remember that a superuser is implicitly a trusted user who has the potential of using UID(0) to access all z/OS resources.

Even if the SHARED.IDS profile is defined, you may still require some UIDs to be shared and others not to be shared. For example, you may require multiple superusers with a UID(0). It is possible to do this using the new SHARED keyword in the OMVS segment of the **ADDUSER**, **ALTUSER**, **ADDGROUP**, and **ALTGROUP** commands.

To allow an administrator to assign a non-unique UID or GID using the SHARED keyword, you must grant that administrator at least READ access to the SHARED.IDS profile and be at the z/OS V1R4 level or above, as follows:

```
PERMIT SHARED.IDS CLASS(UNIXPRIV) ID(UNIXGUY) ACCESS(READ)
SETROPTS RACLIST(UNIXPRIV) REFRESH
```

Once user ID UNIXGUY has at least READ access to the SHARED.IDS profile, UNIXGUY will be able to assign the same UID or GID to multiple users, using the SHARED KEYWORD, as follows: **ADDUSER OMVSKERN OMVS(UID(0) SHARED)**

**Note:** To specify the SHARED operand, you must have the SPECIAL attribute or at least READ authority to the SHARED.IDS profile in the UNIXPRIV class.

## 2.11 Initializing z/OS UNIX

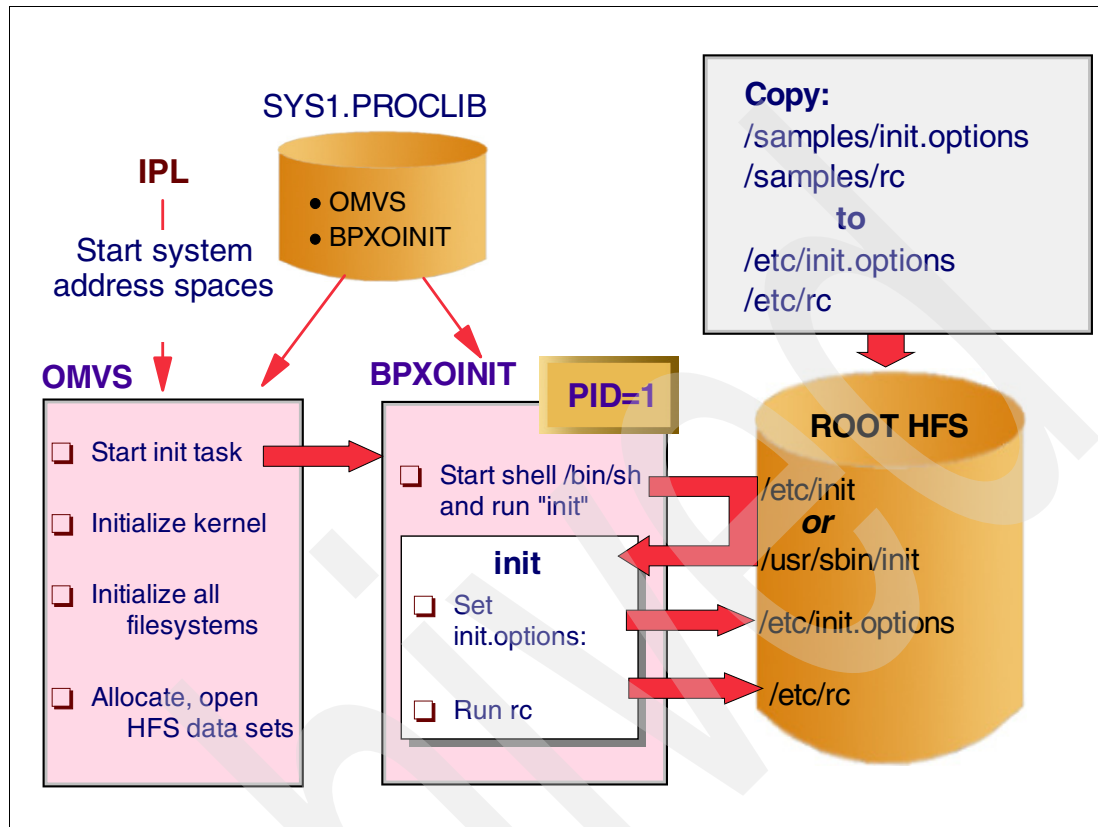


Figure 2-11 Initializing z/OS UNIX

### z/OS UNIX initialization

When a z/OS UNIX OMVS address space is started, its initialization includes running the /etc/init program. The file /etc/init is referred to as the initialization program that is run when the z/OS UNIX component is started, even though the /usr/sbin/init file may really be run.

z/OS UNIX attempts to run the program /etc/init. If no such program is found, z/OS UNIX attempts to run /usr/sbin/init. This file contains the default initialization program shipped with the z/OS UNIX Shell and Utilities.

The /etc/init program invokes a shell to execute an initialization shell script that customizes the environment for shell users. When this shell script finishes or a time interval established by /etc/init expires, z/OS UNIX becomes available for general batch and interactive use.

An option to use a REXX™ exec in an MVS data set is provided as an alternative to writing a customized /etc/init initialization program. To activate the REXX exec for initialization, you must specify its name on the STARTUP\_EXEC statement in the BPXPRMxx parmlib member.

The /usr/sbin/init program invokes a shell to execute an initialization shell script that customizes the environment for shell users. When this shell script finishes or when a time interval established by /usr/sbin/init expires, kernel services become available for general batch and interactive use.

*Kernel* is the part of z/OS UNIX that contains programs for such tasks as I/O, management, and control of hardware and the scheduling of user tasks.

## 2.12 z/OS UNIX File Security Packet

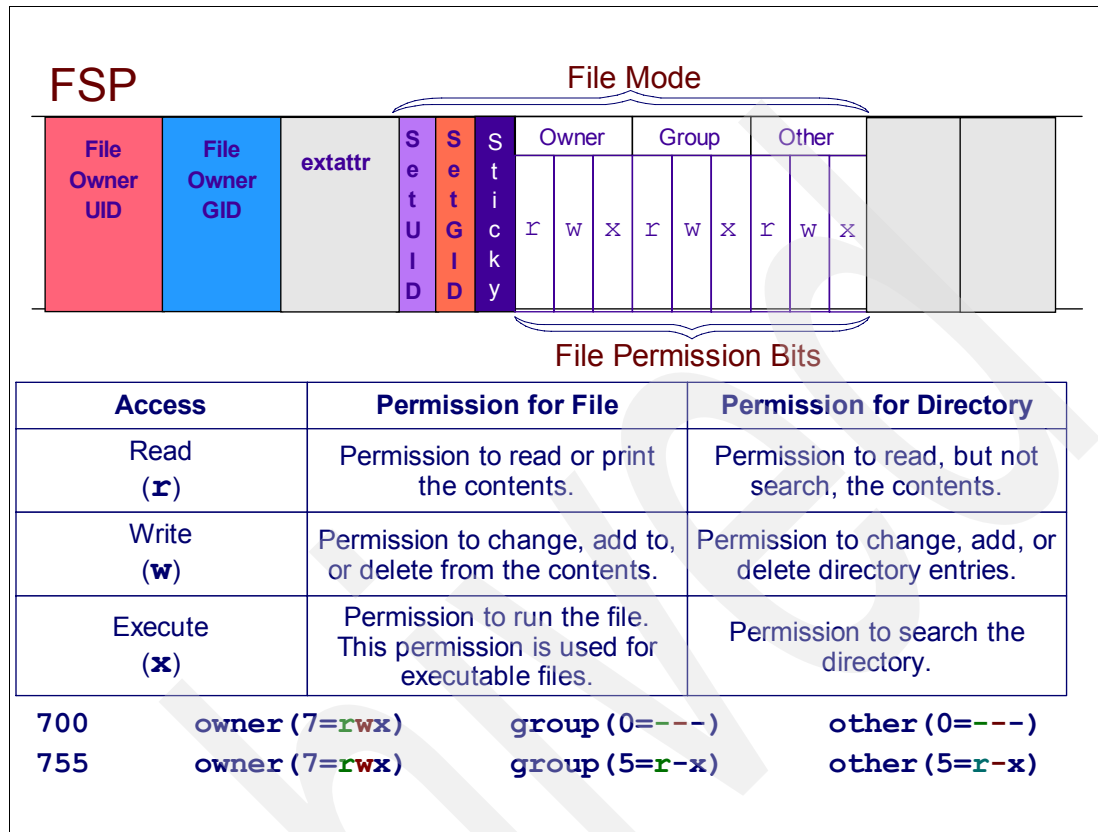


Figure 2-12 File Security Packet (FSP) used by RACF for authorization

### File security packet (FSP)

Each z/OS UNIX file and directory has a file security packet (FSP) associated with it to control access. The FSP is created when a file or directory is created, and is stored in the file system for the life of the file/directory, until the file/directory is deleted, at which time the FSP is also deleted.

The FSP consists of.

- ▶ File owner UID
- ▶ File owner GID
- ▶ File mode

### File Mode

The file mode consists of.

- SetUID** This bit only relates to executable files. If on, it causes the UID of the user executing the file to be set to the file's UID.
- SetGID** This bit only relates to executable files. If on, it causes the GID of the user executing the file to be set to the file's GID.
- Sticky Bit** This bit only relates to executable files. If on, it causes the file to be retained in memory for performance reasons. The implementation of this varies between platforms. In z/OS UNIX, it means programs are loaded from LPA (or LNKST as per normal MVS program search) instead of a HFS file. For a directory, the sticky

bit causes UNIX to permit files in a directory or subdirectories to be deleted or renamed only by the owner of the file, or by the owner of the directory, or by a superuser.

## File mode permission bits

The file mode also has the file permission bits, consisting of:

- ▶ Owner read/write/execute permissions
- ▶ Group read/write/execute permissions
- ▶ Other (or all users) read/write/execute permissions

Where:

- r** Read (r) access to both files and directories
- w** Write (w) access to both files and directories
- x** Execute (x) has a different meaning for files and directories, as follows:
  - ▶ For an executable file, an access of x means that the user can execute the file.
  - ▶ For a directory, an access of x means the user can search the directory.

Both read (r) and execute (x) are required in order to execute a shell script. In order to access HFS files, a user needs the following:

- ▶ Search (x) permission to all the directories in the pathname of files the user wants to access.
- ▶ Write permission to directories where the user will be creating new files and directories.
- ▶ Read and/or write permission, as appropriate, to files for access.
- ▶ Execute (x) permission for an executable file.

**Note:** In z/OS UNIX, these three permissions are not hierarchical. For example, a user with write permission who does **not** have read permission, can only write over existing data or add data to a file, and cannot look at the contents of the file or print the file. Similarly, write and read permission does not allow a user to execute a file or search a directory.

## Extended attributes

Another section of the FSP, which is specific to the z/OS UNIX implementation, is called Extended Attributes (extattr), which contains flags to mark HFS program files as APF-authorized and program controlled. A shell command called **extattr** is used to manipulate these bits.

## Access control lists (ACLs)

You can use access control lists (ACLs) to control access to files and directories by individual user (UID) and group (GID). ACLs are used in conjunction with permission bits. They are created, modified, and deleted using the **setfac1** shell command. To display them, use the **getfac1** shell command. You can also use the ISHELL interface to define and display ACLs.

The HFS, zFS, and TFS file systems support ACLs. The ACLs are created and checked by RACF, not by the kernel or file system. If a different security product is being used, you must check their documentation to see if ACLs are supported and what rules are used when determining file access.

## 2.13 z/OS UNIX interactive interfaces

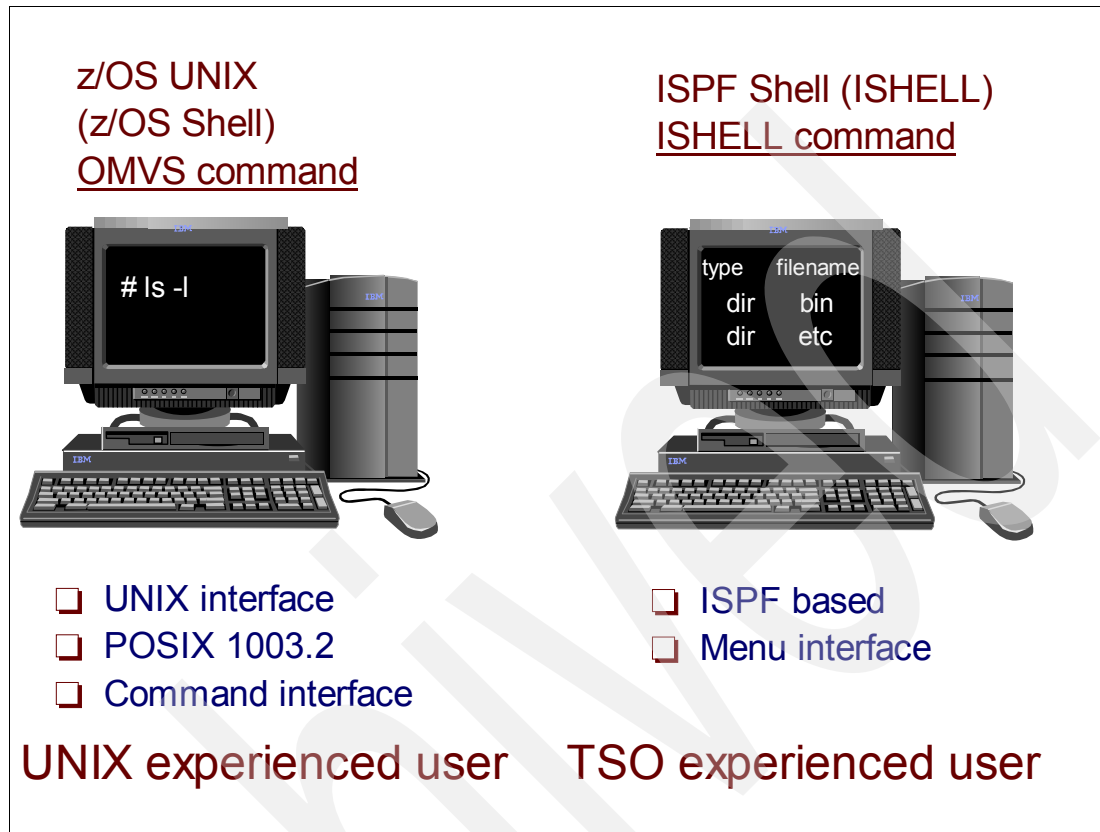


Figure 2-13 z/OS UNIX interactive interfaces

### Using z/OS UNIX

Figure 2-13 shows an overview of the two interactive interfaces, z/OS UNIX shell and the ISPF ISHELL. In addition, there are some TSO/E commands to support z/OS UNIX, but they are limited to certain functions such as copying files, and creating directories.

The z/OS UNIX shell is based on the UNIX System V shell and has some of the features from the UNIX Korn shell. The POSIX standard distinguishes between a command which is a directive to the shell to perform a specific task, and a utility which is the name of a program callable by name from the shell. To the user, there is no difference between a command and a utility.

Interactive users of z/OS UNIX have a choice between using a UNIX-like interface (the shell), a TSO interface (TSO commands), and an ISPF interface (ISPF CUA® dialog). With these choices, users can choose the interface which they are most familiar with and get a quicker start on z/OS UNIX.

The z/OS UNIX shell provides the environment that has the most functions and capabilities. Shell commands can easily be combined in pipes or shell scripts and thereby become powerful new functions. A sequence of shell commands can be stored in a text file which can be executed. This is called a shell script. The shell supports many of the features of a regular programming language.



## **TSO commands**

There are some TSO commands that provide support for UNIX System Services. The **OMVS** command is the command to invoke the z/OS UNIX shell.

The ISPF **ISHELL** command invokes the ISPF shell. The ISHELL is the easy to use starting point for users familiar with TSO and ISPF. The ISPF type dialog provides a panel-driven interface to z/OS UNIX services.

## **ISHELL command**

The ISHELL provides CUA panels where users can work with the hierarchical file system. There are also panels for mounting/unmounting file systems and for doing some z/OS UNIX administration.

Programmers whose primary interactive computing environment is TSO/E and ISPF prefer to work with the ISHELL environment.

## **OMVS command**

An interactive TSO/E user who uses the OMVS command to access the shell can switch back and forth between the shell and TSO/E environments.

Programmers whose primary interactive computing environment is UNIX or AIX find the z/OS shell programming environment familiar.

## 2.14 ISPF Option 6



Figure 2-14 ISPF Option 6 panel

### Interactive use via ISPF Option 6

After a logon to TSO/E, enter Option 6 under ISPF to use the **OMVS** command and the **ISHELL** command.

If you are a user with an MVS background, you may prefer to use the ISPF shell panel interface instead of shell commands or TSO/E commands to work with the file system. The ISPF shell also provides the administrator with a panel interface for setting up users for z/OS UNIX access, for setting up the root file system, and for mounting and unmounting a file system.

You can also run shell commands, REXX programs, and C programs from the ISPF shell. The ISPF shell can direct stdout and stderr only to an HFS file, not to your terminal. If it has any contents, the file is displayed when the command or program completes.

## 2.15 ISHELL ISPF dialog (ISH)

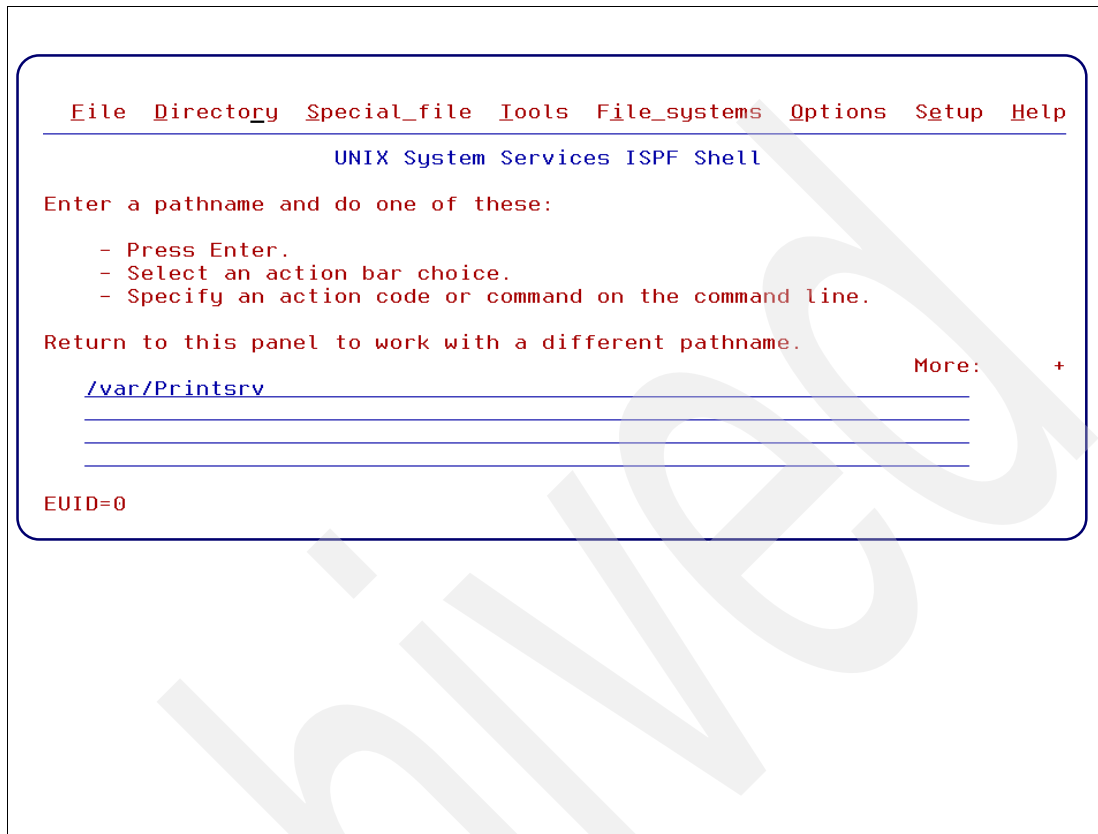


Figure 2-15 Using the ISHELL to display files and directories for /var/Printsrv

### ISHELL ISPF dialog for OpenMVS file system access

In a TSO/E ISPF session, enter the **ISHELL** (**ISH** for short) command to invoke the ISPF shell (dialog) for OpenMVS file system access and various administrative functions. The ISPF shell also provides the administrator with a panel interface for setting up users for z/OS UNIX access, for setting up the root file system, and for mounting and unmounting a file system.

You can also run shell commands, REXX programs, and C programs from the ISPF shell. The ISPF shell can direct stdout and stderr only to an HFS file, not to your terminal. If it has any contents, the file is displayed when the command or program completes. The ISHELL can also be invoked from TSO/E READY mode (provided that SYS1.SBPXEXEC data set is part of the SYSPROC or SYSEXEC DD concatenation and the ISPF libraries are allocated).

Figure 2-15 shows the ISHELL or ISPF Shell panel as a result of the **ISHELL** command. At the top of the panel is the action bar, with seven choices:

- File - Directory - Special file - Tools - File systems - Options - Setup - Help

When you select one of these choices, a pull-down panel with a list of actions is displayed.

**Note:** The effective UID is displayed on the panel as EUID=0. Many Infoprint Server customization functions require the administrator or operator to have an effective UID of 0.

## 2.16 Files and directories in /var/Printsrv

Select one or more files with / or action codes. If / is used also select an action from the action bar otherwise your default action will be used. Select with S to use your default action. Cursor select can also be used for quick navigation. See help for details.

```
EUID=0  /var/Printsrv/
Type  Perm  Changed-EST5EDT  Owner  -----Size  Filename  Row 1 of 44
_ File   664   2003-05-07 08:46  HAIMO         23   ipp.out
_ File   666   2003-05-07 08:45  HAIMO          0   JVM.log
_ Dir    771   2003-05-07 08:45  HAIMO       8192  xf_sockets
_ File   660   2003-05-07 08:45  HAIMO         11  aopipd.pid
_ Dir    771   2003-05-07 08:45  HAIMO       8192  .
_ Dir    770   2003-05-07 08:45  HAIMO       8192  xfd
_ Char   666   2003-05-07 08:45  HAIMO          0  aopxfd.skt
_ File   660   2003-05-07 08:45  HAIMO         11  aopxfd.pid
_ File   660   2003-05-07 08:45  HAIMO         11  aoplpd.pid
_ File   660   2003-05-07 08:45  HAIMO         11  aopd.pid
_ File   660   2003-05-07 08:30  HAIMO      49152  master.db
_ File   660   2003-05-07 08:30  HAIMO     24576  jestoken.db
_ File   660   2003-05-06 03:14  HAIMO          4   aop.id
```

Figure 2-16 Users files and directories displayed from the ISHELL

### Displaying files and directories

Shown in the figure are the files and directories of **/var/Printsrv**. The administrator can then use action codes to do the following:

- b** Browse a file or directory
- e** Edit a file or directory
- d** Delete a file or directory
- r** Rename a file or directory
- a** Show and edit the attributes of a file or directory
- c** Copy a file or directory

## 2.17 OMVS command shell session

```
-rw-rw-rw- 1 HAIMO AOPADMIN 0 May 7 08:45 JVM.log
-rw-rw---- 1 HAIMO AOPADMIN 4 May 6 03:14 aop.id
-rw-rw---- 1 HAIMO OMVSGRP 11 May 7 08:45 aopd.pid
-rw-rw---- 1 HAIMO OMVSGRP 11 May 7 08:45 aopipd.pid
-rw-rw---- 1 HAIMO OMVSGRP 11 May 7 08:45 aoplpd.pid
-rw-rw---- 1 HAIMO OMVSGRP 11 May 7 08:45 aopxfd.pid
crw-rw-rw- 2 HAIMO OMVSGRP 6, 0 May 7 08:45 aopxfd.skt
-rwxrwxrwx 1 HAIMO AOPADMIN 1976 Mar 2 2001 colort.prt
-rw-rw-r-- 1 HAIMO AOPADMIN 23 May 7 08:46 ipp.out
-rw-rw---- 1 HAIMO AOPADMIN 24576 May 7 08:30 jestoken.db
-rw-rw---- 1 HAIMO AOPADMIN 49152 May 7 08:30 master.db
drwxrwxr-x 2 HAIMO AOPADMIN 8192 Feb 20 2001 spool
drwxrwx--x 2 HAIMO AOPADMIN 8192 May 7 08:45 xf_sockets
drwxrwx--- 7 HAIMO AOPADMIN 8192 May 7 08:45 xfd
ROGERS @ SC65: /> ls -al /var/Printsrv
```

Figure 2-17 Displaying files and directories from the OMVS shell

### OMVS command

Use the **OMVS** command to invoke the z/OS shell. After you are working in a shell session, you can switch to subcommand mode, return temporarily to TSO/E command mode, or end the session by exiting the shell.

Shell commands often have options (also known as flags) that you can specify, and they usually take an argument—such as the name of a file or directory. The format for specifying the command begins with the command name, then the option or options, and finally the argument, if any.

## 2.18 Environment variables

- ❑ A name associated with a string of characters made available to the programs that you run
- ❑ Required environment variables for Infoprint Server
  - **PATH - LIBPATH - NLSPATH**
    - Set them in **/etc/profile**
- ❑ Display variables command from shell session:
  - **set**  
  

```
PATH="/usr/lpp/Printsrv/bin:/bin:."
SHELL="/bin/sh"
STEPLIB="none"
TERM="dumb"
TZ="EST5EDT"
_BPXK_SETIBMOPT_TRANSPORT="TCPIPOE"
_BPX_TERMPATH="OMVS"
```

Figure 2-18 Setting environment variables

### Environment variables

When a program begins, an environment is made available to it. The environment consists of strings of the form, name=value, where name is the name associated with the environment variable, and its value is represented by the characters in value. UNIX systems traditionally pass information to programs through the environment variable mechanism.

There are global variables for all shell users and each user can override these variables with an individual set of variables. You can also change any of the values for the duration of your session (or until you change them again). You enter the name of the environment variable and equate it to a new value.

For Infoprint Server processing, some environment variables are required and some are optional. The required variables are shown in Figure 2-18. They should be set in **/etc/profile**.

The **set** command can be used to display the current variables that are set. This command is issued from the OMVS command line.

## Infoprint Server customization

This chapter describes how to customize the Infoprint Server. Following are the Infoprint Server components you must customize to use the different functions that the Infoprint Server components support:

- ▶ Printer Inventory Manager
- ▶ Print Interface
- ▶ IP PrintWay, basic or extended mode
- ▶ Infoprint Port Monitor for Windows (optional)
- ▶ Infoprint Server Transforms and Transform Manager
- ▶ NetSpool
- ▶ Infoprint Central
- ▶ SNMP subagent

You must customize the Printer Inventory Manager to use any functions provided by Infoprint Server and Infoprint Server Transforms.

The Printer Inventory Manager maintains the Printer Inventory and provides Infoprint Server ISPF panels and the Printer Inventory Definition Utility program (pidu) for the administration of entries in the Printer Inventory.

You can customize other components later as your installation decides to use additional functions. If you plan to use more than one Infoprint Server function, customize all related components. For example, to print from VTAM applications to TCP/IP-attached printers, customize the Printer Inventory Manager, NetSpool, and IP PrintWay.

Summary of Infoprint Server components and corresponding Infoprint Server functions:

- ▶ Printer Inventory Manager, Print Interface, and Infoprint Port Monitor for Windows (optional)
  - You can receive print requests from the following sources, and allocate output data sets on the JES spool:
    - Clients that use LPR to LPD protocol

- Clients that use Internet Printing Protocol (IPP)
  - Windows clients that use Server Message Block (SMB) protocol
  - z/OS UNIX
  - z/OS batch jobs that use the AOPPRINT JCL procedure
  - Any Windows application that supports printing
  - z/OS C and C++ programs that use the Infoprint Server Application Programming Interface
  - z/OS batch jobs that specify the Print Interface subsystem on a DD statement
  - z/OS batch jobs that create spool data sets to be printed on the Infoprint Server managed printers
- ▶ Printer Inventory Manager and NetSpool
    - You can receive print requests from VTAM applications (such as CICS and IMS), and allocate output data sets on the JES spool.
  - ▶ Printer Inventory Manager, IP PrintWay, basic or extended mode, and Print Interface (required to transform data when you use the resubmit for filtering function of IP PrintWay basic mode)
    - You can select output data sets from the JES spool and send data to a remote system using one of these transmission protocols:
      - LPR to LPD protocol
      - Internet Printing Protocol (IPP)
      - Direct sockets printing
      - VTAM (basic mode only)
      - E-mail
  - ▶ Printer Inventory Manager, Infoprint Server Transforms, and Transform Manager
    - You can transform data from one format to another, either automatically or with a z/OS UNIX command: afp2pcl, afp2pdf, afp2ps, pcl2afp, ps2afp, pdf2afp, sap2afp, xml2afp, xml2pdf.
  - ▶ Printer Inventory Manager and Infoprint Central
    - You can use Infoprint Central for the Web to work with print jobs, IP PrintWay extended mode printers, PSF printers, and NetSpool logical units.
  - ▶ Printer Inventory Manager and SNMP subagent
    - You can view printer characteristics and status of PSF printers using an SNMP manager.
  - ▶ Printer Inventory Manager and PSF
    - Store PSF system information in the Printer Inventory.



## 3.1 Enabling the Infoprint Server

### ❑ Enabling the Infoprint Server

- Priced feature - Must have an entry in the IFAPRDxx
- Feature name is INFOPRINT SERVER
- Product identifier is 5694-A01

```
PRODUCT OWNER('IBM CORP')  
NAME('z/OS')  
ID(5694-A01)  
VERSION(*) RELEASE(*) MOD(*)  
FEATURENAME('INFOPRINT SERVER')  
STATE(ENABLED)
```

### ❑ Infoprint Server requires LE run-time

- CEE.SCEERUN, CEE.SCEERUN2, and C++  
CBC.SCLBDLL libraries
- LNKLIST recommended
- STEPLIBLIST facility of BPXPRMxx PARMLIB member
- STEPLIB export through etc/profile

Figure 3-1 Enabling the Infoprint Server

### Enable Infoprint Server

If the Infoprint Server priced feature was not included in your original z/OS order, you must dynamically enable Infoprint Server. To do this, you must add an entry for Infoprint Server to the active IFAPRDxx member of SYS1.PARMLIB. The feature name is INFOPRINT SERVER, and the product identifier is 5694-A01.

The D PROD operator command displays information about products that have been registered or displays the product enablement policy. You can also use the command to determine the state (enabled, disabled, not defined, or not found) that, according to the current policy, exists for a specific product or set of products.

The SET PROD= operator command activates the selected IFAPRDxx members of the logical parmlib that contain the desired product enablement policy. If a policy already exists, the system performs the actions defined in the specified member(s) to modify the existing policy.

### LE run-time library

Infoprint Server requires that the Language Environment (LE) run-time library (CEE.SCEERUN and CEE.SCEERUN2) and the C++ run-time library (CBC.SCLBDLL) reside in an APF-authorized library and be available to Infoprint Server and Infoprint transforms.

IBM recommends that you add the SCEERUN, SCEERUN2, and SCLBDLL data sets to the system LNKLST.

Some installations choose not to add the SCEERUN and SCLBDLL libraries to LNKLST because these libraries contain module names that might intersect with names in other libraries. However, the module names in SCEERUN2 do not intersect with module names in other libraries, so you can add SCEERUN2 to LNKLST with no adverse effects. If you do not add SCEERUN2 to LNKLST, specify SCEERUN2 in the same places you specify SCEERUN in the next step.

However, if you do not add the SCEERUN and SCLBDLL data sets to LNKLST, you can take these steps:

- ▶ Add the SCEERUN and SCLBDLL data sets to the STEPLIBLIST facility of the BPXPRMxx PARMLIB member.
- ▶ Add this statement to the **/etc/rc** file:  
**export STEPLIB=hlq.SCEERUN:hlq.SCLBDLL**

For other details see *z/OS Infoprint Server Customization*, S544-5744.

## 3.2 Basics of the z/OS UNIX file system

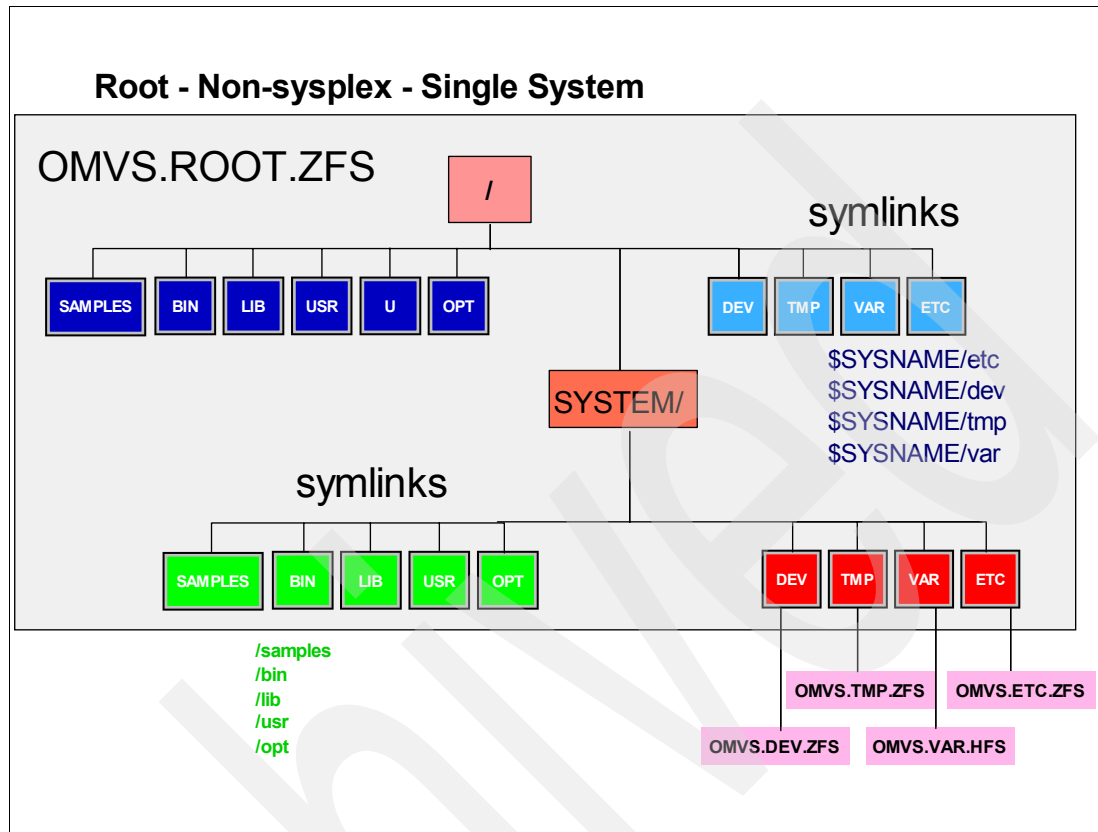


Figure 3-2 Structure of the root file system in a single system environment

### z/OS root file system structure

The z/OS UNIX file system, like other UNIX systems, is a hierarchical file system consisting of the root file system and all the file systems that are added to it. Files are members of a directory, and each directory is in turn a member of another directory at a higher level. The highest level of the hierarchy is the *root directory*. Each instance of the system contains only one root directory.

The root HFS data set contains system code and binaries, including the /bin, /usr, /lib, /opt, and /samples directories. These directories contain files that are installed and serviced by SMP/E.

You must maintain a separate HFS data set for each of the following directories:

- /etc** /etc contains customization data. Keeping the /etc file system in an HFS data set separate from other file systems allows you to separate your customization data from IBM's service updates.
- /dev** /dev contains character-special files that are used when logging into the OMVS shell environment and also during c89 processing.
- /tmp** /tmp contains temporary data that are used by products and applications. /tmp, is created empty, and temporary files are created dynamically by different elements and products. You have the option of mounting a temporary file system (TFS) on /tmp.

**/var** /var contains dynamic data that is used internally by products and by elements and features of z/OS. Any files or directories that are needed are created when code is executed or customized. You can be assured that IBM products will only create files under **/var** when code is executed or customized.

For users, you should logically mount other HFS data sets on the root file system. You should also have your users place their directories and files in the mounted file systems. Separate user file systems offer several advantages:

- ▶ They improve storage management because the system administrator only needs to allocate data sets that are large enough to accommodate the needs of individual users.
- ▶ They enable failure isolation because the system administrator can unmount the user file system that caused an error without affecting other users' data or causing z/OS UNIX to fail.
- ▶ They relieve the contention for system resources that could occur by having multiple users in a single file system.

Name each user's home directory /u/userid where userid is the user ID in lowercase.

In z/OS the root file system contains an additional directory, /SYSTEM; existing directories, /etc, /dev, /tmp and /var are converted into symbolic links. These changes, however, are transparent to the user who brings up a single system environment.

If the content of the symbolic link begins with \$SYSNAME and SYSPLEX is specified NO, then \$SYSNAME is replaced with /SYSTEM when the symbolic link is resolved.

The example below shows what BPXPRMxx file system parameters would look like in an z/OS single system environment, and Figure 3-2 on page 69 shows the corresponding single system image. SYSPLEX(NO) is specified (or the default taken), and the mount mode is read-write.

### Sample BPXPRMxx member

```
FILESYSTYPE TYPE(HFS) ENTRYPOINT(GFUAINIT) PARM(' ') SYSPLEX(NO)
FILESYSTYPE TYPE(ZFS) ENTRYPOINT(IOEFSCM) ASNAME(ZFS,'SUB=MSTR')
ROOT    FILESYSTEM('OMVS.ROOT.ZFS')
        TYPE(HFS) MODE(RDWR)
MOUNT   FILESYSTEM('OMVS.DEV.ZFS') TYPE(ZFS) MODE(RDWR)
        MOUNTPoint('/dev')
MOUNT   FILESYSTEM('OMVS.TMP.ZFS') TYPE(ZFS) MODE(RDWR)
        MOUNTPoint('/tmp')
MOUNT   FILESYSTEM('OMVS.VAR.HFS') TYPE(HFS) MODE(RDWR)
        MOUNTPoint('/var') NOAUTOMOVE
MOUNT   FILESYSTEM('OMVS.ETC.ZFS') TYPE(ZFS) MODE(RDWR)
        MOUNTPoint('/etc')
```

### 3.3 z/OS UNIX files and Infoprint Server

#### ❑ UNIX System Services

- Required to install Infoprint Server
- Uses hierarchical file system (HFS) or zSeries File System (zFS)
- Infoprint Server default base directory `/usr/lpp/Printsrv`
  - Executables - samples
  - Messages - Windows client
- Configuration files - `/etc/Printsrv/aopd.conf`
- Printer Inventory files
  - `/var/Printsrv/master.db`
  - `/var/Printsrv/jestoken.db` and `/var/Printsrv/pwjestoken.db`

Figure 3-3 Requirements to install Infoprint Server

#### UNIX System Services

Infoprint Server uses UNIX System Services (z/OS UNIX). All Infoprint Server customization tasks requires an administrator that has z/OS UNIX superuser authority. (A UID of 0 or READ access to the BPX.SUPERUSER profile in the FACILITY class in RACF is used to define an z/OS UNIX superuser. A superuser can perform any z/OS UNIX function and passes all z/OS UNIX security checks.)

#### File system structure

A hierarchical file system (HFS) consists of the following:

- ▶ zFS or HFS files, which contain data or programs. A file containing a load module or shell script or REXX program is called an executable file. Files are kept in directories.
- ▶ Directories that contain files, other directories, or both. Directories are arranged hierarchically, in a structure that resembles an upside-down tree, with the root directory at the top and the branches at the bottom. The root is the first directory for the file system at the top of the tree and is designated by a slash (/).
- ▶ Additional local or remote file systems, which are mounted on directories of the root file system or of additional file systems.

## Infoprint Server directories

All Infoprint Server executables, messages, and sample files are either HFS or zFS after the install process. These files are installed into the HFS directory `/usr/lpp/Printsrv`. The Printer Inventory Manager uses the following directories:

- /etc/Printsrv** The `/etc/Printsrv` directory contains all Infoprint Server configuration files. This directory is created automatically with the appropriate permission bit settings when you install Infoprint Server.
- /var/Printsrv** The `/var/Printsrv` directory contains the Printer Inventory and other Infoprint Server files. You can change the name of this directory in the base-directory attribute in the Infoprint Server configuration file (`aopd.conf`).
- If the `/var/Printsrv` directory or the directory named in the base-directory attribute does not already exist, the `aopsetup` shell script creates the directory and sets the appropriate permissions for it. If your system is part of a sysplex, the `/var` file system must be system-specific and designated NOAUTOMOVE in the BPXPRMxx parmlib member. If you specify a different base directory in the base directory attribute in the Infoprint Server configuration file, the file system that contains this directory must be system-specific and designated NOAUTOMOVE.
- The `/var/Printsrv` directory is created automatically when you install Infoprint Server. The `aopsetup` shell script defines the appropriate permissions for the `/var/Printsrv` directory.

The `/var/Printsrv` directory and its subdirectories contain several files:

- ▶ **/var/Printsrv:** This directory contains:
  - Printer Inventory files `master.db`, `jestoken.db`, and `pwjestoken.db`.
  - Temporary files that the Print Interface LPD creates as it receives data from clients that send the control file after sending data files. By default, most clients send the control file after sending data files. The Infoprint Port Monitor, however, always sends the control file first. Commands such as `ls` do not display these files because the LPD unlinks them after it opens them. When the LPD closes the files, they are deleted.
  - Language Environment dumps (CEEDUMPs)
- ▶ **/var/Printsrv/printway:** This directory contains temporary files that IP PrintWay extended mode writes before transmitting data to printers. IP PrintWay extended mode always writes temporary files when it uses the LPR transmission protocol. It also writes temporary files when it uses the IPP or direct-socket transmission protocol, but only if more than one copy is requested. IP PrintWay basic mode never writes temporary files in this directory. Instead, IP PrintWay basic mode writes temporary files in hiperspace.
- ▶ **/var/Printsrv/logs and /var/Printsrv/logdb:** These directories contain message logs and the common message log database.
- ▶ **/var/Printsrv/hinv:** This directory contains the Historical Inventory.
- ▶ **/var/Printsrv/searchdb:** This directory contains the search database.
- ▶ **/var/Printsrv/trace:** This directory contains trace information when you activate tracing with the `AOPTRACEON` environment variable. IP PrintWay basic mode does not write trace information to this directory.
- ▶ **/var/Printsrv/xfd and /var/Printsrv/xf\_sockets:** These directories and their subdirectories contain temporary files and messages for Infoprint transform products and Language environment dumps (CEEDUMPs).

### 3.4 Infoprint Server directories/files

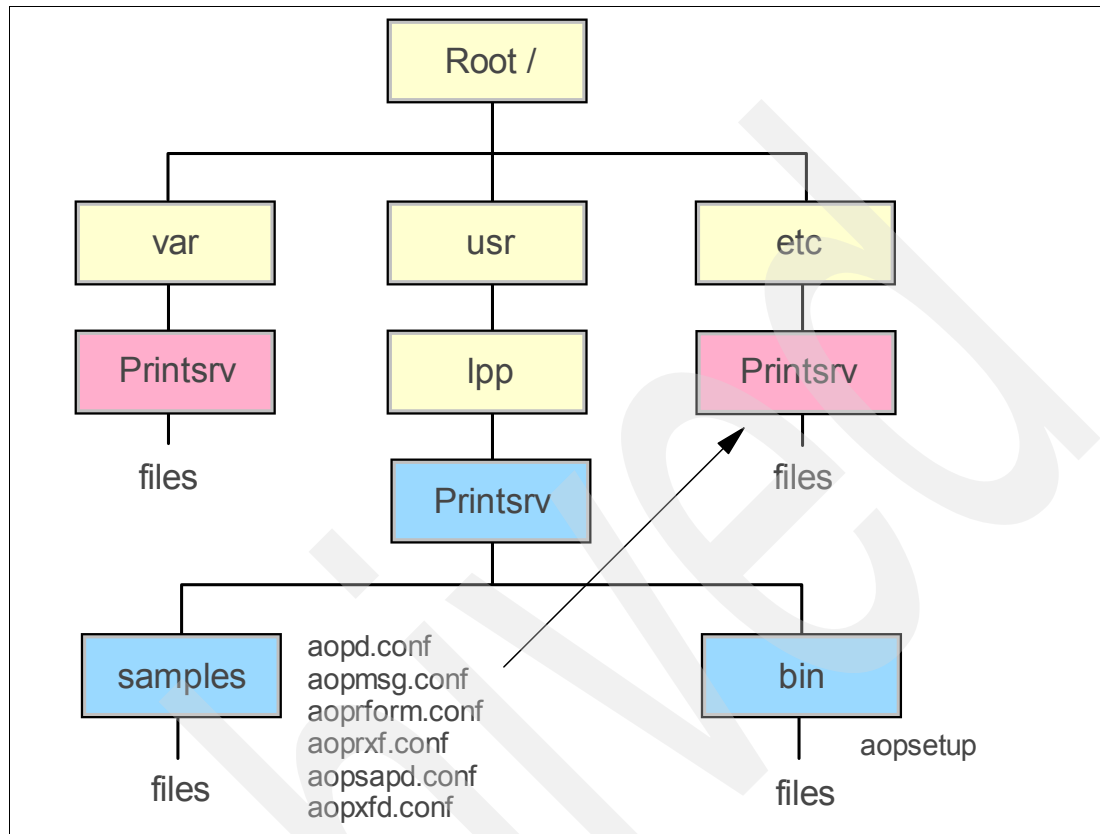


Figure 3-4 HFS directories and files used by Printer Inventory Manager

#### Infoprint Server files and directories

The Printer Inventory Manager uses these hierarchical file system (HFS) directories:

- The `/etc/Printsrv` directory is the default location for Infoprint Server configuration files.

**Note:** The configuration files must be copied from `/usr/lpp/Printsrv/samples` to `/etc/Printsrv`.

- The `/var/Printsrv` directory is the default location for the Printer Inventory files.  
The `aopsetup` shell script defines permissions for the `/var/Printsrv` directory.

**Note:** You access the `aopsetup` command through `/usr/lpp/Printsrv/bin` directory.

All Infoprint Server operator commands to be issued from the OMVS shell are accessed through `/usr/lpp/Printsrv/bin`. Use environment variables to give z/OS UNIX users and Infoprint Server administrators and operators access to the commands.

### 3.5 Infoprint Server configuration file (aopd.conf)

- ❑ **aopd.conf - Infoprint Server configuration file**
  - **Infoprint Server customization data**
  - **Default directory /etc/Printsrv**
    - # aopd.conf - Default Infoprint Server configuration**
    - # lpd-port-number = 515**
    - # ipp-port-number = 631**
    - base-directory = /var/Printsrv**
    - # ascii-codepage = ISO8859-1**
    - # ebcdic-codepage = IBM-1047**
    - # inventory = AOP1**
    - # job-prefix = PS**
    - # log-retention = 1**
    - # max-historical-inventory-size = 10**
    - # resolve-printway-printers = yes**
    - # smf-recording = yes**
    - # start-daemons = { }**
- ❑ **SYS1.SAMPLIB(AOPCPETC) job copies all configuration files to /etc/Printsrv/**
- ❑ **AOPCONF environment variable - full path name of the configuration file**

Figure 3-5 Infoprint Server configuration file (aopd.conf)

#### Customizing the aopd.conf file

The Infoprint Server configuration file, aopd.conf, lets you customize the Printer Inventory Manager and other components of Infoprint Server. This file is optional. If the configuration file does not exist or if an attribute in the configuration file is omitted, the default values shown in Figure 3-5 are used.

The default location of this configuration file is /etc/Printsrv/aopd.conf. The attributes in the aopd.conf default configuration file are sufficient to allow the Printer Inventory Manager to be started.

**Recommendation:** For a secure environment, do not change the owner or permissions of the configuration file. The file should be owned by UID 0 and be writeable only by a user with a UID of 0.

The aopd.conf file contains attributes that customize Infoprint Server. An aopd.conf file containing the base-directory attribute is required. If other attributes are missing, Infoprint Server uses the default.

#### ► **ascii-codepage = codepage**

The name of an IBM-supplied or custom ASCII code page supported by the z/OS iconv utility. For the names of IBM-supplied code page names, see *z/OS XL C/C++ Programming Guide*, SC09-4765.



Infoprint Server converts data from: the document code page to the printer code page; the code page for the z/OS locale to the document code page; and the code page for the z/OS locale to the printer code page. Default: `ascii-codepage = ISO8859-1`

► **console-name = name**

The extended MCS console that Infoprint Central uses to send commands to the z/OS system.

► **base-directory = pathname**

The directory path in which the Printer Inventory Manager creates Printer Inventory files, Historical Inventory files, common message log files, and other files. Also, Print Interface creates files in this directory.

**Note:** Do not change this attribute while any Infoprint Server daemons are running.

Default: `base-directory = /var/Printsrv`

► **ebcdic-codepage = codepage**

The name of an IBM-supplied or custom EBCDIC code page supported by the z/OS `iconv` utility. For the names of IBM-supplied code page names, see *z/OS XL C/C++ Programming Guide*, SC09-4765.

Infoprint Server converts data from: the document code page to the printer code page; the code page for the z/OS locale to the document code page; and the code page for the z/OS locale to the printer code page. Therefore, if you specify a custom code page in this field, make sure that conversion tables exist to let Infoprint Server convert between code pages.

Default: `ebcdic-codepage = IBM-1047`

► **inventory = inventory**

The name of the Printer Inventory and the Print Interface subsystem. Specify exactly 4 uppercase or lowercase letters or numbers. This name is case-sensitive.

If you use the Print Interface subsystem, the first character must be an uppercase letter. The job submitter specifies this name in the `SUBSYS` parameter on the `DD JCL` statement.

If you start NetSpool or IP PrintWay basic mode, specify this name on the `EXEC` statement of the startup procedure. Also, specify this name in the PSF startup procedure if you want PSF to use information that is specified in PSF FSS and FSA definitions in the Infoprint Server Printer Inventory.

Default: `inventory = AOP1`

► **ipp-port-number = portnumber**

The number of the port at which the IPP server waits for print requests. Port 631 is the well-known port for communication between IPP clients and IPP servers.

Default: `ipp-port-number = 631`

► **job-prefix = prefix**

A prefix that Print Interface or NetSpool uses for all Infoprint Server job IDs. Specify 2 letters, numbers, or national (`@ $ #`) characters. Enclose a prefix that contains national characters in single or double quotation marks. The first character cannot be numeric.

Print Interface or NetSpool assigns a unique Infoprint Server job ID to each `SYSOUT` data set that it creates on the JES spool. The Infoprint Server job ID begins with the prefix you specify. The z/OS UNIX `lpstat` and `cancel` commands use the Infoprint Server job ID to display and delete data sets.

The z/OS job ID is different from the Infoprint Server job ID. JES operator commands always return the z/OS job ID.

In most cases, the job ID field of the SYSOUT data set on the JES spool contains the Infoprint Server job ID, instead of the z/OS job ID. However, in most cases where Print Interface processes a data set that already has a z/OS job ID associated with it, Print Interface uses the existing z/OS job ID in the job ID field.

Default: job-prefix = PS

► **lpd-port-number = portnumber**

The number of the port at which the Infoprint Server LPD waits for print requests. Port 515 is the well-known port for communication between LPRs and LPDs. Windows users must specify this port number when they configure the Infoprint Port Monitor for Windows. This attribute is optional.

**Note:** This port number must not be reserved in the hlq.PROFILE.TCPIP data set.

Default: lpd-port-number = 515

► **max-historical-inventory-size = megabytes**

The maximum size in megabytes (MBs) of the Historical Inventory.

► **resolve-printway-printers = yes | no**

**Yes** means that IP PrintWay looks up the full host names of printers in the DNS.

► **smf-recording = yes | no**

**Yes** means that IP PrintWay extended mode writes SMF type 6 records with accounting information.

► **snmp-community = name | public**

The name assigned to the SNMP community for making SNMP requests to the z/OS system. This name must match the community name provided to the z/OS SNMP agent *and* the community name defined to the SNMP manager (Network Printer Manager).

Default: snmp-community = public

► **start-daemons = {[ipdd] [lpd] [netd] [outd] [snmpd] [ssid] [subd] [xpd]}**

The Infoprint Server daemons that are started when you use the **aopstart** command. You can specify one or more of these daemon names, enclosed in braces.

Printer Inventory Manager daemons aopd, aopsdbd, aophinvd, and aoplogd always start, regardless of which daemons you specify in this attribute. *To start only the Printer Inventory Manager daemons, type braces with no daemon names.*

Default: start-daemons = {lpd}

*z/OS Infoprint Customization*, S544-5744 lists the steps for creating and editing the aopd.conf file.

If you do not create the Infoprint Server configuration file in /etc/Printsrv/aopd.conf, you must set the **AOPCONF** environment variable to the full path name of the configuration file. The Printer Inventory Manager, as well as other Infoprint Server daemons, use environment variables specified in the *aopstart* EXEC in /usr/lpp/Printsrv/bin/ directory. If you created the configuration file in /etc/Printsrv/aopd.conf, you do not need to set this environment variable.

## 3.6 Full format of the aopd.conf file

### ❑ Full format of the aopd.conf file:

```
[#comment]
[ ascii-codepage = ISO8859-1 | codepage ]
[ ebcdic-codepage = IBM1047 | codepage ]
[ base-directory = /var/Printsrv | path ]
[ console-name = name ]
[ inventory = AOP1 | inventory ]
[ ipp-port-number = 631 | portnumber ]
[ job-prefix = PS | prefix ]
[ log-retention = 1 | days ]
[ lpd-port-number = 515 | portnumber ]
[ max-historical-inventory-size = 10 | megabytes ]
[ resolve-printway-printers = yes | no ]
[ smf-recording = yes | no ]
[ snmp-community = name | public ]
[ start-daemons = {[ippd] [lpd] [netd] [outd] [snmpd] [ssid] [subd] [xfd]}
```

Figure 3-6 Full format of the aopd.conf file

### aopd.conf configuration file parameters

The following parameters are optional and should be used if those functions are required for use with Infoprint Server:

#### ► console-name = name

The name of the extended MCS console that Infoprint Central uses to send commands to the z/OS system.

**Default:** The Printer Inventory name (specified in the inventory attribute), followed by the last 4 characters of the system name. If the Printer Inventory name is AOP1 and the system name is MYSYS1, the default console name is AOP1SYS1.

#### ► log-retention = days

The number of days worth of messages to keep in the common message log and information about output data sets that are no longer on the JES spool to keep in the Historical Inventory. You can specify a value from 0 - 59. A value of 0 means that no messages are kept in the common message log, and no information is kept in the Historical Inventory.

Default:

- If you start Infoprint Central (the aopssid daemon), the default is log-retention = 1.
- Otherwise, the default is log-retention = 0.

► **max-historical-inventory-size = megabytes**

The maximum size in megabytes (MBs) of the Historical Inventory, /var/Printsrv/hinv. The Historical Inventory contains information about output data sets (print jobs) that are no longer on the JES spool. When the maximum size is reached, Infoprint Server overwrites the oldest historical information. A value of 0 means that Infoprint Server does not limit the size of the Historical Inventory. This attribute is used only if you start the Infoprint Central daemon (aopssid) or the log-retention attribute specifies a value greater than 0.

**Recommendation:** Specify a maximum size if the space available to the file system mounted at the /var or /var/Printsrv mount point is limited.

Default: 100MB, which is adequate for approximately 50,000 data sets

► **smf-recording = yes | no**

Indicates whether IP PrintWay extended mode is to write SMF type 6 records for data sets that it sends to a printer or e-mail destination. SMF type 6 records contain accounting information.

Default: smf-recording = yes

► **start-daemons = {[ippd] [lpd] [netd] [outd] [snmpd] [ssid] [subd] [xfd]}**

The Infoprint Server daemons that are started when you use the **aopstart** command. You can specify one or more of these daemon names, enclosed in braces.

Printer Inventory Manager daemons **aopd**, **aopsdbd**, **aophinvd**, and **aoplogd** always start, regardless of which daemons you specify in this attribute. To start only the Printer Inventory Manager daemons, type braces with no daemon names.

Default: start-daemons = {lpd}

## 3.7 Infoprint Server daemons

- ❑ `start-daemons = { }` **selectable daemons**
  - ❑ **ippd** - **aopipdd**                      ❑ **Internet Printing Protocol**
  - ❑ **lpd** - **aoplpd**                        ❑ **Line printer daemon**
  - ❑ **netd** - **aopnetd**                      ❑ **Netspool**
  - ❑ **outd** - **aopoutd & aopwsmd**       ❑ **IP Printway extended mode**
  - ❑ **snmpd** - **aopsnmpd**                  ❑ **SNMP subagent**
  - ❑ **ssid** - **aopssid**                     ❑ **Infoprint Central**
  - ❑ **subd** - **aopsubd**                    ❑ **Print Interface subsystem**
  - ❑ **xfd** - **aopxfd**                      ❑ **Transform Manager**
- ❑ Printer Inventory daemons **aopd**, **aopsdbd** (search database daemon), **aophinvd**, **aoplogd** and **aopsapd** are always started by the *aopstart* command
- ❑ `start-daemons = { }` starts only the Printer Inventory Manager

Figure 3-7 Infoprint Server daemons

### Infoprint Server daemons

Daemons are programs that run unattended to do continuous functions when requested by users. You should start the daemons that support the Infoprint Server functions that you want to implement. The daemons used by Infoprint Server are as follows:

- ippd**        The *aopipdd* daemon. This IPP server daemon processes print jobs submitted by an Internet Printing Protocol (IPP) client.
- lpd**        The *aoplpd* daemon. This line printer daemon processes print jobs submitted with commands, such as *lpr*, and by the Infoprint Port Monitor for Windows.
- netd**        The *aopnetd* daemon. This NetSpool daemon supports printing from VTAM applications such as CICS and IMS.
- outd**        The *aopoutd* and *aopwsmd* daemons. These IP PrintWay *extended mode* daemons select output data sets from the JES spool and send them to remote printers in a TCP/IP network or to e-mail destinations.
- snmpd**       The *aopsnmpd* daemon. This SNMP subagent daemon provides status information to the z/OS SNMP agent about printers controlled by PSF.
- ssid**        The *aopssid* daemon. This Infoprint Central daemon communicates between Infoprint Central and JES.
- subd**        The *aopsubd* daemon. This Print Interface subsystem daemon processes output data when the subsystem is requested in the SUBSYS parameter on a DD statement.

**xfd** The aopxfd daemon. This Transform Manager daemon manages some of the data stream transforms that Infoprint Server Transforms provide.

The following daemons are always started by the **aopstart** command:

**aopd** The aopd daemon manages the Printer Inventory.

**aophinvd** A Printer Inventory Manager daemon that maintains the Historical Inventory. The Historical Inventory contains information about Infoprint Server output data sets that are not on the JES spool.

**aoplogd** A Printer Inventory Manager daemon that manages the common message log.

**aopsapd** The SAP Callback daemon handles callback notification for the SAP Output Management System (OMS). It starts automatically when a print request is received from SAP R/3.

## 3.8 Customizing the Printer Inventory Manager

### ❑ Printer Inventory Manager customization

- Create Infoprint Server HFS directories
- Make LE run-time libraries available
- Create the Infoprint Server configuration file (aopd.conf)
- Set Printer Inventory Manager environment variables
- Set up security
- Customize startup and shutdown commands and procedures
- Enable Infoprint Server ISPF panels
- Set up backup and restore procedures for the Printer Inventory
- Set Workload Manager goals



Figure 3-8 Customizing the Printer Inventory Manager

### Customizing the Printer Inventory Manager

The Printer Inventory Manager component of Infoprint Server controls the Printer Inventory. The Printer Inventory consists of HFS or zFS files that contain information about the printing environment. The administrator must create and manage information in the Printer Inventory.

**Note:** The Printer Inventory cannot be shared by Infoprint Server running at the same or different levels on other systems.

The Printer Inventory Manager provides Infoprint Server ISPF dialogs and the Printer Inventory Definition Utility program (pidu) for the administrator to create and manage entries in the Printer Inventory. The Printer Inventory Manager also maintains the common message log and the Historical Inventory.

Customization tasks:

- ▶ **Create Infoprint Server directories:** The Printer Inventory Manager uses these directories, which can be in hierarchical file systems (HFS) or z/OS Distributed File Service file systems (zFS):
  - The /etc/Printsrv directory, which is the default location for Infoprint Server configuration files.
  - The /var/Printsrv directory, which is the default location for other Infoprint Server files, including the Printer Inventory files.

The amount of space required for the file system at the /var/Printsrv mount point depends on a variety of factors, such as the number of printers you have, the size of data sets you process, and the number of days worth of messages and historical information you want to keep.

- ▶ **Make Language Environment and C++ run-time libraries available:** CEE.SCEERUN, CEE.SCEERUN2, and CBC.SCLBDLL LE and C++ run-time libraries must reside in an APF-authorized library and be available to Infoprint Server and to Infoprint transforms.
- ▶ **Create the Infoprint Server configuration file (aopd.conf):** The Infoprint Server configuration file, aopd.conf, contains attributes that let you customize the Printer Inventory Manager and other components of Infoprint Server. The default location of this configuration file is /etc/Printsrv/aopd.conf.
- ▶ **Set environment variables for the Printer Inventory Manager:** The Printer Inventory Manager, as well as other Infoprint Server daemons, use environment variables specified in the aopstart EXEC in /usr/lpp/Printsrv/bin/ directory.
- ▶ **Set up security for the Printer Inventory, common message log, and operator commands:** Specify which users can read and update the Printer Inventory, view messages in the common message log, and start and stop Infoprint Server daemons. You can use the Resource Access Control Facility (RACF) or another program that follows system authorization facility (SAF) protocol to set up security.

You must run the **aopsetup** shell script to set up the correct z/OS UNIX permissions for Infoprint Server directories and executable files before you start Infoprint Server daemons for the first time. Also, you must rerun aopsetup whenever you move to a new z/OS release.

- ▶ **Customize startup and shutdown commands and procedures:** To start and stop Infoprint Server daemons, including the Printer Inventory daemon, you can use one of these methods:
  - AOPSTART and AOPSTOP JCL procedures
  - z/OS UNIX **aopstart** and **aopstop** commands

In addition, you can use z/OS automation tools or the /etc/rc shell script to start Infoprint Server daemons automatically during system initialization.

- ▶ **Enable Infoprint Server ISPF panels:** The administrators can use Infoprint Server ISPF panels to add, browse, copy, edit, and delete printer definitions and other objects in the Printer Inventory.
- ▶ **Back up and restore the Printer Inventory:** You should back up the Printer Inventory on a regular basis.

When you restore the Printer Inventory, no print jobs are lost. However, information in the Printer Inventory about current print jobs is lost. This means that users, including SAP R/3 users, are not notified when jobs submitted through Print Interface complete, users cannot query the status of the jobs, and operators cannot see information about the print jobs in Infoprint Central.

- ▶ **Set Workload Manager goals:** There are considerations for defining Workload Manager (WLM) classification rules for prioritizing Infoprint Server work. The WLM component of z/OS lets you manage workload distribution, balance workload, and distribute resources to competing workloads.



### 3.9 Functions using Printer Inventory

- ❑ Receiving print requests from clients - Print Interface
- ❑ Receiving print requests from VTAM applications - Netspool
- ❑ Transforming data from one format to another - Transform Interface
- ❑ Transmitting output from JES spool to remote systems or printers - IP PrintWay
- ❑ Viewing printer status and characteristics - Infoprint Central
- ❑ Viewing printer status stored by PSF - SMNP



The Printer Inventory **cannot** be shared by Infoprint Servers running at the same or different levels on other systems

Figure 3-9 Functions requiring the Printer Inventory

#### Print Inventory

The Printer Inventory is common to the Print Interface, IP PrintWay, and NetSpool components of Infoprint Server.

The administrators use Infoprint Server ISPF panels and the Printer Inventory Definition Utility (pidu) to create and maintain the Printer Inventory. The pidu program is useful for creating many printer definitions at the same time and for backing up the Printer Inventory.

The information in the Printer Inventory is used by several Infoprint Server components:

- ▶ NetSpool uses information in printer definitions and in printer pool definitions.
- ▶ Print Interface uses information in printer definitions.
- ▶ IP PrintWay uses information in printer definitions. Also, IP PrintWay basic mode uses configuration information in FSS and FSA definitions, and IP PrintWay extended mode uses job selection rules.
- ▶ The SNMP subagent uses printer information that PSF stores in the Printer Inventory about PSF printers.
- ▶ Infoprint Central displays Printer Inventory information.

#### PSF for z/OS

PSF for z/OS, a separate product, can, as an option, use printer configuration information that the PSF system programmer specifies in FSS and FSA definitions in the Printer Inventory.

The printer configuration information in the FSS and FSA definitions is the same as the configuration information that the system programmer can alternatively specify in PSF startup procedures and PSF exits.

When the printer configuration information is specified in the Printer Inventory:

- ▶ The information can be changed without restarting all FSAs in the FSS. Only the FSAs with changed configuration information need to be restarted.
- ▶ Authorized operators can use Infoprint Central to work with PSF-controlled printers. For example, operators can start and stop (that is, drain) printers, change JES work-selection criteria, pause or interrupt printers, and turn TCP/IP-connected printers online and offline.

To customizing PSF to use the Printer Inventory you can use a migration program in Infoprint Server to copy parameters from the PRINTDEV statement and EXEC PARM statement into the Printer Inventory. You must manually migrate parameters from Exit 7.

The PSF printer startup procedure has to be changed to specify `INV=inventory` as the first parameter in the PARM field of the EXEC statement:

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

In this statement *inventory* is the four-character name of the Printer Inventory.

The `INV=piname` parameter indicates that PSF uses the Printer Inventory and obtains parameters from the specified Printer Inventory for each printer in the startup procedure. No other parameters in the PARM field are used when PSF uses the Printer Inventory.

**Important:** The Printer Inventory cannot be shared by any Infoprint Server components running at the same or different levels on other systems.

## 3.10 Printer Inventory directories and files

- ❑ Printer Inventory files are created automatically when:
  - Administrator uses ISPF panels or pidu program
- ❑ Pprotect access to the Printer Inventory files
  - Administrator(s) should have effective UID=0
- ❑ Infoprint Server directories
  - `/etc/Printsrv` - Infoprint Server configuration files
  - `/var/Printsrv` - Infoprint Server files, including the Printer Inventory
    - Both created automatically when installing Infoprint Server



**Important:** Allocate at least 750 MB of DASD space for the file system mounted at the `/var/Printsrv` mount point. If you do not run IP PrintWay extended mode or use data stream transforms, you might need less space. If you print extremely large files or if you want to keep many days worth of messages and historical information, you might need more space.

Figure 3-10 Files created in the Printer Inventory

### Printer Inventory directories

The Printer Inventory Manager uses two hierarchical file system (HFS) directories:

- ▶ The `/etc/Printsrv` directory, which is the default location for Infoprint Server configuration files.

The `/etc/Printsrv` directory contains all Infoprint Server configuration files. This directory is created automatically with the appropriate permissions when you install Infoprint Server.

**Recommendation:** Do not change the owner or permissions of the `/etc/Printsrv` directory. For a secure environment, this directory should be:

- ▶ Owned by UID of 0.
- ▶ Writeable only by users with an effective UID of 0.

- ▶ The `/var/Printsrv` directory, which is the default location for other Infoprint Server files, including the Printer Inventory files.

The `/var/Printsrv` directory contains the Printer Inventory and other Infoprint Server files. The `/var/Printsrv` directory is created automatically when you install Infoprint Server. The `aopsetup` shell script defines the appropriate permissions for the `/var/Printsrv` directory.

**Recommendation:** Mount a separate file system at the `/var` mount point and create the `/var/Printsrv` directory in that file system.

Do not change the owner or permissions of the **/var/Printsrv** directory after it is created. For a secure environment, this directory should be:

- Owned by UID of 0
- Readable and writeable only by users with an effective UID of 0 or members of the AOPADMIN group or any other group. (AOPADMIN name is used through the Infoprint publications for the administrator group.)
- Executable by everyone

## Sysplex support

If your system is part of a sysplex, the **/var** filesystem must be system-specific and designated NOAUTOMOVE in the BPXPRMxx parmlib member. If you specify a different base directory in the base-directory attribute in the Infoprint Server configuration file, the filesystem that contains this directory must be system-specific and designated NOAUTOMOVE.

## Printer Inventory files

Infoprint Server creates the Printer Inventory files automatically the first time the administrator uses the Infoprint Server ISPF panels or the Printer Inventory Definition Utility (pidu) to create objects in the Printer Inventory, such as printer definitions. The Printer Inventory files also contain objects that the administrator does not create. For example, Print Interface creates objects for each job processed. These job objects are deleted when the data sets to which they correspond are deleted from the JES spool.

The Printer Inventory is comprised of these files:

- ▶ master.db
- ▶ jestoken.db
- ▶ pwjestoken.db

The master.db, jestoken.db, and pwjestoken.db files are data base files optimized for rapid direct access to objects. As you add objects to the Printer Inventory, these files increase in size. When you remove objects, the files do not decrease in size because the Printer Inventory Manager simply designates as available the space within the file that had been occupied by the removed objects. When you add objects in the future, the Printer Inventory Manager uses available space within the files. The files increase in size only when they do not contain sufficient available storage. So, the size of each file can be characterized as a high-water mark.

**Note:** Make sure that the file system that contains the **/var/Printsrv** directory has enough space.

The **/var/Printsrv** directory and its subdirectories contain these types of files:

- ▶ **/var/Printsrv:** This directory contains:
  - Printer Inventory files **master.db**, **jestoken.db**, and **pwjestoken.db**
  - Temporary files that the Print Interface LPD creates as it receives data from clients that send the control file after sending data files
  - Language Environment dumps (CEEDUMPs)
- ▶ **/var/Printsrv/printway:** This directory contains temporary files that IP PrintWay extended mode writes before transmitting data to printers.

IP PrintWay basic mode never writes temporary files in this directory. Instead, IP PrintWay basic mode writes temporary files in hiperspace.

- ▶ **/var/Printsrv/logs** and **/var/Printsrv/logdb**: These directories contain message logs and the common message log database.
- ▶ **/var/Printsrv/hinv**: This directory contains the Historical Inventory.
- ▶ **/var/Printsrv/searchdb**: This directory contains the search database.
- ▶ **/var/Printsrv/trace**: This directory contains trace information when you activate tracing with the AOPTRACEON environment variable. IP PrintWay basic mode does not write trace information to this directory.
- ▶ **/var/Printsrv/xfd** and **/var/Printsrv/xf\_sockets**: These directories and their subdirectories contain:
  - Temporary files and messages for Infoprint transform products.
  - Language environment dumps (CEEDUMPs). When you stop and restart the Transform Manager, these CEEDUMPs are moved to the /var/Printsrv directory.

Allow enough space to contain all data sets that can be transformed at the same time.

**Important:** Allocate at least 750 MB of DASD space for the file system mounted at the /var/Printsrv mount point. If you do not run IP PrintWay extended mode or use data stream transforms, you might need less space. If you print extremely large files or if you want to keep many days worth of messages and historical information, you might need more space.

## 3.11 Printer Inventory and security

- ❑ Infoprint Server administrators and operators
  - Define a groups
    - GROUP - AOOPER
    - GROUP - AOPADMIN
- ❑ Define RACF administrators access(user IDs or group IDs)
  - Using the RACF PRINTSRV class AOP.ADMINISTRATOR resource

```
RDEFINE PRINTSRV (AOP.ADMINISTRATOR) UACC(NONE)
SETROPTS CLASSACT(PRINTSRV) RACLST(PRINTSRV)
SETROPTS RACLST(PRINTSRV) REFRESH

PERMIT AOP.ADMINISTRATOR CLASS(PRINTSRV) ACCESS(UPDATE) ID(AOPADMIN)
or
PERMIT AOP.ADMINISTRATOR CLASS(PRINTSRV) ACCESS(UPDATE) ID(userID)
SETROPTS RACLST(PRINTSRV) REFRESH
```

Figure 3-11 Defining access security for the Printer Inventory

### Security for the Printer Inventory

The AOP.ADMINISTRATOR profile in the PRINTSRV class protects the Printer Inventory:

- ▶ READ access lets users view the Printer Inventory using ISPF panels, Infoprint Central, or the Printer Inventory Definition Utility (pidu). Users do not need READ access simply to list names of printer definitions with the **lpstat** command or with Infoprint Port Monitor.
- ▶ UPDATE access lets users update the Printer Inventory using ISPF panels or the **pidu** command and lets users view the Printer Inventory using Infoprint Central.

To protect access to the Printer Inventory, the common message log, and daemons, you must define the AOPADMIN and AOOPER groups to RACF for Infoprint Server administrators and operators.

## 3.12 Create users for administration and operation

- ❑ **Sample CLIST: SYS1.SAMPLIB(AOPRACF)**
  - **AOPRACF sample TSO CLIST provides all the RACF commands needed for the setup of Infoprint Server.**
    - Group IDs and user IDs (OMVS segments)
    - Default groups: AOPADMIN and AOPOPER
    - Create HOME directory and shell access
    - Does NOT create HFS or zFS (user file system) for each user ID

Figure 3-12 Create users to administer and operate Infoprint Server

### Defining users for administration and operation - AOPRACF CLIST

An Infoprint Server administrator can view and update the Printer Inventory and can view all messages in the common message log. The suggested group name for administrators is AOPADMIN. However, you can use any name. Additionally, Infoprint Server operators can start and stop Infoprint Server daemons using the **aopstart** and **aopstop** commands or the AOPSTART and AOPSTOP JCL procedures. The suggested group name for Infoprint Server operators is AOPOPER. However, you can use any name.

```
ADDGROUP (AOPADMIN) OMVS(GID(yy))  
ADDGROUP (AOPOPER) OMVS(GID(zz))
```

The AOPADMIN and AOPOPER groups must have an OMVS segment and an OMVS group identifier (GID).

**Note:** You can define as many administrator user IDs and group IDs as required in the AOPRACF CLIST by adding them to the existing commands.

The sample AOPRACF TSO CLIST provides all the RACF commands needed for the setup of Infoprint Server. This CLIST is structured into separate sections (blocks), each with a set of block comments, describing the RACF function that will be set up and what needs to be changed by the customer. The AOPRACF CLIST must be executed by a RACF special user ID.

**Note:** AOPRACF is neither a JCL procedure nor a complete CLIST. All of the commands have been commented out to prevent accidental execution. Customers must uncomment the necessary lines before running this CLIST to perform the respective commands.

Each AOPRACF CLIST section is intended to work independently of other sections. Therefore it is possible that some of these commands will return a non-zero return code since the specific RACF function may already be set up:

- ▶ Block 1 - Define a group for Infoprint Server administrators and a group for Infoprint Server operators. Suggested group names are AOPADMIN and AOPOPER.
- ▶ Block 2 - Set up a profile named AOP.ADMINISTRATOR in the PRINTSRV class and activate the PRINTSRV class. The resource profile must be named AOP.ADMINISTRATOR.
- ▶ Block 3 - Give the Infoprint Server administrator group access to the AOP.ADMINISTRATOR resource profile in the PRINTSRV class. This will allow print administrators to view or update the Printer Inventory using the ISPF panels or the Printer Inventory Definition Utility (pidu) program.
- ▶ Block 4 - Optionally, define Infoprint Server administrator and operator user IDs as UNIX users. The user IDs for Infoprint Server administrators and operators must have an OMVS segment with a user identifier (UID) and be connected to a default group that has a group identifier (GID).
- ▶ Block 5 - Connect administrators to AOPADMIN group and connect operators who will start and stop Infoprint Server to the AOPOPER group.
- ▶ Block 6 - Define user ID to be associated with AOPSTART and AOPSTOP started JCL procedures. The user ID associated with the AOPSTART and AOPSTOP started procedures must have an OMVS segment with a user identifier (UID) and be connected to a default group that has a group identifier (GID).
- ▶ Block 7 - Define AOPSTART and AOPSTOP JCL procedures to the RACF STARTED class. Instead of the RACF STARTED class definitions, the RACF started procedure table (ICHRIN03) can be used for the AOPSTART and AOPSTOP JCL procedures.
- ▶ Block 8 - Give universal RACF access to the Infoprint Server ISPF data sets. This lets all users view the Infoprint Server ISPF panels.



### 3.13 SYS1.SAMPLIB(AOPRACF) commands

```
/* Block 1 - Define the AOPADMIN and AOPOPER group entries. */
ADDGROUP (AOPADMIN) OMVS(GID(100))
ADDGROUP (AOPOPER) OMVS(GID(101))
/* Block 2 - Define resource profile in PRINTSRV class */
RDEFINE PRINTSRV (AOP.ADMINISTRATOR) UACC(NONE)
SETROPTS CLASSACT(PRINTSRV) RACLIST(PRINTSRV)
SETROPTS RACLIST(PRINTSRV) REFRESH
/* Block 3 - Give administrator group access to AOP.ADMINISTRATOR */
PERMIT AOP.ADMINISTRATOR CLASS(PRINTSRV) ACCESS(UPDATE) ID(AOPADMIN)
SETROPTS RACLIST(PRINTSRV) REFRESH
/* Block 4 - Define Infoprint Server administrator and operator */
ALTUSER (AOPADM1) OMVS(UID(0) HOME('/u/aopadm1') PROGRAM('/bin/sh')) DFLTGRP (AOPADMIN)
ALTUSER (AOPOP1) OMVS(UID(0) HOME('/u/aopop1') PROGRAM('/bin/sh')) DFLTGRP (AOPOPER)
/* Block 5 - Connect administrators and operators to respective groups */
CONNECT (AOPADM1) GROUP(AOPADMIN)
CONNECT (AOPOP1) GROUP(AOPOPER)
/* Block 6 - Define user ID for AOPSTART and AOPSTOP started procedures*/
ADDUSER AOPSTC OMVS(UID(777) HOME('/u/aopstc') +
PROGRAM('/bin/sh')) DFLTGRP (AOPOPER) NOPASSWORD
/* Block 7 - Define AOPSTART and AOPSTOP to the RACF STARTED class*/
SETROPTS GENERIC(STARTED)
RDEFINE STARTED AOPSTART.* STDATA(USER(AOPSTC) GROUP (AOPOPER))
RDEFINE STARTED AOPSTOP.* STDATA(USER(AOPSTC) GROUP (AOPOPER))
SETROPTS CLASSACT(STARTED) RACLIST(STARTED)
SETROPTS RACLIST(STARTED) REFRESH
/* Block 8 - Give users access to Infoprint Server ISPF data sets */
ADDGROUP (AOP) SUPGROUP(SYS1) OWNER(SYS1)
ADDSD 'AOP.SAOP*' GENERIC OWNER(AOP) UACC(READ)
```

← User ID to start and stop  
Infoprint Server

Figure 3-13 AOPRACF CLIST after modification

#### Modifying the AOPRACF CLIST

The sample CLIST in SYS1.SAMPLIB(AOPRACF) provides all the RACF commands needed for the setup of Infoprint Server. If you use AOPRACF, you still should create the home directory for the user IDs. Figure 3-13 shows the modification in bold text. Define Infoprint Server administrators and operators to the security product as z/OS UNIX users. An Infoprint Server administrator must be able to read and update the Printer Inventory. Use the AOPADMIN RACF group name for administrators; however, you can use any name. Connect Infoprint Server administrators to the collection (hereafter called group) of Infoprint Server administrators. You must modify AOPRACF before you run it, as follows:

- ▶ Define a group for Infoprint Server administrators and a group for Infoprint Server operators. The groups are AOPADMIN and AOPOPER respectively.
- ▶ Define AOP.ADMINISTRATOR resource profile in PRINTSRV class and activates the PRINTSRV class.
- ▶ Give the Infoprint Server administrator group access to the AOP.ADMINISTRATOR resource profile in the PRINTSRV class. This will allow print administrators to view or update the Printer Inventory using the ISPF panels or the Printer Inventory Definition Utility (pidu) program.

**Note:** You could permit each user separately instead of permitting the AOPADMIN group.

## CONNECT command

Connect the user IDs for Infoprint Server administrators to the AOPADMIN group, and connect the user IDs for operators who will start and stop Infoprint Server to the AOOPER group.

## OMVS Segment

The user IDs for Infoprint Server administrators and operators must have an OMVS segment with a user identifier (UID) and be connected to a default group that has a group identifier (GID). The prototype RACF ALTUSER command in AOPRACF is:

```
ALTUSER (aaaaaaaa) OMVS(UID(uu) HOME('/u/aaaaaaaa')  
PROGRAM('/bin/sh'))DFLTGRP(gggggggg)
```

- Change ALTUSER to ADDUSER if this user ID is new to the database.
- aaaaaaaaa and oooooooooo- refers to the user ID.
- uu - refers to a unique UID number between 1 - 16,777,215
- gggggggg - refers to the default group name. This could be AOPADMIN, AOOPER, or any other previously defined group that has an OMVS GID.

## AOPSTC user ID for starting Infoprint Server

Define user ID AOPSTC to be associated with the AOPSTART and AOPSTOP started procedures. The user ID associated with the AOPSTART and AOPSTOP started procedures must have an OMVS segment with a user identifier (UID) and be connected to a default group that has group identifier (GID), as an example follows:

- ▶ AOPSTC user ID is associated with the started procedures
- ▶ The AOPSTC user ID is defined as a PROTECTED user ID by using the NOPASSWORD operand. PROTECTED user IDs cannot be used to logon to the system, and are protected from being revoked through incorrect password attempts.
- ▶ Define AOPSTART and AOPSTOP cataloged procedures the RACF STARTED class.
- ▶ Create a profile for the extended MCS console in the OPERCMDS class.
- ▶ Give AOOPER group access to the EMCS console used by Infoprint Server.

**Note:** If you do not use the AOPRACF CLIST, you must then define a user ID for starting and stopping Infoprint Server, just as shown in the CLIST in Figure 3-13 on page 91.

If you use the AOPSTART and AOPSTOP procedures, you must define profiles for them in the RACF STARTED class. The following commands assume that the user ID you defined for use with these procedures is AOPSTC and that AOPSTC is connected to group AOOPER:

```
CONNECT (AOPSTC) GROUP(AOOPER)  
RDEFINE STARTED AOPSTART.* STDATA(USER(AOPSTC) GROUP(AOOPER))  
RDEFINE STARTED AOPSTOP.* STDATA(USER(AOPSTC) GROUP(AOOPER))  
SETROPTS RACLIST(STARTED) REFRESH
```

## Define OMVS segment

The user profiles of the Infoprint Server administrators and operators must have an OMVS segment and a home directory. Establish an OMVS segment and define the home directory to RACF using the RACF ADDUSER command to add a new user or the ALTUSER command to change an existing user, or whatever methods are currently in use to do security definitions.

## 3.14 Create a user file system

- ❑ z/OS UNIX users often require a file systems of their own
  - Allocate a zFS data set for users:
    - AOPADM1 and AOPSTC
  - Mount the file systems - if automount not used
    - Use the ISHELL mount panel - or - `usr/sbin/mount REXX exec`
    - Use the TSO/E mount command - `MOUNT + FILESYSTEM('OMVS.AOPSTC.ZFS') TYPE(ZFS) + MOUNTPOINT('/u/aopstc')`
  - Automount facility (If using /u)
    - Go to ISHELL - Enter `/u/userID` - Press Enter
    - Now the file is automatically mounted
    - Add an entry to the BPXPRMxx member in SYS1.PARMLIB

Figure 3-14 Allocating and mounting of a user HFS file system

### Allocate a zFS file system for users

Following is a sample JCL to allocate a file system.

```
//STEP01 EXEC PGM=IEFBR14
//ZFS DD DSN=OMVS.AOPSTC.ZFS,SPACE=(CYL,(5,1,1)),
// DSNTYPE=ZFS,DCB=(DSORG=PO),
// DISP=(NEW,CATLG,DELETE),
// STORCLAS=OPENMVS
```

A user file system is allocated in exactly the same way that you created the root file system. Choose a data set name that has the user name as one of the qualifiers and a size that provides sufficient space for the user's requirements.

You can allocate a zFS file system using Option 4 from the File\_systems pull-down list in an ISHELL.

After the user's file system is allocated, you need to mount it at a mount point off the root directory to make it available. The preferred place to mount all user file systems is a user directory under the /u user directory. z/OS TSO/E MOUNT command to accomplish this:

```
MOUNT FILESYSTEM('OMVS.AOPSTC.ZFS') TYPE(ZFS) MOUNTPOINT('/u/aopstc')
```

File		Directory		Special_file		Tools		File_systems		Options		Setup		Help					
Mount a File System																			
Mount point:														More:		+			
_____																			
_____																			
File system name _____																			
File system type _____				New owner . . . . . _____															
Owning system . . _____				Character Set ID . . _____															
Select additional mount options:																			
- Read-only file system				- Set automove attribute...															
- Ignore SETUID and SETGID				- Text conversion enabled															
- Bypass security																			
Mount parameter:																			
_____																			
F1=Help				F3=Exit				F4=Name				F6=Keyshelp				F12=Cancel			

Figure 3-15 UNIX System Services ISPF Shell Mount Panel

### Automount facility

You can allocate the user's home directory to a user HFS data set and use the automount facility to manage the user HFS data set. The automount facility lets you designate directories as containing only mount points. As each mount point is accessed, an appropriate file system is mounted.

If you want to make the mounting of the OMVS.AOPSTC.ZFS file system permanent, you have to add an entry in the BPXPRMxx member in SYS1.PARMLIB. The mount statement should follow the ROOT statement for the root file system:

```

MOUNT    FILESYSTEM('OMVS.AOPSTC.ZFS')
          TYPE(ZFS)
          MOUNTPPOINT('/u')
          MODE(RDWR)

```

## 3.15 Starting Printer Inventory Manager

- ❑ Make sure AOPSETUP shell script is run - OMVS shell
  - `aopsetup AOOPER AOPADMIN`
    - Sets permissions for for directories and executable files
- ❑ Start from an OMVS shell
  - `aopstart`
  - Starts Printer Inventory (AOPD) -- `start-daemons = { }`
- ❑ Start from an operator console
  - `S AOPSTART`
  - Starts Printer Inventory (AOPD) -- `start-daemons = { }`
- ❑ Users must be authorized
  - Member of AOOPER group or UID 0 or access to BPX.SUPERUSER profile in FACILITY class

Figure 3-16 Different ways to start Infoprint Server

## Starting Printer Inventory Manager

You must run the *aopsetup* shell script to set up the correct z/OS UNIX permissions for Infoprint Server directories and executable files before you start Infoprint Server daemons for the first time. Also, you must rerun *aopsetup* whenever you move to a new z/OS release.

The *aopsetup* shell script sets z/OS UNIX permissions for:

- ▶ **/var/Printsrv directory**

If the /var/Printsrv directory does not exist, *aopsetup* creates it. Also, *aopsetup* sets permissions so that this directory is readable and writeable only by members of the RACF group for Infoprint Server administrators (AOPADMIN) and users with effective UID of 0.

If you specify a different directory in the base-directory attribute in the Infoprint Server configuration file (*aopd.conf*), *aopsetup* creates that directory instead of the /var/Printsrv directory.

- ▶ **Files in the /usr/lpp/Printsrv/bin directory**

This directory contains Infoprint Server executable files; *aopsetup* sets file permissions so that:

- Operator commands, such as *aopstart*, are executable only by members of the RACF group for Infoprint Server operators (AOOPER) and users with an effective UID of 0.

- Administrative commands, such as **pidu**, are executable only by members of the RACF group for Infoprint Server administrators (AOPADMIN) and users with an effective UID of 0.
- The Printer Inventory Manager daemon, **aopd**, has the set-group-ID bit on.

The **aopsetup** shell script requires two positional arguments:

```
aopsetup operator-group administrator-group
```

- *operator-group* is the name of the RACF group you created for Infoprint Server operators.
- *administrator-group* is the name of the RACF group you created for Infoprint Server administrators.

**Example:** If you defined one group to RACF named AOPOPER for operators and another group named AOPADMIN for administrators, enter:

```
/usr/lpp/Printsrv/bin/aopsetup AOPOPER AOPADMIN
```

If you did not define a separate group for Infoprint operators, specify the group name for Infoprint Server administrators (for example, AOPADMIN) in both arguments.

**Rule:** Run **aopsetup** from a user ID with a UID of 0. To switch to an effective UID of 0, you can use the z/OS UNIX **su** command if you are permitted to the BPX.SUPERUSER profile in the FACILITY class in RACF.

### **aopstart command from an OMVS shell**

The **aopstart** command is a REXX EXEC that sets environment variables that Infoprint Server daemons use. It also starts the Printer Inventory daemon and any other daemons specified in the **start-daemons** attribute in the Infoprint Server configuration file (**aopd.conf**). If a daemon is already started, **aopstart** does not start that daemon again.

While it is not very probable that an operator would enter the OMVS shell and enter the **aopstart** command, this command can be used to start the Infoprint Server. The two conditions needed by an operator to be able to do this are:

- ▶ The operator's user ID must have an OMVS segment in the RACF user profile.
- ▶ The operator must enter the OMVS shell by using Telnet.

### **Operator start command from an MVS console**

The S AOPSTART operator command invokes the AOPSTART procedure provided by Infoprint Server. The procedure invokes AOPBATCH program, which invokes the **aopstart** shell command to start the Printer Inventory Manager daemon (**aopd**) and any other daemons that are specified in the **start-daemons** attribute in the **aopd.conf** configuration file.

### **Command security**

To use the **aopstart** command, you must either be a member of the RACF AOPOPER group or have a UID of 0. You can use the z/OS UNIX **su** command to switch to an effective UID of 0 if the user ID has access to the BPX.SUPERUSER profile in the RACF FACILITY class.

## 3.16 AOPSTART JCL procedure

```
/* AOPSTART - This procedure starts the Infoprint Server daemons.
/*          The USERID assigned to this proc must be a member of
/*          the AOPOPER group.
/*-----
//AOPSTART PROC
/* AOPBATCH PARM specifies the path of the program to be run.
/* If Infoprint Server was not installed in /usr/lpp/Printsrv,
/* then this path must be changed.
/*-----
//AOPSTART EXEC PGM=AOPBATCH,PARM='/usr/lpp/Printsrv/bin/aopstart',
//          TIME=NOLIMIT,REGION=0M
//STDOUT    DD    SYSOUT=*
//STDERR    DD    SYSOUT=*
//STDENV    DD    PATH='/var/Printsrv/env.conf',PATHOPTS=ORDONLY
/* STDENV is optionally used to specify environment variables.
/* Uncomment STDENV if you need to trace the environment
/* A Unix System Services file can be specified, for example:
/*    //STDENV DD PATH='/etc/Printsrv/env.conf',PATHOPTS=ORDONLY
/* or an MVS data set, for example:
/*    //STDENV DD DISP=SHR,DSN=hlq.stdenv
/* This file or data set should contain VARIABLE=VALUE pairs
/* starting in column 1, for example:  AOPTRACEON=1
```

Figure 3-17 AOPSTART JCL procedure

### AOPSTART JCL procedure

The AOPSTART procedure provided in SYS1.PROCLIB invokes the aopstart command to start Infoprint Server daemons specified in the start-daemons attribute in the aopd.conf configuration file.

You must edit the AOPSTART procedure if any of these conditions apply to your installation:

- ▶ The default region size set by your installation is not sufficient. Specify the region size in the REGION parameter on the EXEC statement:
  - If you start the IPP server, specify at least 200 MB.
  - If you start the Transform Manager, specify at least 10K more than the value specified in the -M option in the transform configuration file. If you specify too low a region size, you might experience transform failures when Infoprint Server processes large or complicated data streams. IBM recommends at least 256 MB.
- ▶ The aopstart command is not in the default directory /usr/lpp/Printsrv/bin. Specify the directory in the PARM parameter on the EXEC statement.

You can specify any of these environment variables in a data set referred to by the STDENV DD statement:

- ▶ AOPTRACEON=1: To trace Infoprint Server daemons. By default, tracing is off.
- ▶ AOPTRACEON\_DATA=1, AOPTRACEON\_FILTER=1, AOPTRACEON\_MSGLOG=1: To obtain additional trace data.

- ▶ AOPTRACEBYTES, AOPTRACEMAXFILES: To specify how much trace information is recorded.
- ▶ AOPVALIDATEDB: To check the validity of Infoprint Server databases when starting Infoprint Server daemons.
- ▶ LANG: To view Infoprint Server daemon messages in Japanese (LANG=Ja\_JP). By default, messages are written in English.
- ▶ TZ: To specify the time zone for messages in the common message log that Infoprint Central displays. Specify the same value in the HTTP Server environment variables file. The default time zone for common log messages is GMT0.
- ▶ LC\_TIME or LC\_ALL: To specify the locale used to format the date and time in messages in the common message log that Infoprint Central displays. Specify the same value in the HTTP Server environment variables file. The default locale is C.
- ▶ LC\_CTYPE: To specify the locale used to determine the EBCDIC code page for validating attribute values. Specify the same locale in the Language field in the Infoprint Server ISPF panels. If you use Infoprint Central, specify this environment variable with the same value in the HTTP Server environment variables file. The default locale is C.

You must specify all other environment variables in the **aopstart** EXEC.

(The MVS data set referred to by the STDENV DD statement must have these B attributes: RECFM=VB and LRECL=255. For more information about how to define an environment file, see the comments in the AOPSTART procedure.)

### Starting Infoprint Server daemons automatically

To start Infoprint Server daemons automatically during system initialization, you can use one of these methods:

- ▶ z/OS automation tools to execute the AOPSTART procedure.
- ▶ The /etc/rc shell script to run the **aopstart** command. For example, if the aopstart EXEC is in the default directory, add the following command to /etc/rc:

```
/usr/lpp/Printsrv/bin/aopstart
```



## 3.17 Environment variables in /etc/profile

```
# Set the STEPLIB
if Y -z "$STEPLIB" `` && tty -s;
then
    export STEPLIB=none
    exec sh -L
fi
# Set the time zone as appropriate.
TZ=EST5EDT
# Set a default command path, including current working dir (CDW)
export PATH=/usr/lpp/Printsrv/bin:$PATH
# Sets other path environment variables
export LIBPATH=/usr/lpp/Printsrv/lib:$LIBPATH
export MANPATH=/usr/lpp/Printsrv/man/En_US:$MANPATH
export NLSPATH=/usr/lpp/Printsrv/%L/%N:/usr/lpp/Printsrv/En_US/%N:$NLSPATH
AOPPATH=/etc/Printsrv/aopd.conf
# Sets the language
LANG=C
# Sets the name of the system mail file and enables mail notification.
MAIL=/usr/mail/$LOGNAME
# Export the values so the system will have access to them.
export TZ MAIL LANG
export AOPCONF
# Set the default file creation mask - umask
umask 022
# Set the LOGNAME variable readonly so it is not accidentally modified.
readonly LOGNAME
```

Figure 3-18 Environment variables set in /etc/profile

### Environment variables

Infoprint Server uses environment variables to define its environment in the z/OS system. The Printer Inventory Manager, as well as other Infoprint Server daemons, use environment variables specified in the aopstart EXEC. In addition, the daemons accept certain variables from the run-time environment, such as environment variables that control tracing, locale, and the language for messages. The aopstart EXEC lists the variables that Infoprint Server accepts from the run-time environment. If you use the AOPSTART JCL procedure to start Infoprint Server daemons, you must specify these run-time variables in the STDENV data set of the AOPSTART JCL procedure. However, if you use the aopstart command in a UNIX System Services shell to start Infoprint Server daemons, the daemons accept certain environment variables from the shell. The /etc/profile file contains environment variables that apply to most shells.

Infoprint Server and Infoprint Server Transforms commands, such as **aoplogu**, **pidu**, and **ps2afp**, accept environment variables from the /etc/profile file. In general, all shell users will get the environment variables set by /etc/profile. Users can override some of the variables by setting the variables in their private profiles, or in their shell sessions.

### /etc/profile customization

To edit the /etc/profile file, use either the TSO/E OEDIT command or the z/OS UNIX **oedit** command. To set and export an environment variable, use the z/OS UNIX **export** command.

The /etc/profile file contains the environment variables and commands used by most shell users. An IBM-supplied sample is located in /samples/profile. Copy the sample to the /etc directory and make any necessary changes for the Infoprint Server:

- If you installed Infoprint Server libraries in the default locations, add these z/OS UNIX **export** commands to the /etc/profile file:

```
export LIBPATH=/usr/lpp/Printsrv/lib:$LIBPATH
export NLSPATH=/usr/lpp/Printsrv/%L/%N:/usr/lpp/Printsrv/En_US/%N:$NLSPATH
export PATH=/usr/lpp/Printsrv/bin:$PATH
export MANPATH=/usr/lpp/Printsrv/man/En_US:$MANPATH
```

- To display the value of an environment variable, use the z/OS UNIX **echo** command:

```
echo $PATH
```

**PATH** The path used to locate executables. This environment variable is required. If you installed Infoprint Server executables in the default directory, add /usr/lpp/Printsrv/bin to the existing values. Be sure to add the directory before /bin in the PATH environment variable to make sure that the Infoprint Server versions of the **lp**, **lpstat**, and **cancel** commands are invoked.

Default: /usr/lpp/Printsrv/bin:/bin  
(This default applies only to the aopstart EXEC.)

**LIBPATH** The path used to find dynamic link library (DLL) files. This environment variable is required. If you installed Infoprint Server files in the default directory, add /usr/lpp/Printsrv/lib to any existing values. If you installed Infoprint Server files in a different directory, add the directory to any existing values.

Default: /usr/lpp/Printsrv/lib  
(This default applies only to the aopstart EXEC.)

**NLSPATH** The path of directories that contain message catalogs. This environment variable is required. If the LANG environment variable identifies the language in which you want to receive Infoprint Server messages, add /usr/lpp/Printsrv/%L/%N to the values in this variable. Otherwise, add one of these values:

English: /usr/lpp/Printsrv/En\_US/%N  
Japanese: /usr/lpp/Printsrv/Ja\_JP/%N

%L represents the value of the LANG environment variable.  
%N is the catalog filename.

Default: /usr/lpp/Printsrv/%L/%N:/usr/lpp/Printsrv/En\_US/%N:  
/usr/lib/nls/msg/%L/%N  
(This default applies only to the aopstart EXEC.)

**MANPATH** The path of directories that contain the man pages. This environment variable is required. If the LANG environment variable identifies the language in which you want to view Infoprint Server man pages, add /usr/lpp/Printsrv/man/%L to the values in this variable; otherwise, add one of the following values to any existing values:

English: /usr/lpp/Printsrv/man/En\_US  
Japanese: /usr/lpp/Printsrv/man/Ja\_JP  
Spanish: /usr/lpp/Printsrv/man/Es\_ES

**Note:** Add the new directory path before /usr/man/%L in the MANPATH environment variable, to display the Infoprint Server versions of the **lp**, **lpstat**, and **cancel** man pages.

### 3.18 Additional environment variables in /etc/profile

❑ Variables are optional if you specify default paths

- **AOPCONF** - /etc/Printsrv/aopd.conf
- **AOPMSG\_CONF** - /etc/Printsrv/aopmsg.conf
- **AOPVALIDATEDB**
- **LANG**
- **LC\_ALL**
- **LC\_CTYPE**
- **LC\_TIME**
- **TZ**

Figure 3-19 Additional environment variables in /etc/profile

#### Environment variables

Following is a summary of the variables that affect the Printer Inventory Manager, including the common message log and the **pidu** command. Required variables must be set. Optional variables must be set only if the default value is not suitable for your installation.

##### **AOPCONF**

The full path name of the Infoprint Server configuration file. This environment variable is optional. If you created the configuration file in /etc/Printsrv/aopd.conf, you do not need to set this environment variable.

Default: /etc/Printsrv/aopd.conf

##### **AOPMSG\_CONF**

The full path name of the Infoprint Server message configuration file, aopmsg.conf. This environment variable is optional. If you did not create the configuration file or if you created the configuration file in /etc/Printsrv/aopmsg.conf, you do not need to set this environment variable.

Default: /etc/Printsrv/aopmsg.conf

##### **AOPVALIDATEDB**

When set to any value, Infoprint Server checks internal databases for validity when it starts daemons. Infoprint Server starts daemons when you run the **aopstart** command or AOPSTART JCL procedure. This environment variable is optional. Specify it only if an Infoprint Server message indicates that the databases might be corrupted or if instructed to do so by IBM service personnel.

Default: None

<b>LANG</b>	<p>The language used for messages. Infoprint Server provides messages in English and Japanese. Specify En_US for English messages, or Ja_JP for Japanese messages.</p> <p>Default: C (equivalent to En_US)</p>
<b>LC_ALL</b>	<p>The locale used to format time and date information in messages and the language of messages. Specify the same LC_ALL value in the environments for the Printer Inventory Manager and Infoprint Central. This locale overrides the locale in the LC_TIME variable.</p> <p>Default: C (also called POSIX)</p>
<b>LC_CTYPE</b>	<p>The locale that determines the EBCDIC code page used to validate Infoprint Server attribute values. Specify the same LC_CTYPE value in the environments for the Printer Inventory Manager and Infoprint Central. Also, specify a comparable value in the Language field on the Infoprint Server ISPF Configuration panel.</p> <p>Tip: LC_ALL does not override LC_CTYPE.</p> <p>Default: C (also called POSIX)</p>
<b>LC_TIME</b>	<p>The locale used to format time and date information in messages. Specify the same LC_TIME value in the environments for the Printer Inventory Manager and Infoprint Central.</p> <p>Default: C (also called POSIX)</p>
<b>TZ</b>	<p>The time zone used to format date and time information in messages displayed by Infoprint Central. Specify the same TZ value in the environments for the Printer Inventory Manager and Infoprint Central.</p> <p>For information about the allowed values, see <i>z/OS UNIX System Services Command Reference</i>, SA22-7802.</p> <p>Default: The z/OS HTTP Server default is EST5EDT (Eastern U.S. time zone). The default for the Printer Inventory Manager is GMT0.</p>

## 3.19 Editing aopstart rexx exec

```
/* rexx
*/
/* aopstart is a REXX exec used to establish a secure environment in which
/* to start the Infoprint Server daemons.
/*
/*
/* This exec assumes a default installation environment. If this exec is
/* to be used in a non-default environment, then it should be copied to
/* another directory (for example /usr/sbin), and edited to accommodate the
/* environment changes.
/*
/* Items that may need to be changed include:
/*
/* REXX EXEC VARIABLES:
/* install_path <-- edit this variable's value if Infoprint Server
/* was installed in a non-default directory.
/*
/* SECURE ENVIRONMENT VARIABLES:
/* PATH AOPCONF
/* LIBPATH AOPXFD_CONF
/* CLASSPATH AOPTRACEDIR
/* NLSPATH BPX_SHAREAS
/* JAVA_HOME _BPX_UNLIMITED_OUTPUT
/* OTHER VARIABLES THAT MAY NEED TO BE DEFINED:
/* STEPLIB AOPSPD_CONF
/* _BPXK_SETIBMOPT_TRANSPORT AOP_APPLID
/* AOP_SAP2AFP_RESOURCES
/* It is recommended that nothing else be changed in this exec.
```

Figure 3-20 aopstart rexx exec

### aopstart REXX EXEC

The **aopstart** command is a REXX EXEC in `/usr/lpp/Printsrv/bin/`. It sets environment variables used by Infoprint Server daemons and starts the Printer Inventory daemon and any other daemons specified in the `start-daemons` attribute in the Infoprint Server configuration file (`aopd.conf`).

The **aopstart** EXEC provided with Infoprint Server is suitable for most installations. However, you can edit the EXEC to change the environment variables for your installation.

Edit the **aopstart** EXEC when:

- ▶ Infoprint Server files were not installed in the default directory `/usr/lpp/Printsrv`
  - Edit: `LIBPATH`, `NLSPATH`, `PATH` `CLASSPATH`
- ▶ The Infoprint Server configuration file is not in the default location `/etc/Printsrv/aopd.conf`
  - Edit: `AOPCONF`
- ▶ The LE and C++ libraries are not in the system `LNKLST`
  - Edit: `STEPLIB`
- ▶ You run the Print Interface LPD or IP PrintWay extended mode, and you have set up multiple TCP/IP stacks on your system
  - Edit: `_BPXK_SETIBMOPT_TRANSPORT`

- ▶ The transform configuration file is not in the default location /etc/Printsrv/aopxfd.conf
  - Edit: AOPXFD\_CONF
- ▶ You run the Infoprint Server SAP Output Management System (OMS)
  - Edit: LIBPATH
- ▶ You run the SAP OMS, and the SAP Callback configuration file is not in the default location /etc/Printsrv/aopsapd.conf
  - Edit: AOPSAPD\_CONF
- ▶ You run the SAP to AFP transform, and transform resources are not in the default location /usr/lpp/Printsrv/sap2afp
  - Edit: AOP\_SAP2AFP\_RESOURCES
- ▶ You run the Print Interface IPP server, and Java™ was not installed in the default directory /usr/lpp/java/J1.4
  - Edit: JAVA\_HOME
- ▶ You do not want Infoprint Server daemon trace files to be written to the default directory /var/Printsrv/trace
  - Edit: AOPTRACEDIR
- ▶ You want IP PrintWay extended mode to remove blanks from the end of lines and records for line data and text data.
  - Edit: AOP\_BLANK\_TRUNCATION\_CLASSES

## 3.20 Stopping Infoprint Server daemons

- ❑ From an OMVS session
  - **aopstop**
    - Stops all daemons
  - **aopstop -d xfd -d lpd**
    - Stops specified daemons
- ❑ Operator commands to stop
  - **S AOPSTOP**
    - Stops **AOPD** and all **start-daemons**
  - **S AOPSTOP,OPTIONS='-d xfd -d lpd now'**
    - Stops specified daemons
- ❑ Users must be authorized
  - Member of **AOOPER** group or **UID 0** or access to **BPX.SUPERUSER** profile in **FACILITY** class

Figure 3-21 Commands to stop Infoprint Server daemons

### Stopping Infoprint Server

Infoprint Server operators can stop Infoprint Server daemons using the **aopstop** command or the AOPSTOP procedure.

To stop the Printer Inventory Manager daemon, as well as all other Infoprint Server daemons, enter the **aopstop** command in a UNIX shell without specifying any daemon names. The Print Interface component of Infoprint Server stops accepting new print requests and completes all work in progress before shutting down.

To use the **aopstop** command, you must either:

- ▶ Be a member of the Security Server RACF **AOOPER** group, or
- ▶ Have a **UID** of 0

You can use the z/OS UNIX **su** command to switch to an effective **UID** of 0 if you have access to the **BPX.SUPERUSER** profile in the RACF **FACILITY** class.

**Note:** Before you stop the Printer Inventory Manager daemon, stop other programs that are using the Printer Inventory, such as NetSpool, IP PrintWay, and PSF for z/OS.

To stop all active daemons (including the Printer Inventory Manager daemon) after current activity ends, use this command from the OMVS shell:

```
aopstop
```

The format of the aopstop command is as follows:

```
aopstop [-d daemon]... [now] [force]
```

The arguments have the following meanings:

-d daemon	Specifies the daemon to stop.
now	Stops the daemons immediately.
force	Stops the daemons immediately using a “destructive” shutdown. If work is in progress, incorrect output or data loss might result. For daemons that manage a database, such as aopd, aophinvd, or aopsdbd, the database might be corrupted.

To stop the Transform Manager and line printer daemons, use this command from the shell:

```
aopstop -d xfd -d lpd
```

To stop all daemons immediately, use this command. Documents that are being processed may be lost.

```
aopstop now
```

### Using the AOPSTOP procedure

When the operator invokes this procedure without any options, all active daemons stop when current activity ends. The operator can specify **aopstop** command options to stop daemons immediately and to stop selected daemons. For example, to stop the Transform Manager and line printer daemons immediately, specify this MVS START command:

```
START AOPSTOP OPTIONS='-d xfd -d lpd now'
```



## 3.21 AOPSTOP JCL procedure

```
/** AOPSTOP - This procedure stops the Infoprint Server daemons.
/**          The USERID assigned to this proc should be a member
/**          of the AOPOPER group.
/**-----
//AOPSTOP  PROC  OPTIONS=
/**-----
/** AOPBATCH PARM specifies the path of the program to be run.
/** If Infoprint Server was not installed in /usr/lpp/Printsrv,
/** then this path should be changed as required for the installation
/** directory.
/**-----
//AOPSTOP  EXEC  PGM=AOPBATCH,
//          PARM='/usr/lpp/Printsrv/bin/aopstop &OPTIONS'
//STDOUT   DD    SYSOUT=*
//STDERR   DD    SYSOUT=*
/**-----
/** STDENV is optionally used to specify environment variables.
/** A Unix System Services file can be specified, for example:
/**    //STDENV DD  PATH='/etc/Printsrv/env.conf',PATHOPTS=ORDONLY
/** or an MVS data set, for example:
/**    //STDENV DD  DISP=SHR,DSN=hlq.stdenv
/** This file or data set should contain VARIABLE=VALUE pairs
/** starting in column 1, for example:
/** AOPCONF=/etc/Printsrv/aopd.conf
```

Figure 3-22 AOPSTOP JCL procedure

### AOPSTOP JCL procedure

The AOPSTOP procedure provided in SYS1.PROCLIB invokes the **aopstop** command to stop some or all active Infoprint Server daemons immediately or after current activity stops.

You need to edit the AOPSTOP procedure if any of these conditions apply to your installation:

- ▶ The **aopstop** command is not in the default directory **/usr/lpp/Printsrv/bin**. Specify the directory in the PARM parameter on the EXEC statement.
- ▶ If the Infoprint Server configuration file is not in the default location **/etc/Printsrv/aopd.conf**. Specify the location of the file in the AOPCONF environment variable in a data set referred to by the STDENV DD statement.

The AOPSTOP procedure does not use environment variables that are set elsewhere, for example in the **/etc/profile** file.

When the operator invokes this procedure without any options, all active daemons stop when current activity ends. The operator can specify aopstop command options to stop daemons immediately and to stop selected daemons. For example, to stop the Transform Manager and line printer daemons immediately, specify this MVS START command:

```
S AOPSTOP OPTIONS='-d xfd -d lpd now'
```

## **aopstop command**

The **aopstop** command stops all Infoprint Server daemons or only the specified daemons. Unless you specify the **now** or the **force** option, the daemons stop after current activity completes. The command format is:

```
aopstop [-d daemon]... [now] [force]
```

### **Using the -d daemon option**

This option specifies which daemons to stop. If you omit this option, all active daemons are stopped, including the Printer Inventory Manager daemons (aopd, aophinvd, aoplogd, and aopsdbd). You can specify one or more of these values, as follows:

<b>ippd</b>	Stops the Internet Printing Protocol daemon, aopippd.
<b>lpd</b>	Stops the line printer daemon, aoplpd.
<b>netd</b>	Stops the NetSpool daemon, aopnetd.
<b>outd</b>	Stops the IP PrintWay extended mode daemons, aopoutd and aopwsmd. When you start IP PrintWay again, any IP PrintWay printers that were stopped return to the started state, and any printers that were redirected to other printers are no longer redirected.
<b>snmpd</b>	Stops the SNMP subagent daemon, aopsnmpd.
<b>ssid</b>	Stops the Infoprint Central daemon, aopssid.
<b>subd</b>	Stops the Print Interface subsystem daemon, aopsubd.
<b>xfd</b>	Stops the Transform Manager daemon, aopxfd.

### **Using the now option**

This option stops the daemons immediately. If work is in progress, incorrect output or data loss might result. If this option is not specified, the daemons stop after current activity completes.

### **Using the force option**

This option stops the daemons immediately using a “destructive” shutdown. If work is in progress, incorrect output or data loss might result. For daemons that manage a database, such as aopd, aophinvd, or aopsdbd, the database might be corrupted. If this option is not specified, the daemons stop after current activity completes.

### **Command messages**

After you enter the **aopstop** command, you see one or more messages. The messages for each daemon can include any of the following messages:

```
AOP078W Daemon name is not running.  
AOP079I A shutdown of daemon name has been initiated.  
AOP134E The stop of daemon name failed.  
AOP135I An abrupt shutdown of daemon name has been initiated.  
AOP136I A destructive shutdown of daemon name has been initiated.
```

## 3.22 Infoprint Server address spaces

- ❑ Printer Inventory daemons
  - **aopd - aopsdbd - aophindv - aoplogd**
- ❑ IP PrintWay extended mode daemons
  - **aopoutd - aopwsmd**
- ❑ Infoprint Server Transform daemons
  - **aopxpd - afp2psd - pcl2afpd - .....**
- ❑ IPP Server daemon - NetSpool daemon - LPD daemon
  - **aopipppxp                      aopnetd                      aoplpd**
- ❑ Print Interface subsystem daemon - Infoprint Central daemon
  - **aopsubd    aopssid**
- ❑ List Infoprint Server address spaces **MVS D A,AOP\*** operator command output:

<b>AOPD</b>	<b>STEP1</b>	<b>AOPSS</b>	<b>NSW</b>	<b>AO</b>					
<b>AOPLOGD</b>	<b>STEP1</b>	<b>AOPSS</b>	<b>OWT</b>	<b>AO</b>	<b>AOPSDBD</b>	<b>STEP1</b>	<b>AOPSS</b>	<b>OWT</b>	<b>AO</b>
<b>AOPHINDV</b>	<b>STEP1</b>	<b>AOPSS</b>	<b>OWT</b>	<b>AO</b>	<b>AOPXPD</b>	<b>STEP1</b>	<b>AOPSS</b>	<b>OWT</b>	<b>AO</b>
<b>AFP2PSD</b>	<b>STEP1</b>	<b>AOPSS</b>	<b>OWT</b>	<b>AO</b>	<b>PCL2AFPD</b>	<b>STEP1</b>	<b>AOPSS</b>	<b>OWT</b>	<b>AO</b>
<b>AOPLPD</b>	<b>STEP1</b>	<b>AOPSS</b>	<b>OWT</b>	<b>AO</b>	<b>AOPD6</b>	<b>STEP1</b>	<b>AOPSS</b>	<b>OWT</b>	<b>AO</b>
<b>AOPSUBD</b>	<b>STEP1</b>	<b>AOPSS</b>	<b>NSW</b>	<b>AO</b>	<b>AOPNETD</b>	<b>STEP1</b>	<b>AOPSS</b>	<b>OWT</b>	<b>AO</b>
<b>AOPSSID</b>	<b>STEP1</b>	<b>AOPSS</b>	<b>OWT</b>	<b>AO</b>	<b>AOPWSMD</b>	<b>STEP1</b>	<b>AOPSS</b>	<b>OWT</b>	<b>AO</b>
<b>AOPOUTD</b>	<b>STEP1</b>	<b>AOPSS</b>	<b>OWT</b>	<b>AO</b>					

Figure 3-23 Infoprint Server address spaces after start command is issued

### Infoprint Server daemons

The **aopstart** command starts Infoprint Server daemons. If a daemon is already started, aopstart does not start that daemon again.

The **aopstart** command always starts daemons aopd, aopsdbd, aophindv, and aoplogd. The **aopstart** command also starts the optional daemons specified in the start-daemons attribute in the Infoprint Server configuration file.

### Order of starting daemons

Make sure that TCP/IP has finished initialization before you start the LPD (aoplpd), the IPP Server daemon (aopipppd), the IP PrintWay extended mode daemons (aopoutd and aopwsmd), or the z/OS SNMP agent. You do not have to start TCP/IP to start other daemons.

Start the Printer Inventory Manager daemons before you start IP PrintWay basic mode or a NetSpool task. Also, start the Printer Inventory Manager daemons before you start PSF if PSF uses the Printer Inventory.

### Start of Infoprint Server

When you start Infoprint Server from the OMVS shell using the aopstart command, you see the following messages appear on the screen:

```
PAUL @ SC70: />aopstart
AOP075I Daemon aoplogd was started successfully. (program:aopd)
```

AOP075I Daemon aopsdbd was started successfully. (program:aopd)  
AOP075I Daemon aophinvd was started successfully. (program:aopd)  
AOP075I Daemon aopxpd was started successfully. (program:aopd)  
AOP075I Daemon aoplpd was started successfully. (program:aopd)  
AOP075I Daemon aopipdxdp was started successfully. (program:aopd)  
AOPS002I Subsystem established successfully. (program:aopsubd)  
AOP075I Daemon aopsubd was started successfully. (program:aopd)  
AOP075I Daemon aopnetd was started successfully. (program:aopd)  
AOP075I Daemon aopssid was started successfully. (program:aopd)  
AOP075I Daemon aopoutd was started successfully. (program:aopd)  
AOP075I Daemon aopwsmd was started successfully. (program:aopd)  
AOP075I Daemon aopd was started successfully.

### 3.23 Working with Printer Inventory Manager (AOPD)

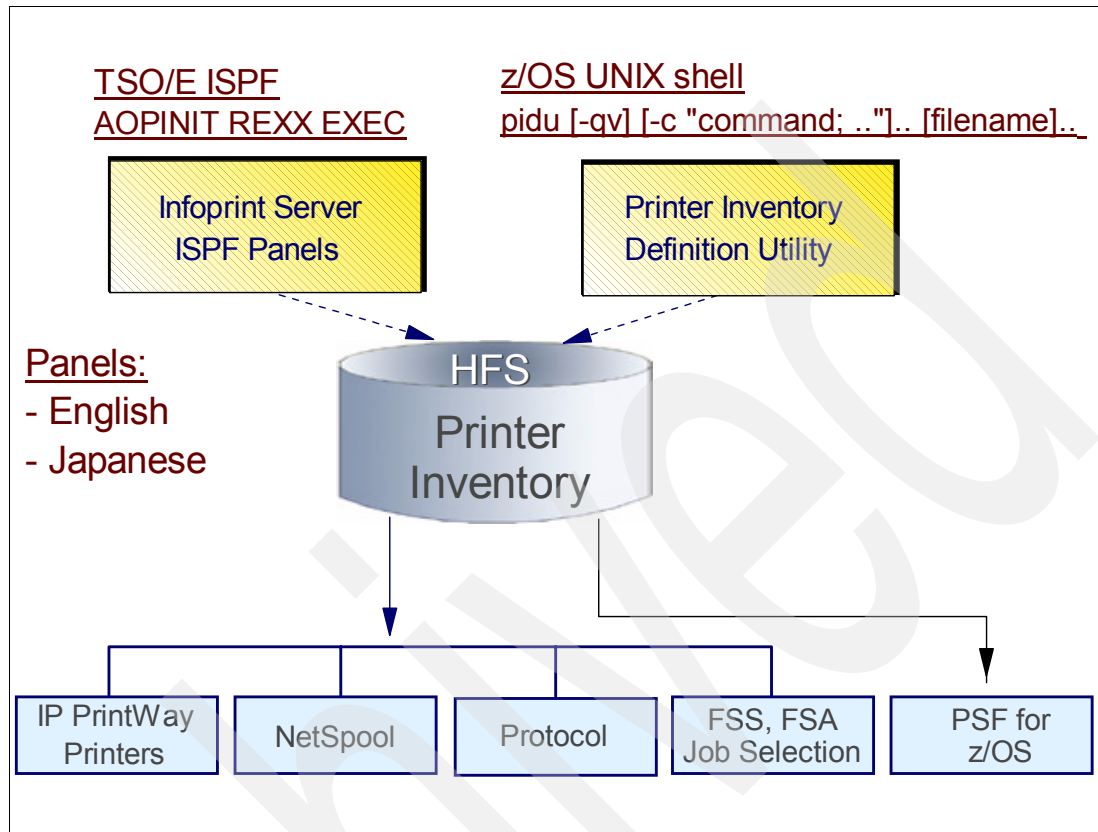


Figure 3-24 Working with Printer Inventory Manager

#### Printer Inventory Manager component

All Infoprint Server components use a common Printer Inventory managed by the Printer Inventory Manager. The Printer Inventory consists of HFS files that contain information about the z/OS print environment.

The Printer Inventory Manager is a key component because in order to use the functions provided by Infoprint Server, you must define the printer environment in the Printer Inventory. Infoprint Server provides the following tools to manage the Printer Inventory:

- ▶ ISPF application to create the printer definitions.
- ▶ The Printer Inventory Definition Utility (pidu) to create and maintain the Printer Inventory. The pidu is useful for creating many printer definitions at the same time and for backing up the Printer Inventory.

Printer definitions are used by all the following Infoprint Server components:

- ▶ IP PrintWay
- ▶ NetSpool
- ▶ Infoprint Central
- ▶ SNMP subagent
- ▶ PSF for z/OS

You must customize the Printer Inventory Manager component to use any function that Infoprint Server provides. You can customize other components later as your installation decides to use additional functions.

## 3.24 Enabling Infoprint Server ISPF panels

- ❑ Used by administrators to:
  - Work with printer definitions and other functions
    - (Add, browse, delete, edit, copy)
- ❑ Enabling the ISPF panels
  - Update TSO logon procedure
    - Update DDs (ISPPLIB and ISPMLIB)
    - AOP.SAOPPENU and AOP.SAOPMENU
    - Update DD SYSPROC or SYSEXEC
    - AOP.SAOPEXEC
  - Customize ISPF primary panel
- ❑ Customize AOPINIT EXEC
  - Provided in AOP.SAOPEXEC(AOPINIT)
    - Default values for configuration panel

Figure 3-25 Customization of ISPF panels

### ISPF panels

The Infoprint Server administrator can use Infoprint Server ISPF panels to add, browse, copy, edit, and delete printer definitions and other objects in the Printer Inventory.

Before using the Infoprint Server Printer Inventory Manager ISPF dialog, review the AOPINIT EXEC provided in AOP.SAOPEXEC(AOPINIT) and modify it, if necessary, for your installation.

Normally, using the instructions in the ISPF Setup sections in the z/OS Program Directory, you need to concatenate the following libraries into the TSO logon procedure:

- AOP.SAOPPENU panel libraries to DD statement ISPPLIB
- AOP.SAOPMENU message libraries to DD statement ISPMLIB

Concatenate library AOP.SAOPEXEC to DD statement SYSPROC or SYSEXEC.

The ISPF z/OS System Programmer Primary Option Menu (member ISR@390S in the ISP.SISPPENU library - Option 8) provides a sample of how to invoke the Printer Inventory Manager ISPF application. You can update your ISPF primary option menu to include a selection option for the Printer Inventory Manager.

ISR@390S panel option 8:

```
&ZSEL = TRANS (TRUNC (&ZCMD, '.'))  
8, 'CMD(AOPINIT &ZTRAIL) NEWAPPL(AOP) SCRNAME(INFOPR) NOCHECK'
```

```
X,EXIT  
  ' ' ' '  
    ,  
  *, '?' )
```

## Modifying the AOPINIT EXEC

The AOPINIT EXEC sets default values for fields that display on the Infoprint Server Configuration panel. You can change any of these values in the AOPINIT EXEC. ISPF users can also change the values for their ISPF sessions on the Configuration panel (select option 7 Configure on the Infoprint Server Printer Inventory Manager panel).

**Attention:** As shipped, the AOPINIT EXEC fails if you change the DOLIBS='N' assignment statement to DOLIBS='Y'. The reason for the failure is the ISPEXEC LIBDEF statements use of wrong library names: AOP.SAOPPLIB and AOP.SAOPMLIB. The correct names are AOP.SAOPPENU and AOP.SAOPMENU.

## 3.25 Inforprint Server configuration panel

Configuration panel showing defaults from AOPINIT

```
AOPIPCF                                     Configuration
Command ==> _____

/ Confirm delete requests
- Trace Printer Inventory ISPF internals
ISPF trace directory. . /var/Printsrv/trace

Printer Inventory:
Configuration file . /etc/Printsrv/aopd.conf
NLS path . . . . . /usr/lpp/Printsrv/%L/%N
Language . . . . . En_US
Default printer. . . lp1

IP PrintWay:
Message log. . . . . ANF.MESSAGE
Trace data set . . . ROGERS.ANFISPF.TRACE
```

Figure 3-26 Inforprint Server configuration panel

### Infoprint Server ISPF configuration panel

The AOPINIT EXEC sets default values for fields that are displayed on the Infoprint Server Configuration panel, Option 7. You can change any of the following values, shown in Figure 3-26, in the AOPINIT EXEC. To modify AOPINIT, follow the instructions in the comments in AOPINIT to make your changes.

The parameters are as follows:

► **Printer Inventory Manager ISPF trace file:**

The name of the file to contain the trace of ISPF processing.

The default file name is **/var/Printsrv/trace**.

► **Configuration file:**

The path name of the Infoprint Server configuration file.

The default file name is **/etc/Printsrv/aopd.conf**.

► **NLS path:**

The directory path that contains the Infoprint Server message catalog. If the LANG environment variable identifies the language in which you want to receive Infoprint Server messages, specify **/usr/lpp/Printsrv/%L/%N**; otherwise, add one of the following values:

- English **/usr/lpp/Printsrv/En\_US/%N**
- Japanese **/usr/lpp/Printsrv/Ja\_JP/%N**



The default is `/usr/lpp/Printsrv/%L/%N`.

► **Language:**

The value of the LANG environment variable.

The default is `En_US`.

► **Default printer:**

The name of the printer definition that the `lpstat` command uses as the default printer. Also, the name of the printer definition that `lp` command uses if the `PRINTER` and `LPDEST` environment variables are not set.

The default is `lp1`.

► **IP PrintWay message log data set:**

The name of the data set where IP PrintWay writes messages.

The default is `ANF.MESSAGE`.

► **IP PrintWay ISPF trace data set:**

The name of the data set to contains the trace of IP PrintWay ISPF processing.

The default is `userid.ANFISPF.TRACE`.

## 3.26 An alternative REXX EXEC - Start ISPF application

```
/* AOPIN -- REXX -- Init AOPINIT Environment */
Trace "O"
/* Setup AOPINIT libs */
Address ISPEXEC
  "LIBDEF ISPMLIB DATASET ID('AOP.SAOPMENU')"
  "LIBDEF ISPPLIB DATASET ID('AOP.SAOPPENU')"
Address "TSO"
  "ALTLIB ACT APPL(EXEC) DS('AOP.SAOPEXEC') "
  /* Suppress IGD104I message for Selection 6 */
  x = OUTTRAP("OF."); "PROF LIST"; x = OUTTRAP("OFF");
  do i = 1 to of.0; if wordpos("WTPMSG",of.i) <> 0 then do;
    "PROF NOWTPMSG"; wtp = 1; leave; end; else wtp = 0;
  end
/* Invoke AOPINIT rexx */
Address ISPEXEC
"CONTROL ERRORS RETURN"
"SELECT CMD(AOPINIT) NEWAPPL(AOP) PASSLIB "
/* Undo AOPINIT environment */
"LIBDEF ISPMLIB "
"LIBDEF ISPPLIB "
Address "TSO"
  if wtp then "PROF WTPMSG"
  "ALTLIB DEACT APPL(EXEC) "
Exit 0
```

TSO command line:  
EX 'TC.EXECS(AOPIN)'.

Figure 3-27 Sample REXX exec to invoke the Printer Inventory Manager ISPF application

### Sample REXX EXEC

Figure 3-27 shows an alternative REXX exec that dynamically activates the Printer Inventory Manager ISPF application environment and invokes the application. You can use the supplied AOPINIT without any changes.

### AOPINIT EXEC

Before using the ISPF panels, review the AOPINIT EXEC provided in AOP.SAOPEXEC(AOPINIT) and modify it, if necessary, for your installation. The AOPINIT EXEC sets default values for fields that display on the Infoprint Server Configuration panel. You can change any of these values in the AOPINIT EXEC. ISPF users can also change the values for their ISPF sessions on the Configuration panel.

## 3.27 Infoprint Server ISPF primary panel

❑ Before using the Infoprint Server ISPF dialog, Infoprint Server must be started.

- The Printer Inventory Manager daemon, **aopd**, must be active.
- Access to the AOP.ADMINISTRATOR resource profile in the PRINTSRV class

```
AOPIPP1                Infoprint Server: Printer Inventory Manager
Option ==> 1_

Printer Definitions
  1 Add                  Add a printer definition
  2 List                 List printer definitions
  3 Select               Select printer definitions to list

Other Functions
  4 Other Definitions    Manage FSS, FSA, pool, and job selection definitions
  5 PrintWay Queue       View IP PrintWay basic mode transmission queue
  6 PrintWay Message     View IP PrintWay basic mode message log
  7 Configure            Change panel configuration
```

Figure 3-28 Infoprint Server ISPF primary panel

### ISPF primary panel

Before using the Infoprint Server ISPF panels, Infoprint Server must be started. The Printer Inventory Manager daemon, **aopd**, must be active.

To use the Infoprint Server panels, you must have READ access to view, and UPDATE access to view and update, to the AOP.ADMINISTRATOR resource profile in the PRINTSRV class, or be a member of the AOPADMIN group.

Before you use the panels for the first time, check the panel configuration. On the Infoprint Server: Printer Inventory Manager panel, select 7 (Configure) and press Enter.

**Note:** Individual ISPF users can also change the values for their ISPF sessions on the Configuration panel itself.

On the Configuration panel, fill in the fields to configure your panels. Your system programmer can change the default values that appear on this panel in file AOPINIT, as described above. Use the ISPF online help for information about each field. To save the new settings and exit the panel, press the END function key.

## 3.28 Add a printer using ISPF panels

```
A0PIPP1                      Infoprint Server: Printer Inventory Manager
Option ==> 1_

Printer Definitions
1 Add          Add a printer definition
2 List        List printer definitions
3 Select      Select printer definitions to list
```

---

```
A0PIPP1                      Infoprint Server: Printer Inventory Manager
0 Choose a Definition Type and Protocol
P A0IPDAP => _

  Type                Protocol
1 IP PrintWay         LPR
2 IP PrintWay         direct sockets
3 IP PrintWay         IPP
4 IP PrintWay         VTAM
5 IP PrintWay         e-mail
6 PSF
7 General

F1=HELP  F2=SPLIT  F3=END  F4=RETURN  F5=RFIND
F6=RCHANGE F7=UP    F8=DOWN F9=SWAP   F10=LEFT
```

Figure 3-29 Add a printer using ISPF panels

### Add a printer using ISPF panels

To add a printer definition:

- ▶ On the Infoprint Server Printer Inventory Manager panel, select 1 Add and press Enter.
- ▶ On the Choose a Definition Type and Protocol panel, select the type of printer definition and press Enter. Types are: IP PrintWay, PSF, and General. If the type is IP PrintWay, also select the type of transmission protocol (LPR, direct sockets, IPP, VTAM, or e-mail). Depending on selected type and protocol, different ISPF panels are displayed.

**Recommendation:** Create one printer definition for each printer so that all job submission methods can be used to print on a printer, including those job submission methods that do not permit the specification of the JES output class, destination name, or IP address. For example, the **lp** command does not let you specify the JES output class or destination name. And, when the IP PrintWay basic mode resubmit for filtering option is used, the printer's IP address cannot be specified on the OUTPUT JCL statement.

This panel lets you select the type of printer definition. If the type is IP PrintWay, you must also select the protocol that you want IP PrintWay to use to transmit data to the remote system. To select a type and protocol, type a selection number in the Option field and press Enter. A description of the A0IPDAP panel, Choose a Definition Type and Protocol follows.

## HELP for the choice of a Definition Type and Protocol panel

Type	Protocol	Meaning
1 IP PrintWay	LPR	A printer to which IP PrintWay transmits data using TCP/IP line printer requester (LPR) to line printer daemon (LPD) protocol.
2 IP PrintWay	direct sockets	A printer to which IP PrintWay transmits data using TCP/IP direct-sockets protocol.
3 IP PrintWay	IPP	A printer to which IP PrintWay transmits data using TCP/IP Internet Printing Protocol.
4 IP PrintWay	VTAM	A printer to which IP PrintWay basic mode transmits data using Virtual Telecommunications Access Method.
5 IP PrintWay	e-mail	A destination to which IP PrintWay transmits data as electronic mail using TCP/IP sendmail function.
6 PSF		An AFP printer controlled by PSF.
7 General		A printer that is not one of the other types.

To leave this panel and return to the previous panel, type **END** or **CANCEL** on the Option line or press the **END** or **CANCEL** function key.

### Types of printer definitions

There are three types of printers as follows:

#### ► IP PrintWay

Select this type for TCP/IP-attached, non-AFP printers, and for VTAM-controlled printers, for IBM AFP printers controlled by Infoprint Manager for AIX or Windows, and for e-mail destinations. Also select this type when you print on a printer that is TCP/IP-attached to a different z/OS system running another instance of Infoprint Server.

When you use the Infoprint Server ISPF panels to select the printer definition type, you also must select the transmission protocol that IP PrintWay is to use to transmit data to the printer. You can select the LPR, direct sockets, Internet Printing Protocol (IPP), VTAM, or e-mail protocol.

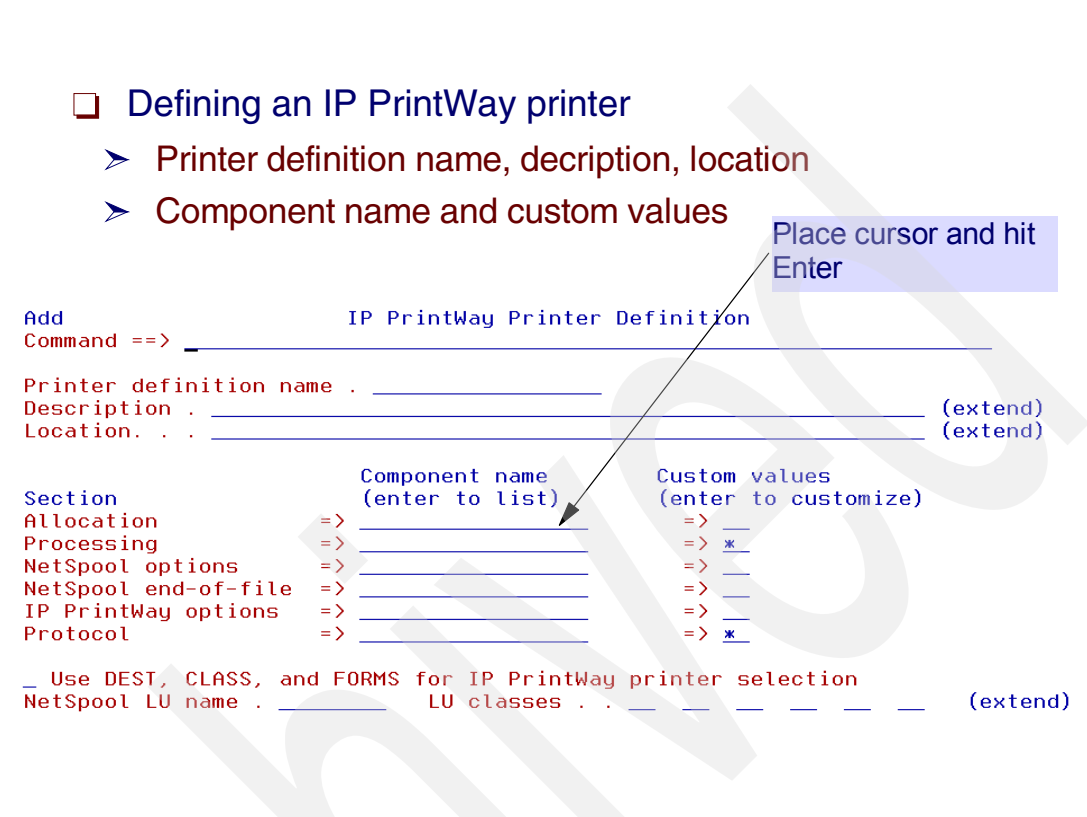
#### ► PSF

Select this type for IBM AFP printers that PSF or a comparable product controls. AFP printers can be locally-attached, SNA-attached, or TCP/IP-attached.

#### ► General

Select this type for printers that do not fall into the other two categories. For example, select General for line printers that JES controls.

## 3.29 Add a printer using ISPF panels



**Defining an IP PrintWay printer**

- Printer definition name, decription, location
- Component name and custom values

Place cursor and hit Enter

Add IP PrintWay Printer Definition

Command ==> \_\_\_\_\_

Printer definition name . \_\_\_\_\_

Description . \_\_\_\_\_ (extend)

Location. . . \_\_\_\_\_ (extend)

Section	Component name (enter to list)	Custom values (enter to customize)
Allocation	=> _____	=> _____
Processing	=> _____	=> *
NetSpool options	=> _____	=> _____
NetSpool end-of-file	=> _____	=> _____
IP PrintWay options	=> _____	=> _____
Protocol	=> _____	=> *

\_ Use DEST, CLASS, and FORMS for IP PrintWay printer selection

NetSpool LU name . \_\_\_\_\_ LU classes . . . \_\_\_\_\_ (extend)

Figure 3-30 Add a printer using ISPF panels

### Add a printer using ISPF panels

On the IP PrintWay Printer Definition panel, the PSF Printer Definition panel, or the General Printer Definition panel:

- ▶ Fill in the printer definition name, description, and location fields on this panel.
  - To display online help information about how to use components and how to specify custom values, place the cursor on the command line and press the HELP function key.
  - To display help information about each field, place the cursor on the input area of the field and press the HELP function key.

### Component name

Creating components is optional. To fill in the fields for each section of the printer definition (for example, the Allocation section), you can do one of these:

- ▶ Select a component that contains the attributes you want. To do this, place the cursor on the Component Name field for the section and press Enter. On the Component List panel, select the component name from the list and press Enter. If a component doesn't exist, create the component on the Component List panel. See "Component name list" on page 122.
- ▶ From this panel, you can also create and manage all of the components for the different sections of the printer definition. To do this, place the cursor on the Component Name field for the section and press Enter.

**Note:** Before you can save a printer definition that includes a component, all included components must exist in the Printer Inventory.

## Custom values

Fill in the fields in each section that is required directly in this printer definition. To do this, place the cursor on the Custom Values field for the section and press Enter. On the next panel, fill in the fields and press the END function key to return to the Printer Definition panel. The fields you fill in are not saved in the inventory until you press the END function key from the Printer Definition panel.

**Note:** An asterisk (\*) is displayed in this field when one or more custom values have been specified in the Processing section.

- ▶ If you want to validate the fields before you save the printer definition, press Enter.
- ▶ To save the new definition but keep the panel on the screen, type SAVE on the command line and press Enter.

**Tip:** Type SAVE on the command line to easily add more than one printer definition of the same type.

- ▶ To save the new definition and exit the panel, press the END function key.

### 3.30 Component name list

## Component Name List

- ❑ After hitting Enter on the Component name field
  - A Component List panel appears as shown below
  - If the component name needed does not exist
    - Place an A-ADD on the command line as shown below
  - If name needed exists - /-Select
    - That component name is then added to the previous panel

Command ==>

a

Component List

Row 1 to 2 of 2  
Scroll ==> HALF

Allocation  
Actions:  
A-Add B-Browse C-Copy D-Delete E-Edit /-Select P-Show use R-Rename  
A Component Name Description  
=====

AL-C-ANYLOCAL  
AL-J-NA-STD  
\*\*\*\*\* Bottom of data \*\*\*\*\*

Figure 3-31 Creating a component name

#### Component names

Components are named and contain attributes (such as Data format) that Print Interface, NetSpool, and IP PrintWay use to process data and validate that a job can print on this printer. Component definitions are not required; you can also use the Custom values field to specify these attributes. If you do not specify a component or custom values, default values are used.

Because a Printer Inventory might contain hundreds or thousands of printer definitions, changing information in many printer definitions can be time-consuming. Therefore, you might want to create additional objects in the Printer Inventory called components and include them in printer definitions. Consider creating components when several printer definitions share the same attributes. To use components:

- Specify the shared attributes in a component, instead of specifying the same attributes in many printer definitions.
- Include the component in all printer definitions to which those attributes apply.

#### Creating components

With the cursor on this field, press Enter to do one of the following:

- Create custom value fields for the new component
- View custom values and values specified in the named component.
- Specify attributes that are not specified in the named component.
- Override attributes that are specified in the named component.



**Note:** When you create components, you do not need to specify every attribute in that component. Instead, you might want to specify some attributes in the printer definitions. For example, the Protocol component contains an attribute that defines the IP address of the remote printer. Because the IP address is usually unique for each remote printer, you could omit the IP address from the component and instead specify it in the printer definition.

When you include components in a printer definition, you can override some of the attributes specified in the components by specifying a different value in the printer definition itself.

## Printer definitions considerations

If you want to minimize the number of printer definitions you create in the Printer Inventory, you can create one printer definition for several actual printers or e-mail destinations in these situations:

- ▶ If job submitters can specify the address of an IP PrintWay printer or the addresses of e-mail recipients on either the OUTPUT JCL statement or in Infoprint Server job attributes, you can create one printer definition for all printers or e-mail destinations that share the same characteristics.
- ▶ If job submitters use the Print Interface subsystem to submit print requests to PSF printers, you can create one printer definition for printers that share the same characteristics. This is because job submitters can print to one printer definition, but direct the output data set to a specific PSF printer by specifying that printer's work-selection criteria (for example, the printer's CLASS and DEST values) on the OUTPUT JCL statement.
- ▶ If your installation has several PSF or JES printers that share the same JES work-selection criteria, you need to create only one printer definition for all the printers. For example, if there are several similar PSF printers in building 8, you could specify the same JES work-selection criterion (for example, destination BLDG8) in the JES FSA definitions for all printers in building 8. This lets JES balance the workload among the printers in building 8. Job submitters can submit all output data sets to the same printer definition and let JES direct the output data set to the printer that becomes available first.

**Note:** IP PrintWay creates automatically a default printer definition DFLTENTRY. You can modify the default printer definition for your installation. However, you must not delete it.

## 3.31 Component definitions

Add Allocation

Command ==> \_\_\_\_\_

Component name. . . . . \_\_\_\_\_

Description . . . . . \_\_\_\_\_

More: +

Spool allocation values:

CLASS . . . . .	_____	LINECT. . . . .	_____
DEST. . . . .	_____	PRMODE. . . . .	_____
JES node. . . . .	_____	PRTY. . . . .	_____
FCB . . . . .	_____	SEGMENT . . . . .	_____
FLASH count . . . . .	_____	THRESHLD. . . . .	_____
FLASH name. . . . .	_____	UCS . . . . .	_____
FORMS . . . . .	_____	WRITER. . . . .	_____
GROUPID . . . . .	_____		
USERDATA . . . . .	_____		

(extend)

BURST . . . . . 1. Yes 2. No

HOLD. . . . . 1. Yes 2. No

OUTDISP . . . . . 1. Purge (without printing) 2. Leave

3. Keep 4. Hold 5. Write

Values for Separator Pages:

Address . . . . . \_\_\_\_\_ (extend)

Building . . . . . \_\_\_\_\_

Department . . . . . \_\_\_\_\_

Name . . . . . \_\_\_\_\_

Room . . . . . \_\_\_\_\_

Title . . . . . \_\_\_\_\_

Figure 3-32 Panel to define an allocation component

### Defining components

The panel shown in Figure 3-32 is the *allocation component* definition panel. You can also create other types of component definitions (one type for each section of a printer definition): *Processing*, *NetSpool Options*, *NetSpool End-of-File*, *IP PrintWay Options*, and *Protocol*.

Notice that the panel is the same as the custom value allocation selection panel except that you are creating a component.

In this allocation component definition, you could specify the work-selection criteria that the system programmer has defined to JES for the IP PrintWay FSA. For example, if the IP PrintWay FSA selects jobs in output class P, specify class P in this component.

You might need to specify some allocation attributes in the printer definitions themselves. For example, if job submitters need to select these printer definitions using the CLASS, DEST, or FORMS parameters on an OUTPUT JCL statement, the value for the DEST or FORMS attribute must be unique for each printer definition. Because this value is unique, do not specify it in the component. Instead, specify the DEST or FORMS value in the printer definition itself.

### Using components

You can include the same components in different types of printer definitions. For example, IP PrintWay and PSF printer definitions can share the NetSpool End-of-File components, if the same NetSpool end-of-file rule is appropriate.

### 3.32 Specifying printer definition attributes

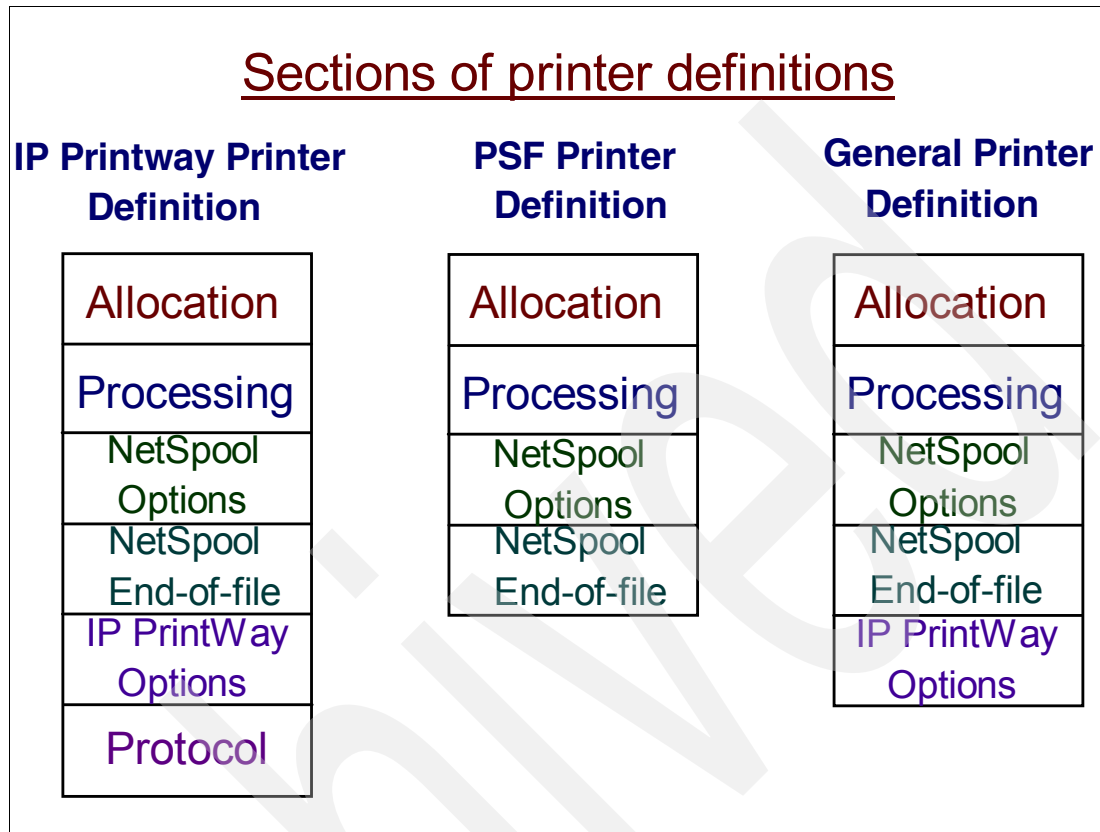


Figure 3-33 Specifying attributes in a printer definition

#### Specifying attributes in a printer definition

The characteristics of a printer and print jobs that you specify in a printer definition are called *printer attributes*. When you use ISPF panels to create or edit printer definitions, you specify printer attributes as values in panel fields. If you use the Printer Inventory Definition Utility to create or edit printer definitions, you specify printer attributes as keyword and value pairs. Although you can specify many attributes in a printer definition, you do not need to specify all of them because Infoprint Server and JES supply default values for many of the attributes.

#### Printer attributes

Some attributes are used by all components of Infoprint Server. However, other attributes are used only by one component of Infoprint Server. Therefore, depending on which components of Infoprint Server you plan to use in your installation, you might not need to specify all of the attributes. For example, if you do not plan to use NetSpool to print VTAM application data, you do not need to specify attributes that only NetSpool uses. The online help for each panel field describes which components use the field.

Most of the attributes that you can specify in a printer definition are divided into logical groups called *sections*. The sections of a printer definition are:

- ▶ Allocation, Processing, NetSpool Options, NetSpool End-of-File, IP PrintWay Options, and Protocol.
- ▶ Figure 3-33 shows the different types of printer definitions and the sections that each type contains.

## Allocation section

The Allocation section contains attributes that tell NetSpool and Print Interface how to allocate output data sets on the JES spool. For example, in this section you can specify the output class, destination name, or job priority.

Each attribute in the Allocation section corresponds to a parameter that you can specify on an OUTPUT JCL statement.

Some print-submission methods let the user specify the same attributes that you can specify in the Allocation section. In this case, the attribute specified during job submission overrides the attribute in the printer definition. For example, if the lp command contains the form-definition attribute, Print Interface allocates the job on the JES spool with the form definition name on the lp command.

## Processing section

The Processing section contains attributes that tell NetSpool, Print Interface, and IP PrintWay how to process data. For example, in this section you can specify attributes that control the page-formatting done by NetSpool and IP PrintWay and attributes that control data transforms done by Print Interface.

The Processing section also contains attributes that Print Interface and IP PrintWay use to determine whether the document can print on the printer. These are called *validation attributes*. If the document cannot print on the printer, Print Interface rejects the print request, and IP PrintWay places the job on the system hold queue. For example, you can specify the types of data formats that the printer supports or the maximum size job that can print.

## NetSpool options section

The NetSpool Options section contains attributes that tell NetSpool how to convert the input data stream before writing the data to the JES spool. NetSpool can convert the input data stream to either a line or PCL data stream. Or, you can request that NetSpool write the input data stream without change to the JES spool.

## NetSpool end-of-file section

The NetSpool End-of-File section contains attributes that tell NetSpool when to close the output data set on the JES spool so that the data can be printed.

## IP PrintWay options section

The IP PrintWay Options section contains attributes that tell IP PrintWay such things as how long to retain data sets on the JES spool after transmission to the remote system, how often to retry unsuccessful transmissions, or which exits to call while processing data.

## Protocol section

The Protocol section contains attributes that tell IP PrintWay which transmission protocol to use to transmit data sets to the remote system: line printer requester (LPR), direct-sockets printing, Internet Printing Protocol (IPP), VTAM, or e-mail. This section also contains attributes that are specific to the type of transmission protocol you select. For example, in this section you can specify LPR options such as whether you want the LPD on the remote system to print a banner page.

The Protocol section also contains the name of the RACF profile that controls who can work with a printer using Infoprint Central for the Web. The RACF profile applies only if you run IP PrintWay extended mode.

### 3.33 Components for IP PrintWay printer definitions

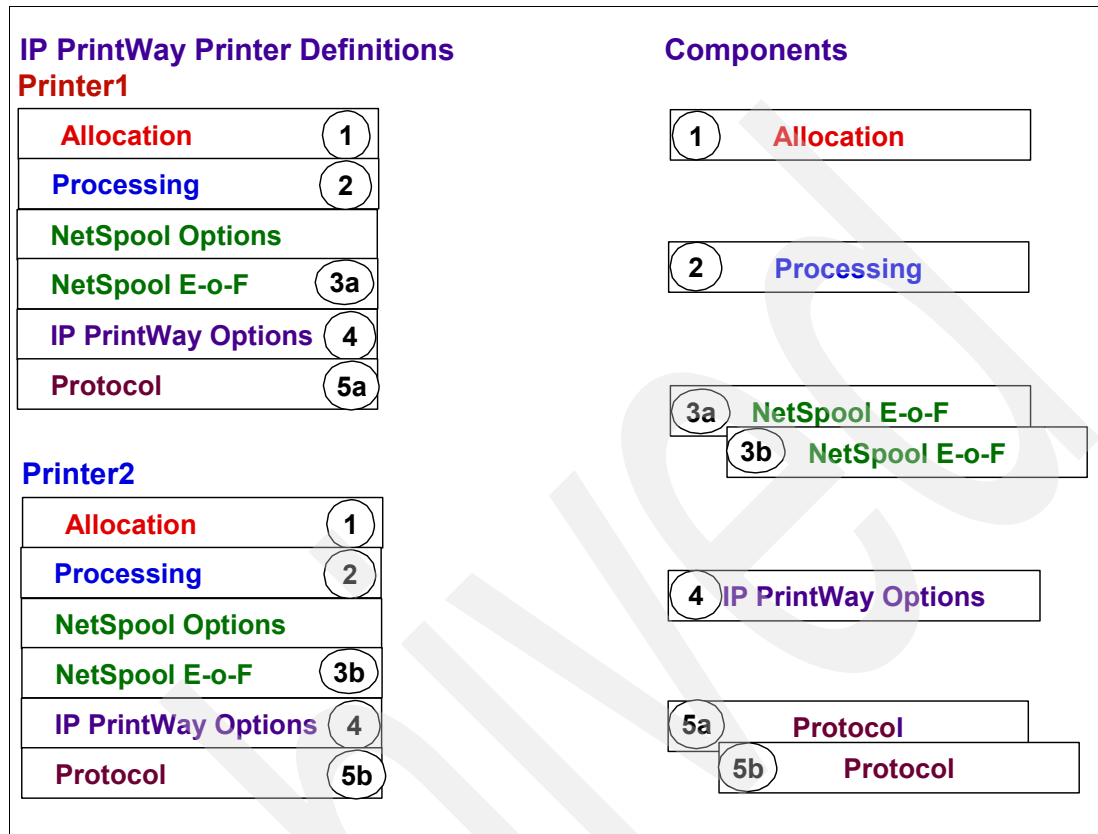


Figure 3-34 Components for IP PrintWay printer definitions

#### Components for IP PrintWay printer definitions

Because a Printer Inventory might contain hundreds or thousands of printer definitions, changing information in many printer definitions can be time-consuming. Therefore, you might want to create additional objects in the Printer Inventory called components and include them in printer definitions. Consider creating components when several printer definitions share the same attributes. To use components:

- Specify the shared attributes in a component, instead of specifying the same attributes in many printer definitions.
- Include the component in all printer definitions to which those attributes apply.

When you change the attributes in a component, all printer definitions that include that component pick up the new attributes.

Creating components is optional. Some printer definitions in the Printer Inventory might include components, while other printer definitions might not. Whether or not you use components, you can use the Printer Inventory Definition Utility (pidu) to make changes to many printer definitions at one time.

When you create components, you do not need to specify every attribute in that component. Instead, you might want to specify some attributes in the printer definitions. For example, the Protocol component contains an attribute that defines the IP address of the remote printer. Because the IP address is usually unique for each remote printer, you could omit the IP address from the component and instead specify it in the printer definition.

When you include components in a printer definition, you can *override* some of the attributes specified in the components by specifying a different value in the printer definition itself. For example, if one printer definition requires a longer retention time on the JES spool, you can override the attribute that specifies the retention time in the printer definition itself. You do not need to create a new component. Notice, however, that if you override an attribute in a printer definition, when you change the same attribute in the component, the printer definition does not pick up the new attribute.

You can create these types of components (one type for each section of a printer definition): Allocation, Processing, NetSpool Options, NetSpool End-of-File, IP PrintWay Options, and Protocol.

With the cursor on a component field, you can do one of the following:

- ▶ Press Enter to see a list of components. From this list you can create, edit, and select a component; then when you return to this panel, the component name is automatically displayed in this field.
- ▶ In Browse Mode, press Enter to view the named component.

### **Custom values**

The custom value fields lets you view the section so that you can specify custom values for this printer definition. This section contains attributes (such as Document format and Maximum document size) that Print Interface, NetSpool, and IP PrintWay use to process data and validate that a job can print on this printer.

Specifying custom values is optional; if you do not specify custom values, specify a Processing component. If you do not specify custom values or a component, default values are used.

### 3.34 Add a printer using ISPF panels - Allocation(1/3)

The screenshot shows the 'Allocation' panel in an ISPF environment. At the top, it says 'AOPIPDAL' and 'Allocation'. Below that is a 'Command ==>' field. The 'Printer definition name' is 'POKEPS'. There is a 'More: +' button. The 'Spool allocation values:' section contains two columns of fields: CLASS, DEST, JES node, FCB, FLASH count, FLASH name, FORMS, GROUPID, USERDATA, LINECT, PRMODE, PRTY, SEGMENT, THRESHLD, UCS, and WRITER. Each field has a line for input. Below this is a section for 'Values for Separator Pages:' with fields for Address, Building, Department, Name, Room, and Title. There are also some status fields like BURST, HOLD, and OUTDISP with numeric options (1, 2, 3, 4, 5). The panel has an '(extend)' button at the bottom right.

Figure 3-35 Add a printer using ISPF panels - Allocation

#### Add a printer using ISPF panels - Allocation (1/3)

The Allocation Panel lets you view and customize the allocation attributes for a printer definition. NetSpool and Print Interface use these attributes (such as CLASS and DEST) to allocate output data sets on the JES spool. If you select Use DEST, CLASS, and FORMS for IP PrintWay printer selection on the IP PrintWay Printer Definition panel, you **must** specify at least one DEST, CLASS, or FORMS value on this panel.

You see this panel when you press Enter from the Custom values field on the Printer Definition panel. The fields on this panel include the printer definition name, JCL keywords and other attributes that specify how a job is sent to the printer, separator page attributes (such as address, title, and name), resource-related attributes, and error-reporting attributes.

On this panel you can:

- Specify values to customize attributes for this printer definition.
- Override values that are displayed from the component by typing other values in the fields.
- View the allocation attributes without making any changes. (This is your only option if you are in Browse Mode.)

The allocation values you specify on this panel apply only to this printer definition; they do not change the attributes specified in the component.

You can customize your ISPF CUA settings so that values that are specified in components are displayed differently from other values. To customize ISPF:

- ▶ On the ISPF Settings panel, select Colors from the action bar and then select CUA attributes.
- ▶ Change an attribute, such as color, for the Choice Entry Field so that it is different from the Normal Entry Field. Values from components are displayed with the CUA attributes for Choice Entry fields.

To **display help for a field** on this panel, place the cursor on the field and press the HELP function key, as shown in the following example.

To save your changes and return to the Printer Definition panel, type END on the Command line or press the END function key. Otherwise, type CANCEL on the Command line or press the CANCEL function key to return to the Printer Inventory Manager panel without saving your changes.

### Spool allocation values heading

The fields under this heading correspond to the OUTPUT JCL parameters that JES can use to direct output data sets from the JES spool to IP PrintWay, a PSF printer, or another JES functional subsystem application (FSA). In these fields, specify the appropriate JES work-selection parameters for the target printer:

- ▶ In an IP PrintWay printer definition, specify the job-selection criteria defined in the IP PrintWay extended mode job selection rule, or the JES work-selection criteria defined for the IP PrintWay basic mode FSA. For example, if the job-selection criterion is class P, specify P in the CLASS field.
- ▶ In a PSF printer definition, specify the JES work-selection criteria for the PSF printer FSA.

### Values for separator page

Address, Building, Department, Name, Room, Title: Installation-written IP PrintWay exits can read the values specified in these fields. For example, your installation can write an IP PrintWay Begin Data Set exit to print these values on a separator page.

Title: The value in this field is also used as (1) the subject line of the e-mail when you select the e-mail protocol and (2) the title passed to the LPD when you select the LPR protocol. You can specify up to 60 characters, including blanks and any other printable characters. A title specified by the job submitter overrides this value.

### Default values for separator page

For a separator page created by an IP PrintWay or PSF exit: The default value is the title your administrator specified in the Allocation section of the printer definition. If none is specified, there is no default value.

For the LPD's separator page: The default value is the title your administrator specified in the Allocation section of the printer definition. If none is specified, the default value is the title or filename your administrator specified in the Protocol section of the printer definition. If none is specified, the default value is the data set name.

For the subject of an e-mail: The default value is the title specified in the Allocation section of the printer definition. If none is specified, the default value is the job name. Depending on how the print request was submitted, the job name might be the ID of the user who submitted the print request.



### 3.35 Add a printer using ISPF panels - Allocation(2/3)

```
AOPIPDAL                                     Allocation
Command ==> _____

Printer definition name . POKEPS _____

Resource Related Values:
Form definition . _____
Character sets . _____
Overlay front . . _____ Back . . _____
Input tray . . . _____
Output bin . . . _____
Page definition . _____
Resource library. _____ (extend)
Resource directories. _____ (extend)
Image shift x-direction front . . _____ Back . . _____
                      y-direction front . . _____ Back . . _____

Error Reporting Values:
Print error reporting. . . 1. None 2. All 3. Character 4. Position
Error disposition. . . . 1. Default 2. Hold 3. Quit
Print error messages . . 1. No 2. Yes
Maximum messages. . . . _____

Other Values:
Notify . . . . . _____ at node . . _____
                      _____ at node . . _____
                      _____ at node . . _____
                      _____ at node . . _____

Checkpoint pages . . . _____
Checkpoint seconds . _____
Copies . . . . . _____
```

Figure 3-36 Add a printer using ISPF panels - Allocation

#### Add a printer using ISPF panels - Allocation (2/3)

Figure 3-36 shows the next part of the allocation panel.

##### Resource related values

On the Allocation panel, you can specify these fields, which IP PrintWay transmits to Infoprint Manager if Print Interface or NetSpool allocate the data set on the JES spool.

##### Error reporting values

The attributes indicate whether the printer reports character and position errors. Character errors are caused by trying to use a code point that is not assigned to a character in a font. Position errors are caused by trying to print outside the printable area.

##### Other values

- Notify (basic mode): You can specify up to four user IDs that IP PrintWay basic mode notifies when a data set has been successfully or unsuccessfully transmitted, or when IP PrintWay deletes the data set from the JES spool. IP PrintWay extended mode cannot notify users.
- Copies: Specify the number of copies to be printed. Allowed values are 1 to 32640. The default value is 1.

- Specify this field only when you select the IP PrintWay LPR, direct sockets, and IPP transmission protocols. For other protocols, IP PrintWay ignores the number of copies requested and prints only one copy.
- Some IPP printers do not support the copies IPP job attribute. In this case, only one copy prints.
- If you select the LPR protocol and only one copy prints, deselect the Optimize copies field.
- If each copy does not start on a separate page, you might be able to specify printer instructions to force each copy to start on a new page.

### 3.36 Add a printer using ISPF panels - Allocation(3/3)

```
Copy group . . . . . _ _ _ _ _  
Color map. . . . . _  
Com setup member . . _  
JES form length. . . _  
Resolution . . . . . _  
AFP parameters . . . _  
Duplex . . . . . _ 1. Simplex 2. Duplex 3. Tumble  
Label data pages . . _ 1. Yes 2. No  
Restrict printable area . . _ 1. Yes 2. No  
_ Table reference characters  
_ Save AFP statistics
```

Figure 3-37 Add a printer using ISPF panels - Allocation

#### Add a printer using ISPF panels - Allocation (3/3)

This is the last part of the allocation panel, which includes the last part of the Other Values section, as follows:

- ▶ Duplex: Specify whether to print on one or both sides of the sheet.
  - The field applies only when IP PrintWay uses the IPP protocol to transmit data to the printer. If the IPP server for the printer does not support the sides IPP job attribute, the printer's default is used.

### 3.37 Add a printer using ISPF panels - Processing (1/3)

```
AOPIPD RR                                     Processing
Command ==> _____

Printer definition name . POKEPS _____

More: +

Document code page . . . _____
Printer code page. . . IS08859-1 _____

Supported Data Formats and Associated Filters:
Data format:  Filter:
/ Line data      _____ (extend)
/ MO:DCA-P      _____ (extend)
/ PostScript    _____ (extend)
/ Text          aopfiltr.so _____ (extend)
/ PCL           _____ (extend)
/ PDF           _____ (extend)
/ SAP           _____ (extend)
/ XML           _____ (extend)
/ TIFF          _____ (extend)
/ JPEG          _____ (extend)
/ Other         _____ (extend)

_ Resubmit for filtering

AFP to PDF Transform Encryption:
User identifier . _____ (extend)
Owner identifier. _____ (extend)
Encryption level 2 1. Low (40-bit key) 2. High (128-bit key)
```

Figure 3-38 Add a printer using ISPF panels - Processing

#### Add a printer using ISPF panels - Processing

The Processing section contains attributes that tell NetSpool, Print Interface, and IP PrintWay how to process data. For example, in this section you can specify attributes that control the page-formatting done by NetSpool and IP PrintWay and attributes that control data transforms done by Print Interface.

The Processing section also contains attributes that Print Interface and IP PrintWay use to determine whether the document can print on the printer. These are called validation attributes. If the document cannot print on the printer, Print Interface rejects the print request, and IP PrintWay places the job on the system hold queue. For example, you can specify the types of data formats that the printer supports or the maximum size job that can print.

The Processing Panel lets you view and customize processing and validation attributes for a printer definition. Print Interface, NetSpool, and IP PrintWay use these attributes to process data and validate that a print job can print on this printer. You see this panel when you press Enter from the Custom Values field on the Printer Definition panel.

On this panel you can:

- Specify values to customize attributes for this printer definition.
- Override values that are displayed from the component by typing other values in the fields.
- View the attributes without making any changes. (This is your only option if you are in Browse Mode.)

The values you specify on this panel apply only to this printer definition; they do not change the attributes specified in the component.

To display help for a field on this panel, place the cursor on the field and press the HELP function key, as shown in the following example.

To save your changes and return to the Printer Definition panel, type END on the Command line or press the END function key. Otherwise, type C CEL on the Command line or press the CANCEL function key to return to the Printer Inventory Manager panel without saving your changes.

### Document code page

The code page used to create documents. Print Interface, NetSpool, and IP PrintWay use this code page as the source when converting documents between ASCII and EBCDIC. This field is not required.

### Printer code page

The code page that the printer uses. Print Interface, NetSpool, and IP PrintWay use this code page as the target when converting data between ASCII and EBCDIC. This field is not required. However, if it is blank, some documents might print incorrectly.

### Supported Data Formats and Associated Filters

An indicator of which **data formats** are accepted as input. Select all data formats that the printer supports and also all data formats that can be transformed into a supported format using an associated filter.

#### Data formats:

Line data	Character data; can have carriage controls and TRCs
MO:DCA-P	IBM Mixed Object Document Content Architecture Presentation
PostScript	Adobe PostScript
Text	Character data; can have LF, CR, HT, VT, FF controls
PCL	Hewlett Packard Printer Control Language
PDF	Adobe Portable Document Format
SAP	SAP output text format (OTF) or ABAP format
XML	Extensible Markup Language format
Other	All data formats that are not one of the types above, including the Xerox data format

**Note:** The job is rejected if the data format is not selected. With transforms configured, PSF can support all data formats. Without transforms configured, PSF can support Line data, MO:DCA-P, Text, and XML.

### Filter Field

The filter DLL or UNIX filter used for documents that contain line data. A filter can transform data from one format to another. If this field is blank, no filter is used. Infoprint Server Transforms provides filters for printing line data formatted with an AFP page definition on an ASCII printer.

### 3.38 Add a printer using ISPF panels - Processing(2/3)

```
AOPIPDORR                                     Processing
Command ==> _____

Printer definition name . POKEPS _____

Encryption level  2  1. Low (40-bit key) 2. High (128-bit key)  More:  - +
Protected actions:
  Restrict print. . . 2  1. Yes  2. No
  Restrict copy . . . 2  1. Yes  2. No
  Restrict update . . 2  1. Yes  2. No

SCS Conversion:
Margins:  Top . . . . . Bottom . . . . . Left . . . . . Right . . . . .
Line length . . . . . Page length . . . . .
Tabs:    Vertical . . . . . (extend)
         Horizontal . . . . . (extend)

NetSpool PCL Conversion:
Print density . . . . .
Line density . . . . .
Orientation . . . . . 1  1. None  2. Portrait  3. Landscape
_ SCS automatic page orientation

IP PrintWay Line-to-Text Conversion:
/ Pagination
  Margins: Top . . . . . Bottom . . . . .
  Page height . . . . . 58
/ Print page header
```

Figure 3-39 Add a printer using ISPF panels - Processing(2)

#### SCS conversion

Specifies the default page-formatting values, which NetSpool uses for input SCS data streams only.

#### NetSpool PCL conversion

SCS data streams: NetSpool uses SCS Set Horizontal Format (SHF) and SCS Set Vertical Format (SVF) controls to format data into lines and pages. The SHF and SVF controls specify page-formatting values such as line length, page length, margins, and tabs. The SHF and SVF controls take effect immediately and remain in effect until either the next SHF or SVF control or until NetSpool starts writing a new output data set to the JES spool

If the SCS data stream does not contain SHF and SVF controls, NetSpool uses default values. In each printer definition, you can specify the default values that NetSpool is to use for the line length, page length, margins, and tabs.

3270 data streams: NetSpool uses page-formatting values in the 3270 Write Control Characters (WCCs) in the 3270 data stream to format data into lines and pages. To change page-formatting values for 3270 data, the application programmer must change the WCCs generated by the VTAM application that creates the 3270 data. For 3270 data streams, NetSpool does not use any of the default page-formatting values for line length, page length, margins, and tabs that you specify in the printer definition.

### **IP PrintWay Line-to-Text conversion**

An indicator of whether IP PrintWay is to format line data without carriage control characters into pages. When formatting data, IP PrintWay can add a header to each page and leave blank lines for top and bottom margins. If you select this option, you can also specify these fields: Margins: Top and Bottom, Page height, and Print page header.

### 3.39 Add a printer using ISPF panels - Processing(3/3)

```
AOPIPDORR                                     Processing
Command ==> _____

Printer definition name . POKEPS

Page height . . . . 58
/ Print page header

Maximum document size . _____
Maximum copies. . . . . _____
Forms supported . . . . . _____ (more)
Duplex supported. . . . / Simplex / Duplex / Tumble
Print-error reporting supported . / Character / Position

Input tray name:   Number: (more)   Output bin name:   Number: (more)
_____
_____

IP PrintWay Basic Mode Attributes:
SOSI mode . . . . . _ 1. None 2. ASCII 3. Space 4. EBCDIC
Translation dataset qualifier . _____
Double-byte translate table . . . _
  1. BIG5          2. EUCKANJI      3. HANGEUL
  4. IBMKANJI      5. JIS78KJ-ASCII  6. JIS78KJ-JISROMAN
  7. JIS83KJ-ASCII  8. JIS83KJ-JISROMAN  9. KSC5601
 10. SCHINESE      11. SJISKANJI     12. TCHINESE
```

Figure 3-40 Add a printer using ISPF panels - Processing(3)

#### Input tray and output bin

Identification of the paper source used to print a data set and output bin a printer supports.

#### IP PrintWay basic mode attributes

The shift-out shift-in (SOSI) mode that IP PrintWay basic mode uses to delimit double-byte character set (DBCS) data in the ASCII data stream.

The name of the TCP/IP translation table that IP PrintWay basic mode uses to convert single-byte character set (SBCS) data between EBCDIC and ASCII. This translation table name is used with both single-byte data and double-byte data.

The name of the translation table IP PrintWay basic mode uses to convert double-byte character set (DBCS) data from EBCDIC to ASCII.



## 3.40 Add a printer using ISPF panels - IP PrintWay Options (1/2)

```
AOPIPDI0                                IP PrintWay Options
Command ==> _____

Printer definition name . POKEPS _____

Retention period:
  Successful. . . . . _____ Failure . . . . . _____
Retry time . . . . . _____
Retry limit. . . . . _____

Connection timeout . 30
Response timeout . . 600

Exits:
  Begin data set. . . . . _____ End data set. . . . . _____ Record. . . . . _____

Document header . . . . . _____ (extend)
  / Translate document header
Document trailer . . . . . _____ (extend)
  / Translate document trailer

Formatting:
  Transparent data char . 35
  Delete form feed. . . . 1 1. None 2. Leading 3. Trailing 4. Both
  Carriage control type . _ 1. None 2. Machine 3. ANSI
  _ Automatic dataset grouping (extended mode)
  Dataset grouping. . . . 2 1. None 2. Job 3. Concatenate job
  _ Omit line termination at EOF
```

Figure 3-41 Add a printer using ISPF panels - IP PrintWay Options

### Add a printer using ISPF panels - IP PrintWay Options

The IP PrintWay Options Panel lets you view and customize attributes in the IP PrintWay Options section for this printer definition. These attributes define how IP PrintWay is to process data prior to transmission to the remote system, how long to wait for a response from the remote system, and how long to retain data sets on the JES spool after transmission. These attributes apply to all of the transmission protocols that IP PrintWay supports (lpr, IPP, direct sockets).

On this panel you can:

- Specify values to customize the attributes for the printer definition.
- Override values that are displayed from the component by typing other values in the fields.
- View the attributes without making any changes.

Values that you specify on this panel apply only to this printer definition; they do not change the values specified in the component.

To display help for a field on this panel, place the cursor on the field and press the HELP function key.

To save your changes and return to the Printer Definition panel, type END on the Command line or press the END function key. Otherwise, type CANCEL on the Command line or press the CANCEL function key to return to the Printer Inventory Manager panel without saving your changes.

## Retention period

IP PrintWay can retain data sets on the JES spool after successful transmission or after completing all transmission attempts. Although retaining data sets on the JES spool uses spool space, it lets you retransmit data sets that have not printed correctly or have not been successfully sent to the e-mail destination.

If IP PrintWay retains a data set on the JES spool, you can either (1) correct the problem that caused the transmission to fail and retransmit the data set to the same printer or e-mail destination or (2) transmit the data set to a different printer or e-mail destination. For information about how to retransmit data sets that IP PrintWay retains on the JES spool.

## Exits

The IP PrintWay Begin Data Set exit routine is called before processing any records in a data set. In this exit you can add one or more records to the beginning of the data set and you can change IP PrintWay options. This field is not required.

The IP PrintWay End Data Set exit routine is called after processing all records in a data set. In this exit you can add one or more records to the end of the data set. You can also inspect IP PrintWay options, but you cannot change them. This field is not required.

The document header field specifies additional data IP PrintWay is to add to the beginning of each document. For example, you could specify printer commands to change the printer default font. You can enter data in EBCDIC or ASCII.

The document trailer field specifies additional data IP PrintWay is to add to the end of each document. For example, you could specify printer commands to restore the default printer configuration. You can enter data in EBCDIC or ASCII.

## 3.41 Add a printer using ISPF panels - IP PrintWay Options(2/2)

```
A0PIPDIO                                IP PrintWay Options
Command ==> _____

Printer definition name . POKEPS_____

More: -

Document header . . . _____ (extend)
  / Translate document header
Document trailer . . . _____ (extend)
  / Translate document trailer

Formatting:
Transparent data char . 35
Delete form feed. . . . 1 1. None 2. Leading 3. Trailing 4. Both
Carriage control type . - 1. None 2. Machine 3. ANSI
  - Automatic dataset grouping (extended mode)
Dataset grouping. . . . 2 1. None 2. Job 3. Concatenate job
  - Omit line termination at EOF

Basic Mode Formatting:
Line termination. . . . _____
Formatting. . . . . - 1. None 2. Standard
                      3. Translate only 4. Use FCB
PostScript header . . . - 1. Add 2. Ignore
                      3. Landscape 4. Always landscape
```

Figure 3-42 Add a printer using ISPF panels - IP PrintWay Options (2)

### Basic Mode Formatting

The line termination field specifies the end-of-line controls required by the target system. IP PrintWay basic mode translates the controls to ASCII, if necessary, and adds them to the end of each line. This field is not required. If it is blank, PrintWay uses one of these end-of-line controls, depending on the protocol:

- ▶ LPR, direct sockets, IPP, e-mail -- 25 line-feed
- ▶ VTAM -- 0D15 carriage-return and new-line

Formatting specifies the type of translation and formatting for IP PrintWay basic mode.

1. **None:** Does not translate or format data. Select to print binary data (AFP or PCL).
2. **Standard:** Translates data to ASCII or EBCDIC, adds line termination controls, and formats data into pages. For the VTAM protocol, converts line data to SCS or DSC/DSE format. (default)
3. **Translate only:** Translates data to ASCII or EBCDIC and adds line termination controls. However, it does not format data into pages. For the VTAM protocol, this option is the same as Standard.
4. **Use FCB:** Same as Standard, except that if an FCB is specified for the data set, formats using the FCB. For these conditions, PrintWay uses "None" and ignores any other values:
  - a. Send as transparent data field is selected on the VTAM Protocol panel.
  - b. Remote PSF option is selected on the LPR Protocol panel.
  - c. NetSpool converted the data to PCL format.
  - d. For protocols other than VTAM, Print Interface processed the data set.

## 3.42 Add a printer using ISPF panels - LPR Protocol

```
AOPIPDIL                                LPR Protocol
Command ==> _____

Printer definition name . POKEPS _____
Operator security profile _____
. . . . .

Printer IP address . _____ (extend)
Print queue name . . . _____ (extend)

LPR Processing Options:
Mode . . . . . 2  1. Control file first  2. Control file last
                  3. Stream              4. Remote PSF

_ Optimize copies
_ Restrict ports
/ Print banner page
  Banner class . . _____
  Banner job name _____ (extend)
Filename . . . . . _____
Indent . . . . . _____
Owner . . . . . _____
Print function . . . f _____ (extend)
Title . . . . . _____ (extend)
Width . . . . . _____
User options . . . . _____ (extend)
```

Figure 3-43 Add a printer using ISPF panels - LPR Protocol

### Add a printer using ISPF panels - LPR Protocol

The LPR Protocol Panel lets you view and customize LPR protocol attributes for this printer definition. IP PrintWay uses these attributes to determine how to transmit data sets from the JES spool to the remote printer or print server.

On this panel you can:

- ▶ Specify values to customize the attributes for the printer definition.
- ▶ Override values that are displayed from the component by typing other values in the fields.

Consider specifying custom values for the following fields, instead of specifying the values in a component, because these fields identify the target printer and are usually unique for each printer definition:

- ▶ Printer IP address
- ▶ Print queue name

The values you specify on this panel apply only to this printer definition; they do not change the values specified in the component.

To change the protocol used by IP PrintWay, use the X action on the Printer Definition List panel. (Use the List or Select function on the main Infoprint Server panel to display the Printer Definition List panel.)

To save the definition in the Printer Inventory and remain on this panel, type **SAVE** on the Command line.

**Hint:** If you are in Add or Copy mode, you can repeat these steps to add multiple definitions with different target addresses: (1) Change the Printer definition name field. (2) Change the address of the target printer. (3) Type **SAVE** on the Command line.

Use **END** to return to the Printer Definition panel, retaining changes made on this panel. Use **CANCEL** to get out without saving changes.

### Printer IP address

The IP address of the remote printer or print server. You can specify either the host name (also called domain name) or the dotted-decimal IP address. If the IP address is in colon hexadecimal format, you must specify the host name. This field is required when you select either the LPR or direct sockets protocol type.

**Note:** Print Interface, IP PrintWay extended mode, and the SNMP subagent support IPv6. If you enable or disable IPv6 while Infoprint Server daemons are running, you must stop and restart the LPD (aoplpd), the IPP Server daemon (aopipdpx), the IP PrintWay extended mode daemons (aopoutd and aopwsmd), and the SNMP subagent daemon (aopsnmpd).

If a printer's IP address is in the IPv6 colon-hexadecimal format, define a host name for the IPv6 address in the DNS. This is required because administrators cannot specify IP addresses in the colon-hexadecimal format in printer definitions.

You must use the host name (instead of the colon-hexadecimal address) in the **DEST=IP: JCL** parameter, in Infoprint Server job attributes, and in printer definitions.

### Print queue name

The name of the print queue in the remote printer or print server. This field is required when you select the LPR protocol type.

### 3.43 Printer Inventory Definition Utility (pidu)

- ❑ Program that manages objects in Printer Inventory
  - Can be used in place of ISPF panels
- ❑ **pidu** command can be used in 2 ways
  - From the UNIX shell
    - pidu [-qv] [-c "command ... ;"] [filename]
    - pidu -c "display printer pokesps ; "
  - From a batch job
    - ```
//AOPPIDU JOB , 'pidu'
//PIDU EXEC PGM=AOPBATCH,
// PARM='pidu -c "display printer pokesps;" '
//*STDIN DD DSN=hlq.INVDEFS, DISP=SHR
//STDOUT DD SYSOUT=*
//STDERR DD SYSOUT=*
```

Figure 3-44 Using the printer inventory definition utility

#### Printer Inventory definition utility

The Printer Inventory Definition Utility (pidu) program can be used to manage objects in the Printer Inventory. Inventory objects include printer definitions, printer pool definitions, components, FSS definitions, FSA definitions, and job selection rules.

The pidu program is useful for creating and editing many objects at the same time. Also, it lets you do some functions that you cannot do with Infoprint Server ISPF panels. For example, you can export or dump objects in the Printer Inventory to a file, and you can do more powerful searches of the Printer Inventory.

Before the pidu program can run, the Printer Inventory Manager daemon aopd must be active.

In order to use the pidu program you must have a UID of 0 or be a member of the AOPADMIN group *and* have access to the AOP.ADMINISTRATOR resource profile in the RACF PRINTSRV class:

- ▶ READ access is required to display, dump, export, or list objects in the Printer Inventory.
- ▶ UPDATE access is required to create, delete, force-create, modify, or rename objects in the Printer Inventory.

If the AOP.ADMINISTRATOR profile does not exist in the PRINTSRV class, READ access to the AOPADMIN profile in the FACILITY class is required to do any functions.

## Executing the pidu program

You can run the pidu program in two ways:

- ▶ From the z/OS UNIX shell using the **pidu** command.
- ▶ As a batch job.

You can run the pidu program as a batch job by using either the AOPBATCH program or the BPXBATCH utility program. IBM recommends using the AOPBATCH program.

You can invoke AOPBATCH in JCL with this EXEC statement:

```
//stepname EXEC PGM=AOPBATCH,PARM='pidu [-v] [-q]'
```

- q Suppresses informational messages that the **pidu** command writes to the output data set named in the STDOUT DD statement.
- v Writes the name of the Printer Inventory to the output data set named in the STDERR DD statement. Also provides additional informational messages.

If you use the BPXBATCH utility program to run the pidu command, you must always set the PATH, LIBPATH, and NLSPATH environment variables, even if your installation installed Infoprint Server files in default locations. Specify the AOPCONF environment variable if the Infoprint Server configuration file is not in /etc/Printsrv/aopd.conf or in \$HOME/.aopconf.

For information about the BPXBATCH utility program, see *z/OS UNIX System Services Command Reference*, SA22-7802.

## 3.44 pidu commands

### ❑ pidu commands

- `pidu [-qv] [-c "command; ..."]... [filename]...`
  - **create** -- Create a new object
  - **delete** -- Delete an object
  - **display** -- Display the attributes of an object
  - **dump** -- Dump all objects
  - **export** -- Generate create commands for all objects or only objects that meet certain criteria
  - **force-create** -- Create an object and replace an object of the same name and in the same object class if it exists
  - **list** -- List all objects or only objects that meet certain criteria
  - **modify** -- Modify attributes of an object
  - **rename** -- Rename an object

Figure 3-45 pidu commands

### Running the pidu commands

Following is the command format:

```
pidu [-qv] [-c "command; ..."]... [filename]...
```

Where the options are as follows:

`-c 'command; ...'`

`-c "command; ..."`

Specifies one or more pidu commands. Enclose the commands in single or double quotation marks, and end each statement with a semicolon. If a command contains a value that requires single quotation marks, such as a hexadecimal value, enclose the commands in double quotation marks. You can repeat the `-c` option.

If you do not specify the `-c` option or the name of a file, pidu reads the commands from standard input (stdin), which can be either keyboard data or output from another command.

`-q` Suppresses informational messages that the **pidu** command writes to stdout.

`-v` Writes the name of the Printer Inventory to stderr. Also provides additional informational messages.

`filename` The name of a UNIX file or sequential MVS data set that contains the commands. You can repeat this option. If the data set is an MVS data set, specify `//` before the filename. If you specify a fully qualified data set name,



also enclose the data set name in single quotation marks and code a backslash before each single quotation mark.

For example, if the output file is named USERID.MYFILE, enter:  
`//\ 'USERID.MYFILE\ '`

If you want your TSO user ID prefixed to the data set name, specify:  
`//MYFILE`

To specify commands from stdin, omit the file name and the `-c` option.

## **pidu commands**

Following are the pidu commands:

|                           |                                                                                                    |
|---------------------------|----------------------------------------------------------------------------------------------------|
| <code>create</code>       | Create a new object.                                                                               |
| <code>delete</code>       | Delete an object.                                                                                  |
| <code>display</code>      | Display the attributes of an e object.                                                             |
| <code>dump</code>         | Dump all objects.                                                                                  |
| <code>export</code>       | Generate create commands for all objects or only objects that meet certain criteria.               |
| <code>force-create</code> | Create an object and replace an object of the same name and in the same object class if it exists. |
| <code>list</code>         | List all objects or only objects that meet certain criteria.                                       |
| <code>modify</code>       | Modify attributes of an object.                                                                    |
| <code>rename</code>       | Rename an object.                                                                                  |

You can specify pidu commands in the `-c` option or in a file. The pidu command writes a report of errors to standard error (STDERR) and writes informational messages and command output to standard output (STDOUT).

### 3.45 pidu -c "display printer pokeps ; "

```
# display printer pokeps
printer-codepage = ISO8859-1
filters = { text -> aopfiltr.so }
location = 2c-16
description = "3130 Postscript"
printer-type = ip-printway
retry-limit = 1
retry-time = 0000:00:10
failure-retention-period = 0000:01:00
successful-retention-period = 0000:00:00
print-queue-name = afccu2
printer-ip-address = 9.12.2.4
normal-output-disposition = purge
hold = no

forms = STD
output-class = J
burster-trimmer-stacker = no
destination = POKEPS
document-formats-supported = {
  line
  postscript
  text
  pcl
}
lpr-print-banner = no
printway-formatting = none
dcf-routing = yes
duplex = yes
```

Figure 3-46 Example of a pidu command output

#### Example of a pidu command output

Figure 3-46 shows a **pidu** command **display printer pokeps** output.

#### Using AOPBATCH

You can invoke AOPBATCH in JCL with this EXEC statement:

```
//stepname EXEC PGM=AOPBATCH,PARM='pidu [-v] [-q]'
```

IBM provides sample JCL in the AOPPIDU member of SYS1.SAMPLIB for running the pidu program using the AOPBATCH utility, as follows:

```
//AOPPIDU JOB , 'pidu'
//PIDU EXEC PGM=AOPBATCH,PARM='pidu'
//STDIN DD DSN=h1q.INVDEFS,DISP=SHR
//STDOUT DD SYSOUT=*
//STDERR DD SYSOUT=*
/* STDENV may point to a dataset containing environment variables.
/* Builtin values will work for the default installation.
/*STDENV DD DSN=environment,DISP=SHR
```

STDIN DD-statement specifies the pidu input source file or data set when the **-c** option is not used.

## 3.46 Backing up and restoring Printer Inventory

- ❑ Dump the contents to a file
  - HFS file or an MVS data set
    - `pidu -c "dump /\ 'rogers.pidu\ ' ; "`
    - (Does not create comment statements in output)
- ❑ Export objects to a file
  - HFS file or an MVS data set
    - `pidu -c "export /\ 'invent.export\ ' ; "`
    - (Create comment statements in output)
- ❑ Restore (create) inventory entries
  - `pidu /\ 'invent.export\ '`

Figure 3-47 Backing up and restoring Printer Inventory

### Backing up and restoring Printer Inventory

You should back up the Printer Inventory on a regular basis. You can use the Infoprint Server Printer Inventory Definition Utility (`pidu`) **export** command to back up the Printer Inventory to a file.

When you restore the Printer Inventory, no print jobs are lost. However, information in the Printer Inventory about jobs that were submitted through Print Interface is lost. This means that users, including SAP R/3 users, are not notified when these jobs are completed, and users cannot query the status of these jobs.

You can back up the Printer Inventory to either an HFS file, data set, or a data set in a generation data group (GDG).

#### Rules:

- ▶ Do not use copy commands to create a backup copy of the Printer Inventory because the copy might contain inconsistent data that makes the Printer Inventory unusable.
- ▶ To use the `pidu` and `sdbu` commands, you must be a member of the AOPADMIN group and you must have UPDATE access to the AOP.ADMINISTRATOR resource profile in the PRINTSRV class. (If your security administrator has not defined profile AOP.ADMINISTRATOR, you must have READ access to the AOPADMIN profile in the FACILITY class.)

## pidu export command

The **export** command exports all objects in the Printer Inventory or only those objects that meet specified conditions. You can use the export command to backup the Printer Inventory. Also, the statements in the output file can be used as input to the pidu program on another z/OS system.

The **export** command writes a *create* statement for each exported object to an output file. The output file does not need to exist. However, if the file already exists, the file is replaced.

**export** command format:

```
export filename [objectclass [where condition [and|or condition]... ] ];
```

Where:

**filename** The name of the output file, which can be a UNIX file or an MVS data set. If the data set is an MVS data set, specify // before the filename. If you specify a fully qualified data set name, also enclose the data set name in single quotation marks and code a backslash before each single quotation mark.

**objectclass** The class of the objects that you want to export. If you omit this operand, all objects are exported.

**where condition [and|or condition]...**

One or more conditions, which can limit the objects that are exported. Only objects that are in the specified object class and that meet the conditions are exported. If you omit the where predicate, all objects in the specified object class are exported.

| <u>Object class</u> | <u>Description of object</u>                              |
|---------------------|-----------------------------------------------------------|
| allocation          | An Allocation component                                   |
| fsa                 | An FSA definition                                         |
| job-selection-rule  | A job selection definition                                |
| netspool-eof-rules  | A NetSpool End-of-File component                          |
| netspool-options    | A NetSpool Options component                              |
| printer             | A printer definition                                      |
| printer-pool        | A printer pool definition                                 |
| printway-fss        | An FSS definition for an IP PrintWay functional subsystem |
| printway-options    | An IP PrintWay Options component                          |
| processing          | A Processing component                                    |
| protocol            | A Protocol component                                      |
| psf-fss             | An FSS definition for a PSF functional subsystem          |

## Restore steps for the Printer Inventory

To restore the Printer Inventory from the latest back up:

- ▶ Stop the NetSpool task and IP PrintWay basic mode, if active
- ▶ Stop all Infoprint Server daemons.
- ▶ Move the /var/Printsrv/master.db, /var/Printsrv/jestoken.db, and pwjestoken.db files to another directory. Save these files until you have restored the Printer Inventory, restarted Infoprint Server, and run with the restored Printer Inventory for a few days.
- ▶ Start Infoprint Server daemons.
- ▶ Run the **pidu** command specifying in the *filename* operand the latest back up.

## 3.47 Setting Workload Manager goals

- ❑ NetSpool and IP PrintWay basic mode
  - High-priority started task service class
  - SYSSTC
- ❑ Infoprint Server daemons and transform daemons
  - Classification Group
    - Use Transaction Class Groups - Transaction Name Group

| Transaction Name Group          |                                 | Service Class- OMVSDMN                 |           |          |                                  |
|---------------------------------|---------------------------------|----------------------------------------|-----------|----------|----------------------------------|
| Qualifier type . . . . .        | : Transaction Name              | #                                      | Duration  | Imp      | Goal description                 |
| Group name . . . . .            | : AOP                           | 1                                      | 400       | 3        | 80% complete within 00:00:00.500 |
| Description . . . . .           | : Infoprint Server tasks        | 2                                      | 400       | 4        | 60% complete within 00:00:01.000 |
| Fold qualifier names? . . . . . | : Y (Y or N)                    | 3                                      |           | 5        | Execution velocity of 40         |
| Qualifier Name                  | Description                     | <u>Classification Rule</u>             |           |          |                                  |
| AOP*                            | Infoprint Server daemons        | Subsystem Type . . : OMVS              |           |          |                                  |
| AFP2*                           | From AFP transforms             | Description . . . OpenEdition MVS/UNIX |           |          |                                  |
| PS2AFPD                         | PostScript/PDF to AFP transform | Qualifier                              | Qualifier | Starting | Service                          |
| PCL2AFPD                        | PCL to AFP transform            | #                                      | type      | name     | position                         |
| XML2*                           | XML to AFP/PDF transform        |                                        |           |          | Class                            |
| X2AFPD                          | Xerox to AFP transform          |                                        |           |          | Report                           |
|                                 |                                 |                                        |           |          | Class                            |

Figure 3-48 Workload Manager goals

### Setting Workload Manager goals

The WLM component of z/OS lets you manage workload distribution, balance workload, and distribute resources to competing workloads.

- ▶ Put the NetSpool and IP PrintWay basic mode started tasks in a high-priority started task (subsystem type STC) service class, or assign them to SYSSTC.
- ▶ Assign the Infoprint Server daemons in a Classification Group, using Transaction Class Groups, to a Transaction Name Group (TNG). You can also put the transform daemons into this TNG.

This example uses the group name AOP and a wildcard for all daemons that start with AOP\*, AFP2\*, and XML2:

```
Qualifier type . . . . . : Transaction Name
Group name . . . . . : AOP
Description . . . . . : Infoprint Server tasks
Fold qualifier names? . . . . . Y (Y or N)
Qualifier Name Description
AOP* Infoprint Server daemons
AFP2* From AFP transforms
PS2AFPD PostScript/PDF to AFP transform
PCL2AFPD PCL to AFP transform
XML2* XML to AFP/PDF transform
X2AFPD Xerox to AFP transform
```

- Determine how important your print workload is to your installation compared with other work running on the system. In most cases, using an importance of 3 for the first period is adequate.

**Tip:** The Printer Inventory Manager daemon, aopd, must be dispatched so it can respond to system events--for example, so the Infoprint Server ENF listener exit can run. Depending on what other work is running in the system, you might need to create a short first period with a higher importance to make sure the Printer Inventory Manager is dispatched. As an alternative, you can assign daemon aopd by name to the SYSSTC service class.

- Assign the Infoprint Server daemons to a service class that uses a percentage response time goal for all except the last period. Use a velocity goal for the last period to handle long-running transactions, such as a transform. You can use the service-units-per-second value reported in the RMF SYSRPTS WLMGL options report to determine duration values. You can either create a service class for Infoprint Server daemons similar to the OMVS user service class, or you can use the same service class that you use for OMVS users.

This example shows a service class definition for service class OMVSDMN, which is used for Infoprint Server daemons:

| # | Duration | Imp | Goal description                 |
|---|----------|-----|----------------------------------|
| - | -        | -   | -                                |
| 1 | 400      | 3   | 80% complete within 00:00:00.500 |
| 2 | 400      | 4   | 60% complete within 00:00:01.000 |
| 3 |          | 5   | Execution velocity of 40         |

Assign the TNG name that you defined for Infoprint Server to the service class in Classification Rules, under the OMVS subsystem type.

This example shows the TNG named AOP, assigned to service class OMVSDMN, using the report class RAOP.

|                                        |           |           |          |         |        |
|----------------------------------------|-----------|-----------|----------|---------|--------|
| Subsystem Type . . : OMVS              |           |           |          |         |        |
| Description . . . OpenEdition MVS/UNIX |           |           |          |         |        |
|                                        | Qualifier | Qualifier | Starting | Service | Report |
| #                                      | type      | name      | position | Class   | Class  |
| -                                      | -         | -         | -        | -       | -      |
| 1                                      | TNG       | AOP       | —        | OMVSDMN | RAOP   |

## Print Interface

The Print Interface component of Infoprint Server provides z/OS UNIX System Services printing commands and a subsystem that lets you print from the local z/OS system. Print Interface accepts different data formats; transforms the data streams to EBCDIC line data, AFP, PCL, PostScript, PDF, or other data formats that the printer accepts; and allocates output data sets on the JES spool.

Print Interface also provides a line printer daemon (LPD) that lets you submit print requests from remote workstations that have TCP/IP access. In addition, Print Interface can receive print requests from remote workstations that use the Internet Printing Protocol (IPP) or the Server Message Block (SMB) printing protocol, which is standard on Windows systems. Print Interface also provides an SAP R/3 Output Management System that can receive SAP R/3 print requests.

Print Interface runs as a UNIX application that uses the services of z/OS UNIX System Services. Print Interface accepts data in any format, including but not limited to these formats:

- ▶ Line data
- ▶ MO:DCA-P (also known as AFP)
- ▶ PostScript
- ▶ PDF
- ▶ PCL
- ▶ SAP (OTF and ABAP)
- ▶ XML
- ▶ Text

Each remote print request specifies the name of a printer definition in the Printer Inventory. The printer definition can describe a printer or an e-mail destination. If local job submitters specify a printer IP address on an OUTPUT JCL statement (in the DEST=IP: parameter), but does not specify a printer definition name in the FSSDATA parameter, IP PrintWay uses values specified in the IP PrintWay default printer definition (DFLTNTRY). Print Interface uses information in the printer definition to determine how to process or transform the input data.

Print Interface can, in most cases, automatically detect the data format of the input data and validate that the printer accepts that data format. It can convert data to EBCDIC or ASCII and

call transforms provided by Infoprint Server Transforms and other optional transform products to convert data format.

## Transform Interface

This chapter describes also how to customize the Transform Interface component of Infoprint Server. Transform Interface communicates with transforms that other IBM Infoprint transform products provide.

Transforms convert data from one format to another, for example, from PCL to AFP format and from AFP to PDF format.

When processing a print request, Print Interface, NetSpool, and IP PrintWay extended mode call Transform Interface to transform data from one format to another if the administrator specifies a transform filter in the printer definition. The administrator can specify a different transform filter for each data format. Infoprint Server and Infoprint transforms provide these transform filters:

- ▶ `aoprxf.so`: This filter sends data to Infoprint Transform Manager for Linux to be transformed.
- ▶ `aopform.dll`: This filter sends data to Infoprint Manager for AIX or Windows for transform.
- ▶ An Infoprint transform filter: Some Infoprint transform products provide transform filters. For example, Infoprint Transform for AFP to Adobe PDF for z/OS provides the `afp2pdf.dll` filter, which transforms data from AFP to PDF format.

IP PrintWay basic mode transforms data by resubmitting it to the Print Interface component. This function is called the resubmit for filtering function.

z/OS job submitters can specify one of these z/OS UNIX commands to transform data without printing it:

- ▶ `remotextf` command: This command sends data to Infoprint Transform Manager for Linux for transform.
- ▶ An Infoprint transform command: Some Infoprint transform products provide z/OS UNIX transform commands. For example, Infoprint Transform for AFP to Adobe PDF for z/OS provides the `afp2pdf` command, which transforms data from AFP to PDF format.

Transform Interface communicates with the transform:

- ▶ If the transform runs on z/OS, the Infoprint Server Transform Manager (part of Transform Interface) starts and stops the transform daemons, using configuration information specified by the administrator. (The Infoprint Server Transform Manager does not start the SAP to AFP transform because this transform is not implemented as a daemon.)
- ▶ If the transform runs on a non-z/OS system, Transform Interface sends the data to the transform.



## 4.1 Print Interface software requirements

- ❑ To print using the SMB printing protocol
  - z/OS SMB print server, part of the z/OS Distributed File Service element, must be configured and active
- ❑ To use the Print Interface IPP server
  - 31-bit SDK for z/OS, Java 2 Technology Edition, V5 (5655-N98)
  - SDK for z/OS, Java 2 Technology Edition, V1 (5655-I56) at SDKlevel 1.4
- ❑ To print from SAP R/3, one of these SAP R/3 application servers is required:
  - The SAP R/3 Application Server for z/OS
  - An SAP R/3 application server for another system (such as Windows, AIX, or Solaris)
- ❑ To print SAP R/3 output to IBM AFP printers
  - Infoprint Transforms to AFP for z/OS (5655-N60)
  - Infoprint Transform Manager for Linux (5639-P51)

Figure 4-1 Software requirements for using Print Interface

### Print Interface Software requirements

This figure documents the IBM software and PTFs that are required beyond the z/OS base elements, as follows:

- ▶ To print using the SMB printing protocol, the z/OS SMB print server, which is part of the z/OS Distributed File Service element, must be configured and active.
- ▶ If you start the Print Interface IPP Server, one of these products is required:
  - 31-bit SDK for z/OS, Java 2 Technology Edition, V5 (5655-N98)
  - SDK for z/OS, Java 2 Technology Edition, V1 (5655-I56) at SDK level 1.4The IPP Server cannot use the 64-bit SDK. However, you can install it on the z/OS system.
- ▶ To submit print requests from SAP R/3, one of these SAP R/3 application servers, which can be ordered from SAP, is required:
  - The SAP R/3 Application Server for z/OS
  - An SAP R/3 application server for another system (such as Windows, AIX, or Solaris™)
- ▶ To print SAP R/3 output on IBM AFP printers on the z/OS system, one of these Infoprint transform products is required:
  - Infoprint Transforms to AFP for z/OS (5655-N60)
  - Infoprint Transform Manager for Linux (5639-P51)

## 4.2 Functions provided by Print Interface

- ❑ Line printer daemon (LPR/LPD)
- ❑ Automatic transform of data
  - Infoprint Server Transforms
  - XML Extender for z/OS
- ❑ Printing support for z/OS UNIX users and applications
- ❑ Printing support for SAP R/3 users and applications
- ❑ Printing support for Windows printing protocols
  - Server Message Block (SMB)
  - Internet Printing Protocol (IPP)
  - Infoprint Port Monitor for Windows
- ❑ Support for Print Interface subsystem (JCL DD statement SUBSYS parameter)

Figure 4-2 Functions provided by Print Interface

### Print Interface functions

The functions provided by the Print Interface include the following:

- ▶ A line printer daemon (LPD) that extends JES print capabilities to users and application programs in a TCP/IP LAN environment
- ▶ Automatic transform of data to and from the AFP data stream if Infoprint Server Transforms or Infoprint XML Extender for z/OS is installed
- ▶ Printing support for users and application programs in the z/OS UNIX System Services environment. Users and applications can print on z/OS-controlled printers, including:
  - Advanced Function Presentation (AFP) printers that are attached to z/OS using PSF
  - Infoprint Server managed TCP/IP-attached ASCII printers using IP PrintWay
  - Infoprint Server managed VTAM-controlled printers using IP PrintWay
- ▶ Printing support for users and application programs in the SAP R/3 environment. Users and applications can print on z/OS-controlled printers, including:
  - Advanced Function Presentation (AFP) printers that are attached to z/OS and that use PSF (SAP to AFP transform is required)
  - Infoprint Server managed TCP/IP-attached ASCII printers using IP PrintWay
  - Infoprint Server managed VTAM-controlled printers using IP PrintWay
- ▶ Support for Windows printing protocols:
  - Server Message Block (SMB), used by Windows 98, Windows NT, and Windows 2000
  - Internet Printing Protocol (IPP), used by Windows 2000 and Windows XP
  - Infoprint Port Monitor for Windows LPR/LPD
- ▶ Batch jobs that specify the Print Interface subsystem in the SUBSYS parameter on the DD JCL statement

## 4.3 LPR/LPD protocol - RFC 1179

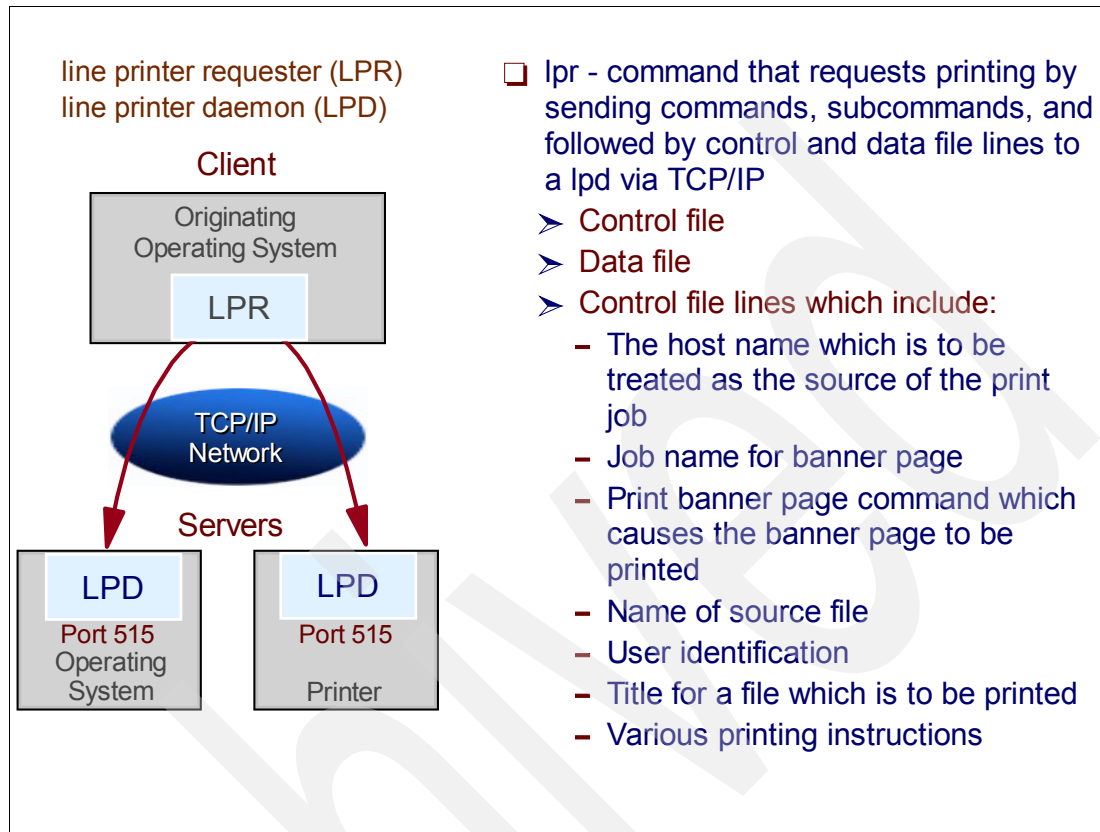


Figure 4-3 RFC 1179 LPR/LPD protocol for TCP/IP printing

### TCP/IP protocol used by Print Interface

The Network Printing Working Group has defined the RFC 1179 Request for Comments for the line printer daemon (LPD) protocol.

In the LPR/LPD printing model a group of hosts request services from a line printer daemon process running on a host. The services provided by the process are related to printing jobs. A printing job produces output from one file. The jobs are requested by users that have names. The specification of the protocol includes file formats for the *control* and *data* files as well as messages used by the protocol.

### Control file lines

The control file lines are sent to the line printer daemon. All commands begin with a single octet code, which is a binary number that represents the requested function. The code is immediately followed by the ASCII name of the printer queue name on which the function is to be performed. If there are other operands to the command, they are separated from the printer queue name with white space (ASCII space, horizontal tab, vertical tab, and form feed). The end of the command is indicated with an ASCII line feed character. The control file lines may include:

- The host name that is to be treated as the source of the print job
- Job name for banner page
- Print banner page command, which causes the banner page to be printed
- Name of source file
- User identification

- ▶ Title for a file that is to be printed
- ▶ Various printing instructions

## RFC 1179

The LPR to LPD protocol is one of the protocols that Infoprint Server uses to accept print requests and to send print data. RFC 1179 describes the existing print server protocol, which is widely used on the Internet for communicating between line printer daemons (both clients and servers). The protocol describes in detail how a line printer daemon client requests and controls printing.

LPR is a TCP-based protocol. The port on which a line printer daemon (LPD) listens is 515. The source port must be in the range 721 to 731, inclusive. An LPD responds to commands sent to its port. Many UNIX operating systems provide line printer spooling with a collection of programs:

|             |                   |
|-------------|-------------------|
| <b>lpr</b>  | Assign to queue   |
| <b>lpq</b>  | Display the queue |
| <b>lprm</b> | Remove from queue |
| <b>lpc</b>  | Control the queue |

These programs interact with an autonomous process called the line printer daemon (LPD).

**Note:** The **lpr** and **lpq** commands are supported on various platforms such as TSO/E, z/VM, AS/400®, AIX, HPUX, and UNIX platforms.

## 4.4 Print Interface customization

- ❑ z/OS Communication server TCP/IP
- ❑ z/OS UNIX shell commands (`lp`, `lpstat`, or `cancel`)
- ❑ AOPPRINT JCL procedure
- ❑ Remote clients that use the TCP/IP LPR to LPD protocol, Internet Printing Protocol (IPP), or Server Message Block (SMB) protocol
- ❑ Infoprint Port Monitor for Windows
- ❑ Infoprint Server Application Programming Interface
- ❑ SAP R/3 Application Server for z/OS
- ❑ Print Interface subsystem (SUBSYS parameter on the JCL statement)

Figure 4-4 Print Interface print request customization

### Print Interface print request customization

To be able to accept print requests, some customization has to be done to the Print Interface.

#### z/OS Communications Server

You must customize and start the IP component (TCP/IP) of z/OS Communications Server if you plan to submit print jobs to Print Interface from remote systems, use IP PrintWay to transmit print jobs to remote printers or e-mail destinations, or use the SNMP subagent.

Print Interface, IP PrintWay extended mode, and the SNMP subagent support IPv6. If you enable or disable IPv6 while Infoprint Server daemons are running, you must stop and restart the LPD (`aoplpd`), the IPP Server daemon (`aopippdxdp`), the IP PrintWay extended mode daemons (`aopoutd` and `aopwsmd`), and the SNMP subagent daemon (`aopsnmpd`).

The z/OS Communications Server customization includes:

- ▶ Updating the `hlq.PROFILE.TCPIP` data set - Special considerations exist for Print Interface and IP PrintWay for these statements: `TCPCONFIG`, `PORT`, and `AUTOLOG`.
- ▶ Updating the `BPXPRMxx` member of `SYS1.PARMLIB` - You should increase the number of active z/OS UNIX processes that the z/OS system allows.

#### z/OS UNIX shell commands

z/OS UNIX shell commands (`lp`, `lpstat`, and `cancel`) - Print Interface provides enhanced z/OS UNIX shell commands for printing: `lp`, `lpstat`, and `cancel`. There are environment

variables that affect the behavior of these commands. You can set the environment variables in the `/etc/profile` file so that they apply for all users of the commands.

## **AOPPRINT JCL procedure**

AOPPRINT JCL procedure - Print Interface provides the AOPPRINT JCL procedure in SYS1.PROCLIB so that local users can submit print requests from a batch job step.

## **Remote client print requests**

Remote clients can use the TCP/IP LPR to LPD protocol, Internet Printing Protocol (IPP), or Server Message Block (SMB) protocol. The Print Interface LPD accepts print requests from clients that use the LPR to LPD protocol as defined by RFC 1179, detects the input data stream and transforms data streams, validates that documents can print as requested, allocates jobs on the JES spool, and provides notification of job completion and job status.

Print Interface, with support that the z/OS SMB server provides, lets Windows users print using the standard Windows Server® Message Block (SMB) protocol. Print Interface validates that the document can print on the selected printer, transforms data from one format to another (optional), and allocates output data sets on the JES spool. Print Interface can also provide job status to the z/OS SMB server.

To submit a print request over the Internet using the IPP protocol, the user's workstation must contain an IPP client. Using an IPP client, the job submitter can print any document (not only Web documents) on any printer defined in the Printer Inventory. The user must first define the printer to the IPP client on the workstation. For example, on a Windows 2000 system, the user must define the printer to Windows as an Internet printer.

## **Infoprint Port Monitor for Windows**

Infoprint Port Monitor for Windows allows a user to submit a print request from LPR clients to the Print Interface LPD, a user typically uses a TCP/IP command such as the LPR command. The Infoprint Port Monitor for Windows also uses the LPR protocol to submit a print request to the Print Interface LPD. You must provide users of **LPR** commands and the Infoprint Port Monitor for Windows with information to define printers.

Infoprint Server also supports printing from a Windows system with the Server Message Block (SMB) protocol and the Internet Printing Protocol (IPP). To use these protocols, Windows users do not need to install the Infoprint Port Monitor for Windows.

## **Infoprint Server API interface**

For the Infoprint Server Application Programming Interface, the API customization task is to define environment variable `AOP_64K_JOBID`. If this environment variable is defined, the `CreatePrintFile` function limits the job ID of a print job to the range of 1 to 65535. If this environment variable is not defined, the job ID can be in the range of 1 to 999999.

## **SAP R/3 Application Server for z/OS**

SAP R/3 Application Server for z/OS - Print Interface lets SAP R/3 users print on any printers defined in the Printer Inventory. If the target printer is an IBM AFP printer, Print Interface can automatically invoke the SAP to AFP transform that Infoprint Server Transforms provide to convert SAP OTF and SAP ABAP data to the AFP format.

## **Batch job support**

Batch jobs can specify the Print Interface subsystem in the SUBSYS parameter on a DD JCL statement. When customizing the `aopd.conf` file, `inventory = inventory` provides the four

uppercase letter name of both the Printer Inventory and the Print Interface subsystem. The job submitter specifies this name in the SUBSYS parameter on the DD JCL statement.

Typically, one printer definition is created for each printer. However, creating printer definitions to be used just with the Print Interface subsystem is simplified because it is necessary to create only one printer definition for all printers that share the same attributes. The print jobs must specify the JCL parameters that are required to direct the output to the desired printer.

The following example shows the DD and OUTPUT statements that are used to direct output to a PSF printer named *anyafpprinter*:

```
//JOB1   JOB ...  
//STEP1  EXEC PGM=USERA  
//OUTDS1 OUTPUT CLASS=F,DEST=PRT003  
//DD1    DD SUBSYS=(AOP1,'anyafpprinter'),OUTPUT=(*.OUTDS1)
```

In general, to direct an output data set to IP PrintWay, you must specify on the DD or OUTPUT JCL statement for a print data set the work-selection criteria defined to JES for the IP PrintWay basic mode functional subsystem application (FSA) or the job-selection criteria defined in the Printer Inventory for IP PrintWay extended mode.

## 4.5 Print Interface - aoplpd daemon (LPR/LPD)

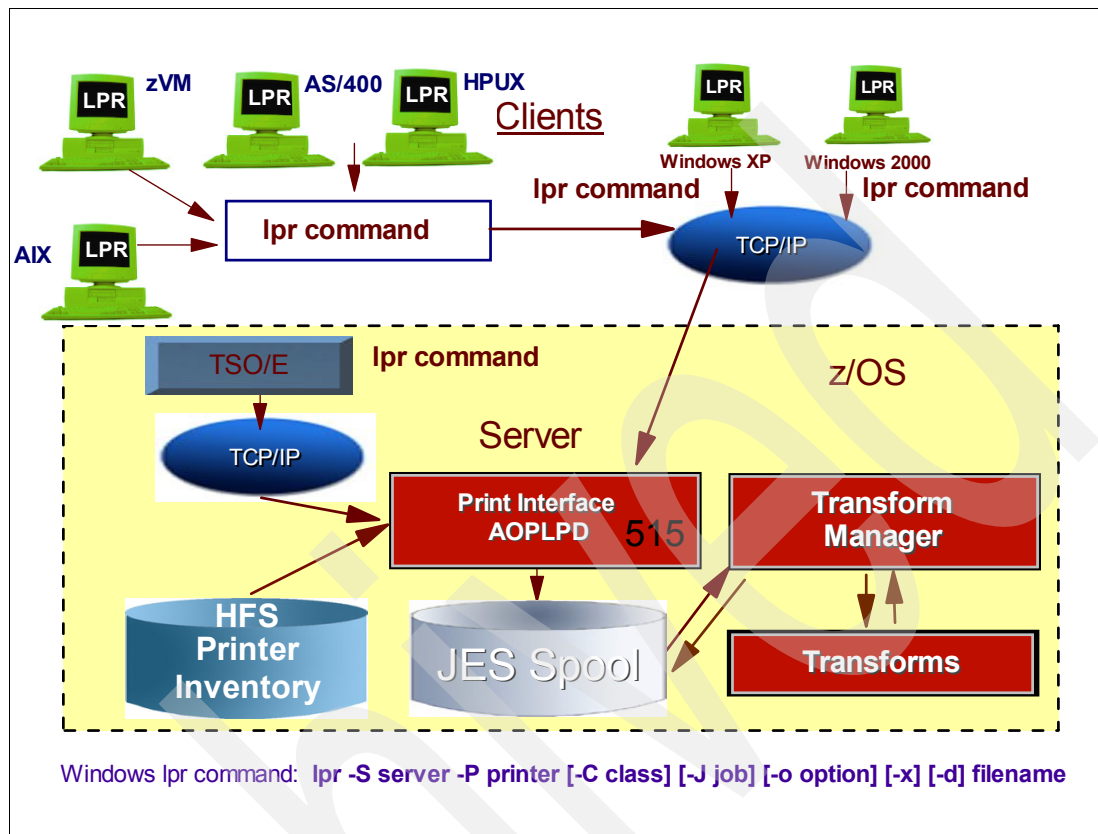


Figure 4-5 Infoprint Server and LPR to LPD protocol

### LPR to LPD support

The Print Interface implementation use of LPR/LPD protocol allows workstation clients, using the `lpr` command, to send print requests to a server, the Print Interface address space which acts as an **LPD**.

### Print requests

Print request can be submitted to Infoprint Server with commands that use the LPR to LPD protocol:

- LPR and LPQ commands from TSO/E, z/VM, HPUX, AS/400, UNIX, and Windows systems. An example of `lpr` command is the Windows command:

```
lpr -S server -P printer [-C class] [-J job] [-o option] [-x] [-d] filename
-S server   Name or ip address of the host providing lpd service
-P printer  Name of the print queue
-C class    Job classification for use on the burst page
-J job      Job name to print on the burst page
-o option   Indicates type of the file (by default assumes a text file)
-d          Send data file first
```

- `lpr`, `enq` and `qstat` commands from AIX systems.

The Print Interface LPD accepts print requests from clients that use the LPR to LPD protocol, detects the input data stream and using Printer Inventory printer definitions validates that



documents can print as requested, transforms data streams, and allocates print data sets for the requests on the JES spool.

## Filters provided with Infoprint Server

Infoprint Server provides these DLL filters in directory /usr/lpp/Printsrv/lib:

► **aopfiltr.so:**

Converts ASCII line-feed controls (X'0A') that are not preceded by carriage-return controls to carriage-return and line-feed controls (X'0D0A'). The X'0D0A' control is suitable for most ASCII printers and print queues.

► **aoprform.dll:**

Sends data to be transformed to Infoprint Manager for AIX or Windows.

► **aoprxf.so:**

Sends data to be transformed to Infoprint Transform Manager for Linux.

► **lpd\_compat.so:**

Provides an LPD filter with similar function to the z/OS Communications Server LPD.

## Using the LPD compatibility filter

The LPD compatibility filter, `lpd_compat.so`, provides support for some LPD command codes and parameters that the Print Interface LPD does not otherwise support. LPD command codes and parameters are specified in the LPD control file sent by the LPR client with each document to be printed.

The LPD compatibility filter provides support that is similar to the support that the z/OS Communications Server LPD provides. For example, with the LPD compatibility filter, the Print Interface LPD supports the **p** command code, which lets you print a header with a page number on each page. The LPD compatibility filter also lets job submitters and the administrator specify the **-f**, **-l**, and **-w** filter options, which correspond to the **FILTER**, **LINECOUNT**, and **WIDTH** parameters of the TCP/IP LPR command.

The LPD compatibility filter can be used for printing to PSF-controlled printers and to other printers that print line data.

## Guidelines

If you want the functions that filter `lpd_compat.so` provides, specify it for the Text and Line data formats in PSF printer definitions, General printer definitions, and in IP PrintWay printer definitions when you select the VTAM protocol.

*Do not* specify filter `lpd_compat.so` in IP PrintWay printer definitions except when you select the VTAM protocol.

## 4.6 Print Interface validating requests

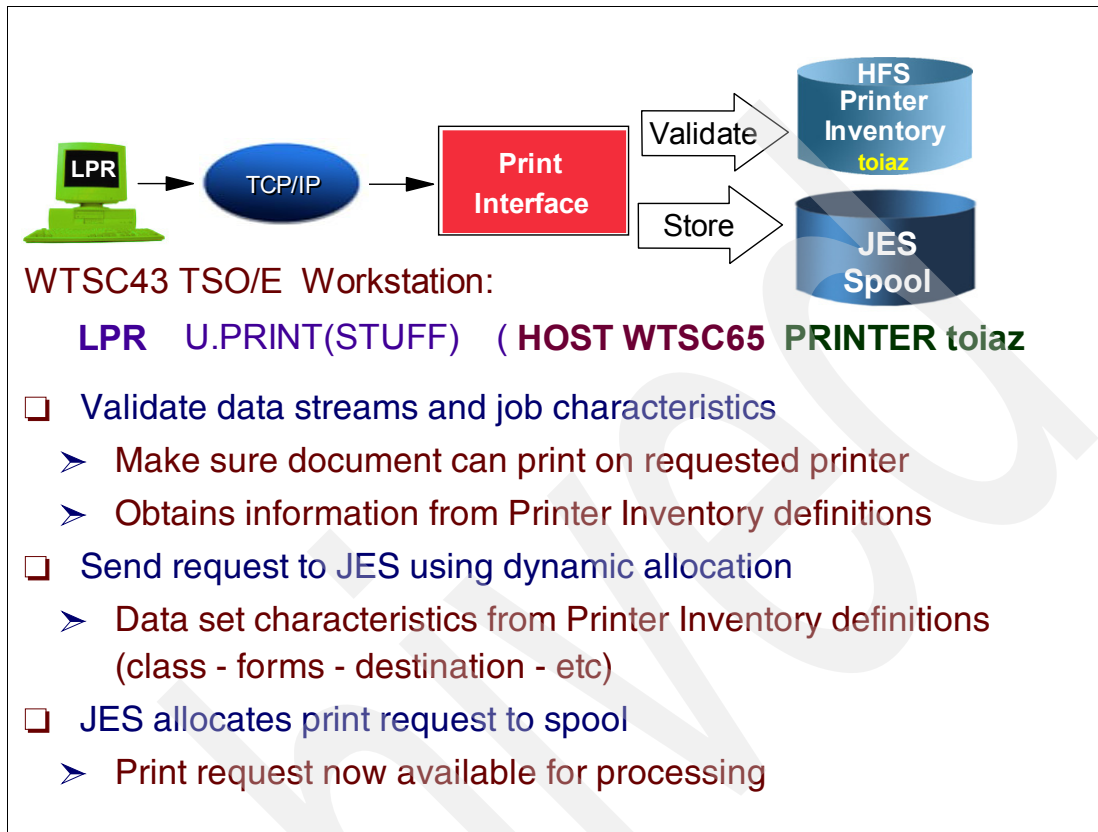


Figure 4-6 Validation of print requests to JES spool

### Print request validation

Before accepting a print request, Print Interface validates that the document can print as requested on the target printer. For example, Print Interface can verify that the target printer's Processing section specification supports the data format of the input document. If Print Interface determines that a document cannot print, Print Interface rejects the print request with a message and does not allocate a data set on the JES spool.

Print Interface can reject documents with data formats that the printer does not support or that are too large to print on the selected printer. If Print Interface cannot determine the input data format, the data format defaults to *other*.

Spool space for the print data in the request is dynamically allocated from the JES. The print request characteristics are taken from the Printer Inventory entry for the specified printer.

The print data will be available to be selected and printed.

### Data set characteristics

Infoprint Server lets the administrator specify printer characteristics for each printer in the Printer Inventory. The user can specify print request attributes in different ways, depending on the method used to submit a print request. These print request characteristics are used to pass them to JES when the data is being placed on the JES spool.

## 4.7 Print Interface and z/OS UNIX printing commands

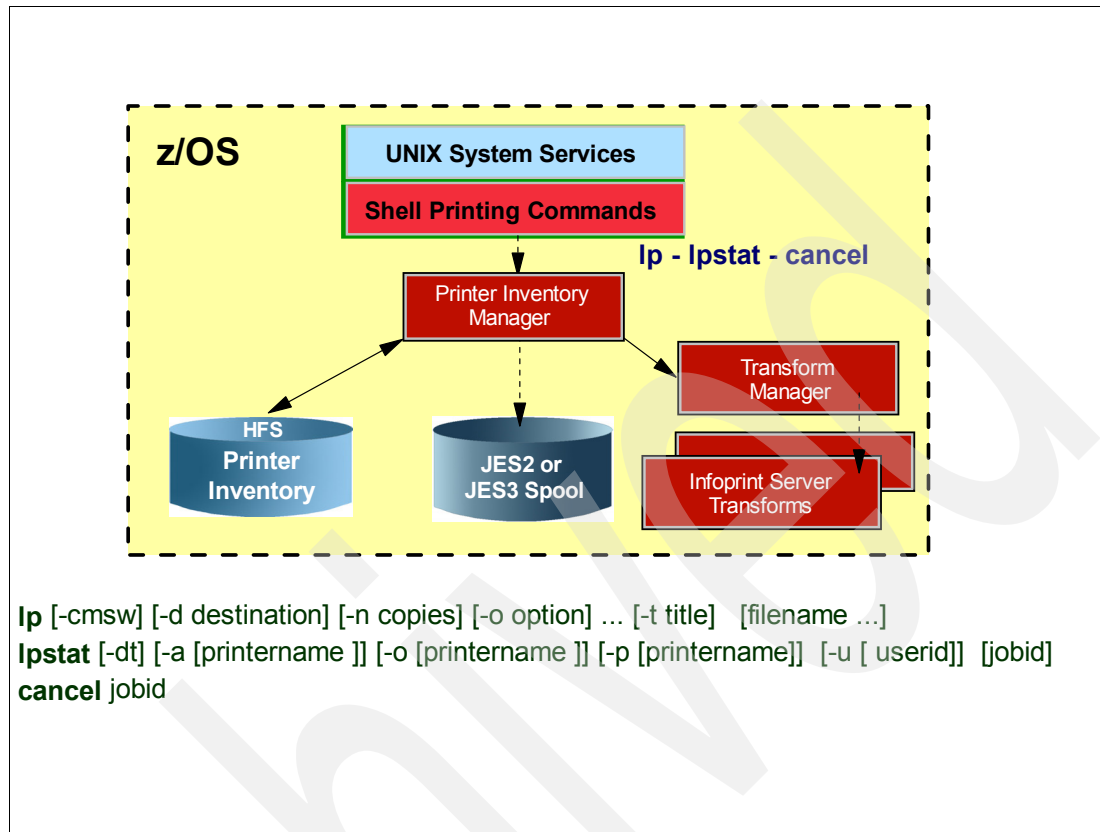


Figure 4-7 z/OS UNIX printing commands

### z/OS UNIX printing commands

Print Interface includes enhanced z/OS UNIX System Services shell printing commands, which provide more function than the printing commands available without Infoprint Server, as follows:

**lp** The **lp** command sends files for printing to Print Interface running on the local z/OS system. The files can be UNIX files or traditional MVS data sets, such as sequential data sets and partitioned data sets. If Infoprint Server Transforms product is installed, Infoprint Server automatically transforms a file from one data format to another. To transform a file, the administrator must request the transform in the printer definition.

The **lp** command returns an Infoprint Server job ID, which you can use to query or cancel the job.

```
lp [-cmsw] [-d destination] [-n copies] [-o option] ... [-t title]
[filename ...]
```

**lpstat** The **lpstat** command queries the status of print jobs. It also queries the names, locations, and descriptions of printers that the administrator has defined in the Printer Inventory.

```
lpstat [-dt] [-a [prntername ...]] ... [-o [prntername ...]] ...
      [-p [prntername ...]] ... [-u [userid ...]] ... [jobid ...]
```

**cancel** The **cancel** command cancels print requests, provided that the data set allocated on the JES pool has not yet been selected for printing.

## 4.8 Print requests from z/OS to Print Interface

### ❑ Print request from TSO/E

#### ➤ **lpr 'sys1.proclib(JES2)' (host 9.12.4.54 printer INFO40**

Note: Printer names in Printer Inventory are case sensitive.

The TSO/E **LPR** command has to be listed as an **AUTHCMD** in **IKJTSOxx** parmlib member.

### ❑ Message on SYSLOG - (ITSO modification)

```
AOPD00I AOPLPD (JOB29731) Prt: INFO40      - WTSCPLX9 .....  
..... ROGERS PS000253 LIB#JES2 from: ROGERS@wtsc43.itso.ibm.com
```

### ❑ Print request from OMVS shell

#### ➤ **ROGERS @ SC43:/>lp -d POKEGEN //test.jcl** **AOP007I Job 256 successfully spooled to POKEGEN.**

```
AOPD00I ROGERS3 (JOB31283) Prt: POKEGEN      - WTSCPLX9 .....  
.....ROGERS PS000256 TEST#JCL from: ROGERS / SC43
```

Figure 4-8 Print requests from TSO/E and OMVS shell

### Print requests

The Print Interface component of Infoprint Server processes print requests received from both remote clients and local users such as TSO/E and the OMVS shell user.

### Print request from TSO/E

From TSO, you use the LPR command to send a print request to Infoprint Server. In the following LPR command, the data set to print is the JES2 procedure from SYS1.PROCLIB and the printer name is INFO40. The host IP address is specified and the request goes to the Print Interface LPD.

```
lpr 'sys1.proclib(JES2)' (host 9.12.4.54 printer INF040
```

**Note:** Because Infoprint Server does not produce any messages regarding this print request arriving on the JES spool, our user modification puts the AOPD001 message in the SYSLOG to indicate the user ID of the submitter, the jobid assigned to the print request, PS000253, and the IP address in the form domain name where Infoprint Server is running.

### Print request from OMVS shell

After receiving a print request, Infoprint Server returns either an error message or a job identifier. The job identifier indicates that Infoprint Server has accepted the print request. You can use the job identifier to query the status of the print request or to cancel the print request.

```
lp -d POKEGEN //test.jcl
```

## 4.9 Print Interface processing for print requests

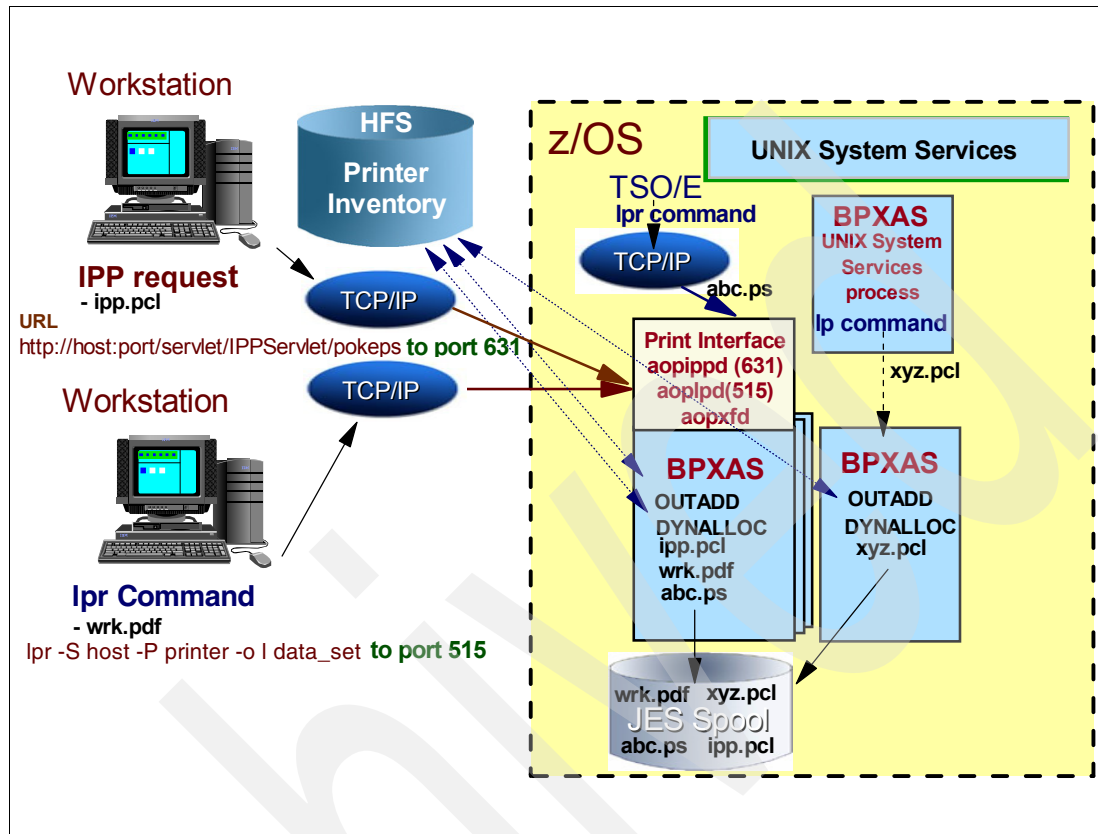


Figure 4-9 Print Interface processing requests

### Request processing by Print Interface

Figure 4-9 shows the Print Interface processing steps for following requests:

- ▶ An lpr command issued from a workstation user
- ▶ An lpr command issued by a TSO user
- ▶ An lp command issued by a z/OS UNIX user
- ▶ An IPP request

### Print Interface address space (process)

Figure 4-9 shows processing steps for three different type print requests: IPP, lpr, and lp. When the print data arrives, via TCP/IP on ports 515 (lpr) and 631 (IPP) and by a forked address space for the z/OS UNIX lp command, Print Interface validates the specified target printer and the data stream's validity for the printer by checking the Printer Inventory entry.

### MVS dynamic allocation

Print Interface uses MVS dynamic output services (OUTADD macro), when it retrieves printer information from the Printer Inventory, to pass data set characteristics to JES along with the dynamic allocation function (DYNALLOC macro) to allocate spool space for the incoming print data sets.

## Printing the output

After Print Interface has created an output data set on the JES spool, the output can be printed by:

- ▶ JES printers or PSF printers.
- ▶ JES can transmit the data sets to another location for printing using NJE.
- ▶ Transmit the output via IP PrintWay.

## Output characteristics

The OUTADD macro is used by Print Interface to create output descriptors from the characteristics defined in the Printer Inventory entry for a target printer.

## Data sets allocated to spool

Print Interface allocation of the output (using dynamic allocation) requests JES to allocate spool space for a new SYSOUT data set.

From a JES perspective, the SYSOUT data set is queued to the BPXAS job that dynamically allocates a data set to the spool.

Infoprint Server assigns its own unique job-ID to the spooled print data set using the job-prefix that is defined in the Infoprint Server configuration file `/etc/Printsrv/aopd.conf`.

In some situations, the job-prefix is not used. For example:

- ▶ If a job submitter specifies the `sysout-job-id` job attribute, the job ID of the data set on the JES spool is the job ID specified in the job attribute.  
The single-valued attribute specifies the job ID to assign to the system output data sets (sysout data sets) that Infoprint Server creates on the JES spool. The system operator can use this job ID to locate the sysout data sets on the JES spool.
- ▶ If a job submitter submits a data set to the Print Interface subsystem, the job ID of the data set on the JES spool is the job ID that JES assigned to the job.
- ▶ If IP PrintWay basic mode resubmits a print job to Print Interface for filtering, the job ID of the data set on the JES spool is the job ID that JES assigned to the original job.

**Note:** JES also assigns a unique z/OS job ID to each output data set. The JES job ID might be different from the job ID of the output data set on the JES spool. JES operator commands use this JES job ID.

The job ID that Print Interface returns to the job submitter does not contain this prefix.

## 4.10 Printing commands customization

- ❑ ISPF dialog *Infoprint Server: Printer Inventory Manager Option 7 Configure* defines the name of the **default** printer definition used when:
  - The **lp** command does not specify a printer definition name and the LPDEST and PRINTER environment variables are not set
  - The Print Interface subsystem is used and no printer definition name is specified in the JCL
- ❑ The environment variables that override the default printer definition name for the **lp** command in */etc/profile* apply for all users
  - AOPTIONS - job attributes and values for **lp**
  - AOPATH - directory for attribute files used by **lp**
  - MANPATH - path of directories for man pages
  - PRINTER or LPDEST - default printer for **lp**

Figure 4-10 Specifying z/OS UNIX default printer and environment variables

### Customizing the z/OS UNIX printing commands

Print Interface provides enhanced z/OS UNIX shell commands for printing: **lp**, **lpstat**, and **cancel**. Information on how to use these commands is in *z/OS Infoprint Server User's Guide*.

### Default printer for z/OS UNIX

The default Infoprint Server printer definition is used in these situations:

- |                                  |                                                                                                                                                               |
|----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>lp command</b>                | When the job submitter does not specify a printer definition on the <b>lp</b> command and the PRINTER and LPDEST environment variables are not set            |
| <b>Print Interface subsystem</b> | When the printer definition is omitted in the SUBSYS parameter on the DD JCL statement and the FSSDATA parameter is not specified on the OUTPUT JCL statement |

### Environment variables

You can set environment variables for all users of the **lp**, **lpstat**, and **cancel** commands in the */etc/profile* file:

- AOPTIONS** Specifies a string of job attributes and values that are to be in effect for each **lp** command. The **lp** command includes the attributes specified in this environment variable before the attributes specified, if any, on the **-o** option of the **lp** command. Because the **lp** command reads the value of the AOPTIONS environment variable before the options you specify on the

command line, a user can override the values of this variable. This variable is optional. Use this environment variable to specify job attributes that are constant for all print jobs.

**AOPPATH** The directory that contains attribute files used by the **lp** command. This variable is optional. If you have not created attribute files for the **lp** command, you do not need to set this variable. See an example in Figure 4-11 on page 171.

**MANPATH** The path of directories that contain the man pages. Infoprint Server man pages are available only in English. Add `/usr/lpp/Printsrv/man/En_US` to the values in this variable. This variable is required.

**Tip:** Add the new directory path before `/usr/man/%L` in the **MANPATH** variable so that the Infoprint Server versions of the `lp`, `lpstat`, and `cancel` man pages will be displayed.

## **PRINTER or LPDEST**

The **PRINTER** or **LPDEST** variable specifies the default printer for the **lp** command. The value in **LPDEST** overrides the value in **PRINTER**. This variable is optional. You can also use Infoprint Serve ISPF panels to define a default printer. The printer named in either **LPDEST** or **PRINTER** overrides the printer named on the ISPF configuration panel.

**Note:** To use the z/OS UNIX printing commands, you do not need to start any daemons in addition to the Printer Inventory Manager daemon.



## 4.11 Using attribute files

- ❑ AOPPATH environment variable defines the directory that contains attribute files
- ❑ Attributes files predefine attribute and value pairs in permanent files
  - The permanent attribute files can be used with the lp, afp2pcl, afp2pdf, afp2ps, and remotexf commands
- ❑ Example: Print file File5 using two attributes files
  - default.att and special.att
    - lp -o "attributes=default.att attributes=special.att" File5
  - File default.att contains these lines:
    - input-tray=bottom
    - duplex=yes
    - output-bin=collator
  - File special.att contains these lines:
    - input-tray=top
    - copies=5
    - title-text='Special Report'

Figure 4-11 Printing from z/OS UNIX using attribute files

### Attribute files

You can predefine attribute and value pairs in permanent files. You can use the permanent attribute files when you need those attribute values with the lp, afp2pcl, afp2pdf, afp2ps, and remotexf commands. You can also use the permanent attribute files when you need to specify those values with the AOPPRINT JCL procedure and when you use the Print Interface subsystem.

### Creating an attributes file

Some considerations for creating an attribute file follow.

- ▶ You can list any job attribute in an attributes file.
- ▶ You can also use the attribute *attributes*. Thus, an attributes file can call other attributes files.
- ▶ Attributes files must not contain any attributes without values.
- ▶ When creating an attributes file, consider spelling out the complete attribute names and attribute values rather than using abbreviations.
- ▶ You can use spaces between the attribute name and the equal sign to align the equal sign and values. This makes your files easier to read and maintain.
- ▶ You can use comment lines in attributes files. The comment starts with a number sign, #, and ends at the end of line.

**Example:** You could create an attributes file called myatts to request 5 copies of a job, simple duplex printing, and a specific output bin. Your file contains these lines:

```
# These are my job attributes
copies      = 5
duplex      = yes
output-bin  = collator # Collate the job
```

Note that you can include a number sign, #, as part of an attribute value if you precede it immediately with a backslash, \#.

### Using an attributes file

To print file File5 on the default printer and to specify the two attributes files default.att and special.att, enter:

```
lp -o "attributes=default.att attributes=special.att" File5
```

Suppose that the file default.att contains these lines:

```
input-tray=bottom
duplex=yes
output-bin=collator
```

The file special.att contains these lines:

```
input-tray=top
copies=5
title-text='Special Report'
```

The preceding command is equivalent to this command:

```
lp -o "input-tray=top duplex=yes output-bin=collator copies=5
title-text='Special Report'" File5
```

The value of input-tray in special.att overrides the value in default.att because you specified special.att last.

## 4.12 RACF OMVS segment and aopd.conf file updates

- ❑ **MAXPROCSYS** - max number processes/system - BPXPRMxx
  - 200 for NetSpool and Print Interface
  - 50 for IP PrintWay extended mode
  - 70 for each IP PrintWay basic mode FSS
- ❑ **MAXPROCUSER** - maximum number of processes/user - BPCPRMxx
  - At least 70
- ❑ **Updating the OMVS segment of identified daemon user ID**
  - **PROCUSERMAX** - maximum number of processes/user
    - At least 70
- ❑ **Editing the aopd.conf configuration file for Print Interface**
  - **ascii-codepage** = codepage - ISO8859-1
  - **job-prefix** = prefix - PS
  - **log-retention** = days - 3
  - **lpd-port-number** = portnumber - 515
  - **ipp-port-number** = portnumber - 631

Figure 4-12 BPXPRMxx and aopd.conf updates

### Updating the BPXPRMxx member of SYS1.PARMLIB

When you use Print Interface, you should increase the number of active z/OS UNIX processes that the z/OS system allows. You can specify the number of allowed processes in these parameters in the BPXPRMxx member of SYS1.PARMLIB:

#### ► MAXPROCUSER

The maximum number of processes that a single z/OS UNIX user ID can have concurrently active.

Recommendation: Specify at least 70.

#### ► MAXPROCSYS

The maximum number of processes that the z/OS system allows.

Recommendation: Add 200 for processes used by both NetSpool and Print Interface. If you run IP PrintWay, add an additional 50 (extended mode) or an additional 70 for each IP PrintWay FSS (basic mode).

**Note:** You can use the SETOMVS or SET OMVS command to dynamically increase the MAXPROCUSER and MAXPROCSYS values.

## Updating the RACF user profile OMVS segment

Information stored in the OMVS segment of the user's profile includes PROCUSERMAX(processes-per-UID).

PROCUSERMAX(processes-per-UID) specifies the maximum number of processes this user is allowed to have active at the same time, regardless of how the process became a z/OS UNIX process. The processes-per-UID you define to RACF is a numeric value between 3 and 32767. PROCUSERMAX is the same as the CHILD\_MAX variable defined in the POSIX standard.

Recommendation: When you use Print Interface specify at least 70.

Example: Change maximum number of processes for user ID AOPSTC:

```
ALTUSER AOPSTC OMVS(PROCUSERMAX(70))
```

## Editing the aopd.conf configuration file

Print Interface uses the following attributes in the aopd.conf configuration file. Add or edit these attributes to customize Print Interface for your installation.

### job-prefix = prefix

A prefix that Print Interface and NetSpool use for the job IDs of output data sets that they allocate on the JES spool. For example, if you specify prefix IS, the job ID of a print job might be IS001234. This prefix can help you identify data sets that Print Interface and NetSpool allocate on the JES spool.

Specify 2 letters, numbers, or national (@ \$ #) characters. Enclose a prefix that contains national characters in single or double quotation marks. The first character cannot be numeric.

In some situations, this prefix is not used. For example:

- ▶ If the sysout-job-id job attribute is used
- ▶ If SUBSYS parameter on DD JCL statement is used
- ▶ If IP PrintWay basic mode resubmits a print job to Print Interface for filtering

JES assigns a unique z/OS job ID to each output data set. The z/OS job ID might be different from the job ID that the Infoprint Server assigns to print requests. JES operator commands use z/OS job ID.

The job ID that Print Interface returns to the job submitter does not contain the prefix.

If you change this attribute while Infoprint Server is running, stop and restart all Infoprint Server daemons so that your changes take effect.

Default: job-prefix = PS

### ascii-codepage

Specify the default ASCII code page Print Interface uses when it converts data between ASCII and EBCDIC. If not specified, the default is code page ISO8859-1.

### log-retention

Specify how many days worth of messages Print Interface writes to the common message log. If not specified, Print Interface writes messages only to the console log.

### **lpd-port-number**

The number of the port at which the Infoprint Server LPD waits for print requests. Port 515 is the well-known port for communication between LPRs and LPDs. If you specify a port other than 515, make sure that the port is not used by any other service on the z/OS system. Windows users must specify this port number when they configure the Infoprint Port Monitor for Windows. This attribute is optional. This port number must not be reserved in the TCPIP profile data set. Assign port 515 to the Print Interface LPD so that you can use all of the functions that Infoprint Server provides.

### **ipp-port-number**

The number of the port at which the IPP server waits for print requests. Port 631 is the well-known port for communication between IPP clients and IPP servers. If you specify a port other than 631, make sure that the port is not used by any other service on the z/OS system. If you change this attribute while Infoprint Server is running, stop and restart the IPP daemon.

Default: `ipp-port-number = 631`

### **start-daemons**

The Infoprint Server daemons that are started when you use the **aopstart** command. You can specify one or more of these daemon names, enclosed in braces.

`start-daemons = {lpd ippd}` starts *aoplpd* daemon and the IPP Server daemon *aopippd*.

Default: `start-daemons = {lpd}`

## 4.13 Customizing the AOPPRINT JCL procedure

### ❑ AOPPRINT JCL procedure in SYS1.PROCLIB

- Users can submit print jobs from a batch job

```
/*-----  
/* AOPPRINT - z/OS Infoprint Server  
/*-----  
//AOPPRINT PROC  
PRINTER='p1',OPTIONS=,OUTCLASS='*',ERRCLASS='*'  
//LP EXEC PGM=AOPBATCH,  
// PARM='/lp -d &PRINTER -o "&OPTIONS" //DD:SYSIN'  
//STDOUT DD SYSOUT=&OUTCLASS  
//STDERR DD SYSOUT=&ERRCLASS  
/* STDENV may point to a dataset containing environment variables  
/* Builtin values will work for the default installation  
/*STDENV DD DSN=USERID.JCL(ENVVARS),DISP=SHR
```

To use the AOPPRINT procedure, you do not need to start any daemons in addition to the **Printer Inventory Manager** daemon.

Figure 4-13 Customizing the AOPPRINT JCL procedure

### Customizing the AOPPRINT JCL procedure

The AOPPRINT Job Control Language procedure, provided in SYS1.PROCLIB, lets you submit print requests from z/OS. This procedure lets you take advantage of all the features of Infoprint Server:

- ▶ You can specify job attributes.
- ▶ If an IBM Infoprint transform product is installed, you can automatically transform jobs from one data format to another.
- ▶ Infoprint Server validates that data can print on the selected printer.

The AOPPRINT JCL procedure in SYS1.PROCLIB uses the **lp** command to submit print requests.

```
/*-----  
/* AOPPRINT - z/OS Print Server batch print procedure  
/*-----  
//AOPPRINT PROC PRINTER='p1',OPTIONS=,OUTCLASS='*',ERRCLASS='*'  
//LP EXEC PGM=AOPBATCH,  
// PARM='/lp -d &PRINTER -o "&OPTIONS" //DD:SYSIN'  
//STDOUT DD SYSOUT=&OUTCLASS  
//STDERR DD SYSOUT=&ERRCLASS  
/* STDENV may point to a dataset containing environment variables.  
/* Builtin values will work for the default installation.  
/*STDENV DD DSN=USERID.JCL(ENVVARS),DISP=SHR
```

You can customize the AOPPRINT JCL procedure for your installation in the following ways:

- ▶ To change the name of the default printer definition, specify the printer definition name in the PRINTER option on the PROC statement. If you do not change the name, the default is p1. The administrator should create a printer definition with the default name.
- ▶ To define environment variables used by the Printer Inventory Manager, define environment variables in an STDENV DD statement. If your installation installed Infoprint Server files in the default directory, /usr/lpp/Printsrv/, you do not need to specify the LIBPATH, NLSPATH, and PATH environment variables. To define environment variables, specify each variable on a separate line, in the format variable=value:

```
LIBPATH=/usr/mylib  
PATH=/usr/mylib
```

- ▶ If you have not added the Language Environment run-time library (CEE.SCEERUN) or the C++ run-time library (CBC.SCLBDLL) to the system LNKST, concatenate these data sets to a STEPLIB DD statement.

### PROC statement parameters

The following symbolic parameters are used on the EXEC statement:

#### **ERRCLASS=class**

Specifies the 1 character alphanumeric name of the system output data set (SYSOUT) class for error messages. The name of the data set where error messages are written is specified by the STDERR data definition name (DDname).

#### **OPTIONS='attribute=value ...'**

Specifies job attributes and values to use in processing the job. Using an attributes file, you can store attributes and values in a UNIX file (such as an HFS file) or in an MVS data set. Use the attribute called attributes to specify the file or DD name from which attributes are to be read, as in the next example.

#### **OPTIONS='attributes=bigjob.att'**

If the attributes are stored in an MVS data set, you must specify four single quotation marks before the data set name and four single quotation marks after it.

#### **OUTCLASS=class**

Specifies the 1 character alphanumeric name of the SYSOUT class for informational messages. The name of the data set where informational messages are written is specified by the STDOUT DDname.

#### **PRINTER=printer\_definition\_name**

Specifies the name of a printer definition created by your administrator. The printer definition identifies a printer or an e-mail destination and sets default values for transform options and for some job attributes. The name of the printer definition is case-sensitive. Enter it exactly as your administrator tells you.

To use the AOPPRINT procedure, you do not need to start any daemons in addition to the Printer Inventory Manager daemon.

## 4.14 AOPPRINT examples

```
* Print JES2 procedure from SYS1.PROCLIB
//ROGAOPP JOB    (POK,999),MSGCLASS=A,NOTIFY=ROGERS
//PRINT1 EXEC AOPPRINT,PRINTER='POK45AN'
//SYSIN DD      DSN=SYS1.PROCLIB(JES2),DISP=SHR
* Send e-mail
//ROGERSY JOB    (POK,999),MSGCLASS=A,NOTIFY=ROGERS
//EMAIL EXEC AOPPRINT,PRINTER='EROSS',
//      OPTIONS='attributes=//DD:MYATTR'
//SYSIN DD      DSN=ROGERS.JCL.VERS5(AOPPRINT),DISP=SHR
//MYATTR DD      *
mail-to-addresses={'paulroge@us.ibm.com'}
mail-cc-addresses={'vaini@fi.ibm.com'}
mail-bcc-addresses={'kgoeze@de.ibm.com'}
mail-from-name="Paul C. Rogers"
mail-reply-address='paulroge@us.ibm.com'
mail-file-name="Redbook Draft"
title-text="ABCs of z/OS System Programming Volume 7"
form-definition=STD
* Other attribute examples
OPTIONS='copies=2'           Print 2 copies
OPTIONS='hold=true'         Place output in operator hold
OPTIONS='document-codepage=ISO8859-1' Specify ASCII code page
```

Figure 4-14 Examples of AOPPRINT batch jobs

### AOPPRINT batch job examples

After Infoprint Server accepts the print job, AOPPRINT returns an Infoprint Server job ID in the STDOUT data set. For example:

AOP007I Job 14584 successfully spooled to myprinter.

The returned job ID can be used in the **lpstat** and **cancel** commands to query and cancel the job. (The JES3 output service inquiry command supports CJ= client job parameter to display SYSOUT data sets using the Infoprint Server job ID.) The SYSOUT data set that AOPPRINT procedure creates has the same job name and job ID as the job that ran the AOPPRINT procedure.

### Options parameter examples

Using the Options parameter you can specify various options for printing, including:

- ▶ Print multiple copies of a data set
- ▶ Submit and hold a job
- ▶ Specify a code page for ASCII jobs
- ▶ Send e-mail



## 4.15 Print Interface and IPP protocol

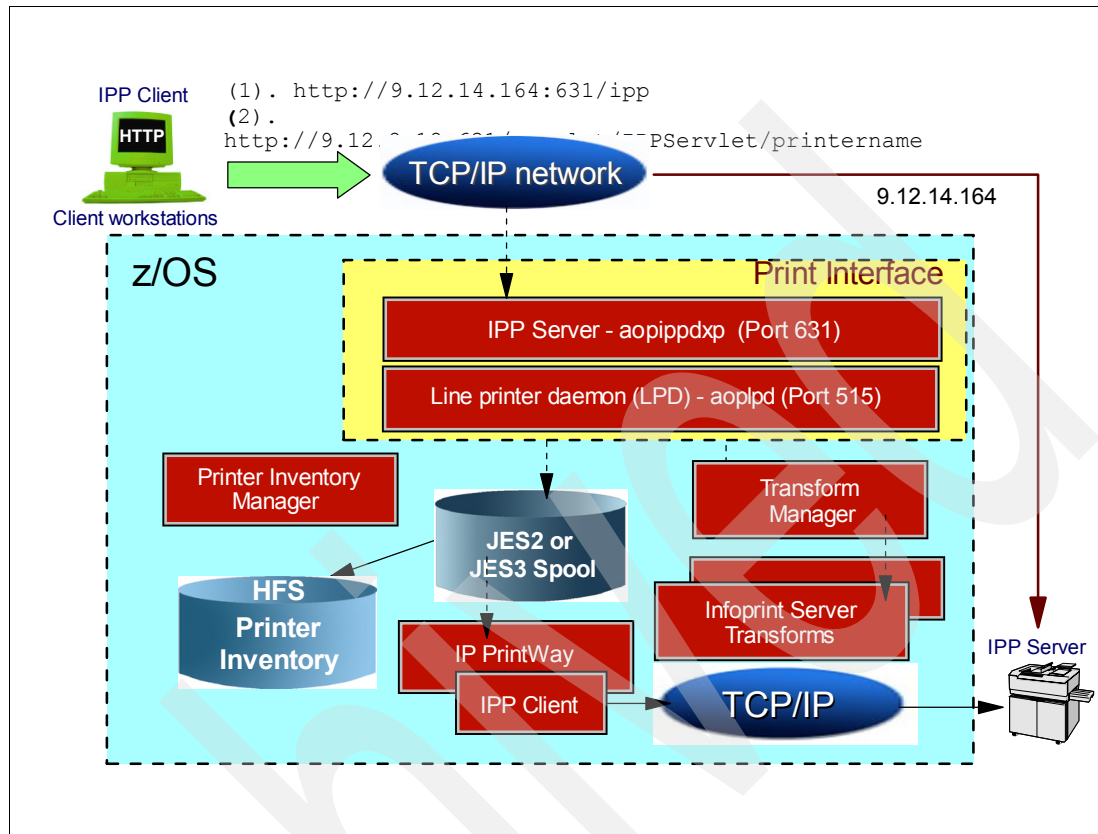


Figure 4-15 IPP processing

### IPP support

The Internet Printing Protocol (IPP) [RFC2911, RFC2910, RFC2567, RFC2567, RFC2568, RFC2569, RFC2708, RFC2910, RFC2911, RFC3995, and RFC3996] is an application level protocol that can be used for distributed printing using Internet tools and technologies.

The IPP protocol is heavily influenced by the printing model introduced in the Document Printing Application (DPA) [ISO10175] standard. There are two parts to the Protocol: (1) the encoding data and (2) the mechanism for transmitting the data between client and server:

1. A simple binary encoding has been chosen. It has a simple structure consisting of sequences of attributes. Each attribute has a name, prefixed by a name length, and a value.
2. The mechanism for transmitting the encoded data is HTTP (and associated response). No modifications to HTTP are required.

### Print Interface IPP Server

The Print Interface component of Infoprint Server provides also an Internet Printing Protocol (IPP) server. The IPP server accepts print requests from IPP clients, which typically run on workstation platforms. Print Interface processes the data and allocates an output data set on the JES spool.

Print Interface provides an Internet Printing Protocol (IPP) server, which accepts print requests from any client that uses the IPP protocol. The Print Interface IPP server supports

the IPP job attributes sent with the print requests. The IPP server can also provide job status to the IPP client. The IPP server does not, however, support canceling a print request.

### **IP PrintWay IPP Client**

IP PrintWay provides an Internet Printing Protocol (IPP) client. The IP PrintWay IPP client transmits output data sets from the JES spool over the Internet to IPP servers running either in a printer or on another host system. The IP PrintWay IPP client transmits IPP job attributes to the IPP printer.

Do not confuse the IPP client that sends print requests to the Print Interface IPP server with the IPP client that is part of the IP PrintWay component. The IPP client in IP PrintWay transmits output data sets from the JES spool to IPP servers running in a remote printer or host system.

An IPP client can specify IPP job attributes with a print request. The Print Interface IPP server supports some of the IPP job attributes that have corresponding Infoprint Server job attributes:

- ▶ Job-name
- ▶ Job-priority
- ▶ Requesting-user-name
- ▶ Copies

### **JES and IPP**

JES does not permit the IPP server to allocate data sets on the JES spool with the owner name that is specified in the IPP job attribute. Therefore, to make the owner name available in SMF records and in the data set name displayed by SDSF, the IPP server uses the owner name as the job name when allocating data sets on the JES spool.

### **Customization tasks for Print Interface IPP Server**

Required tasks:

- ▶ Ensure sufficient memory is available to start the IPP Server
- ▶ Edit the Infoprint Server configuration file (aopd.conf)
- ▶ Customize the IPP workstation client

Optional task:

- ▶ Set environment variables for the IPP Server configuration

## 4.16 Customization tasks for Print Interface IPP server

- ❑ Ensure sufficient memory is available for IPP server
  - At least 200 megabytes (MB) of memory
    - aopstart command (OMVS shell) - TSO logon panel
    - AOPSTART procedure - REGION parameter
    - Telnet, rlogin - MAXASSIZE in BPXPRMxx
- ❑ Edit the aopd.conf configuration file
  - start-daemons = {lpd ippd}
  - Default: ipp-port-number = 631
- ❑ Set environment variables for the IPP server
  - CLASSPATH - full path names of IPP files
  - JAVA\_HOME - path used to locate Java files

Figure 4-16 Customization tasks for Print Interface IPP server

### Ensuring sufficient memory is available to start the IPP server

The IPP server requires at least 200 megabytes (MB) of memory. To make sure that sufficient memory is available when you start the IPP server, do one of these, depending on how you plan to enter the **aopstart** command to start Infoprint Server:

#### ▶ ASSIZEMAX and MAXASSIZE parameters

The ASSIZEMAX parameter of the RACF ADDUSER and ALTUSER commands lets you define the maximum region size for a particular user in the OMVS segment of the user's RACF profile. The ASSIZEMAX parameter overrides the MAXASSIZE parameter set in the BPXPRMxx parmlib member.

MAXASSIZE is the maximum region size (in bytes) for an address space that was created by rlogind, telnetd, and other daemons.

Examples:

This statement in BPXPRMxx sets the maximum region size to 200 MB:

```
MAXASSIZE(209715200)    /* 200*1024*1024 = 200MB      */
```

This operator command sets the maximum region size to 200 MB:

```
SETOMVS MAXASSIZE=209715200
```

Set the hard limit (maximum) address-space-size (RLIMIT\_AS) value for a z/OS UNIX user:

```
ALTUSER userid OMVS(ASSIZEMAX(nnnn))
```

► **aopstart from the TSO/E OMVS command line**

If you enter the **aopstart** command from TSO, the SIZE parameter on the TSO/E LOGON panel determines the maximum region size for an address space.

► **Using the AOPSTART procedure**

If you use the AOPSTART procedure to enter the command, the REGION parameter on the EXEC statement in the procedure determines the maximum region size. If the REGION parameter is not specified, the default region size defined for your installation is used.

## **Editing the aopd.conf configuration file**

Add or edit these attributes in the aopd.conf configuration file:

► **ipp-port-number = 631**

The number of the port at which the IPP Server waits for print requests. Port 631 is the well-known port for communication between IPP clients and IPP Servers.

► **start-daemons = {ippd}**

To start the IPP Server daemon, add **ippd** to any other values in this attribute before using the aopstart command.

## **Setting environment variables for the IPP server**

These environment variables affect the behavior of the IPP server. If you need to change the default values of these environment variables, specify them in the aopstart EXEC.

► **CLASSPATH**

The full path names of Infoprint Server IPP files. If you installed Infoprint Server IPP files in the default directory, /usr/lpp/Printsrv, you do not need to modify this environment variable. If you installed Infoprint Server files in a different *directory*, specify these values:

- *directory*/classes/ipp.jar
- *directory*/classes/ippserver.jar
- *directory*/classes/ippreal.jar

Default:

```
/usr/lpp/Printsrv/classes/ipp.jar:  
/usr/lpp/Printsrv/classes/ippserver.jar:  
/usr/lpp/Printsrv/classes/ippreal.jar
```

► **JAVA\_HOME**

The path used to locate Java files. This environment variable is optional. However, you must set this environment variable if you did not install Java files in the default Java directories.

Default:

```
/usr/lpp/java/J1.4
```

## **Customizing the IPP workstation client**

To submit a print request over the Internet using the IPP protocol, the user's workstation must contain an IPP client. Using an IPP client, the users can print any document (not only Web documents) on any printer defined in the Printer Inventory.

The users must also specify the Uniform Resource Identifier (URI) for Infoprint Server:

```
http://host:port/printers/printername  
http://host:port/servlet/IPPServlet/printername
```

## 4.17 AFP authorization for Java libraries

- ❑ Sample job AOPJAUTH in SYS1.SAMPLIB
  - To APF-authorize Java libraries
    - Required by IPP Server
  - Requires at least READ access to the BPX.FILEATTR.APF FACILITY class profile
  - Requires superuser authority
- ❑ Java not installed in the default Java directories
  - Edit the AOPJAUTH procedure to specify the directories where Java libraries installed

Figure 4-17 APF authorizing Java libraries

### Authorizing the JAVA libraries

To use the IPP Server, you must mark Java 1.4 run-time libraries APF-authorized. To authorize libraries, you can use the sample job AOPJAUTH or the z/OS UNIX `extattr` command.

### AOPJAUTH job

Infoprint Server provides sample job AOPJAUTH in SYS1.SAMPLIB, which you can use to APF-authorize Java libraries. AOPJAUTH authorizes all Java 1.4 libraries that are installed in default Java directories on your z/OS system. Do the following:

- To run AOPJAUTH, you must have at least READ access to the BPX.FILEATTR.APF FACILITY class profile. Also, you must have superuser authority (UID of 0) or access to the BPX.SUPERUSER FACILITY class profile.
- If you did not install Java in the default Java directories, edit the AOPJAUTH procedure to specify the directories in which you installed Java libraries.
- If you edit AOPJAUTH, do not modify the PARM parameter for the AOPBATCH program on the EXEC statement. Also, because the STDIN DD statement contains executable z/OS UNIX commands, make sure that these lines do not contain sequence numbers in columns 73 - 80.

**Note:** When maintenance is applied to Java, APF authorization is lost. Therefore, you must rerun AOPJAUTH or the `extattr` command.

## Using the extattr command to authorize Java libraries

Instead of using AOPJAUTH, you can use the z/OS UNIX **extattr** command to APF-authorize Java libraries.

Authorize all libraries with an extension of .so in these Java directories:

- ▶ /usr/lpp/java/J1.4/bin
- ▶ /usr/lpp/java/J1.4/bin/classic

### Example:

```
cd /usr/lpp/java/J1.4/bin/  
extattr +a *.so  
cd classic  
extattr +a *.so
```

## 4.18 Customizing z/OS Communication Server (TCP/IP)

- ❑ Customizing z/OS Communications Server (TCP/IP)
  - Recommendation: Only Print Interface LPD on z/OS system
    - One TCP/IP stack for Print Interface LPD
    - Separate TCP/IP stack for Print Interface LPD and Communication Server LPD (LPSERVE)
- ❑ Editing the hlq.PROFILE.TCPIP data set
  - Editing the PROFILE data set
    - PORT statement - Replace LPSERVE with AOPLPD
    - ; 515 TCP LPSERVE ; LP Server
    - ; 515 TCP AOPLPD ; Infoprint LPD Server
  - z/OS SNMP agent: ports 161 and 162
- ❑ Using the LPD compatibility filter
  - LPD compatibility filter, lpd\_compat.so, for PSF printers
  - Supports for some LPD command codes and parameters that Print Interface LPD does not support

Figure 4-18 z/OS Communications Server customization

### Considerations for the Print Interface LPD

You must customize and start the IP component (TCP/IP) of z/OS Communications Server if you plan to submit print jobs to Print Interface from remote systems, use IP PrintWay to transmit print jobs to remote printers or e-mail destinations, or use the SNMP subagent.

z/OS Communications Server also provides an LPD (LPSERVE) on the z/OS system. You must decide whether to run:

- ▶ Only the Print Interface LPD
- ▶ Both the Communications Server LPD and the Print Interface LPD

**Recommendation:** IBM recommends that you run only the Print Interface LPD, without the Communications Server LPD, for these reasons:

- ▶ The Print Interface LPD provides comparable function to the Communications Server LPD. The Print Interface LPD is suitable for most clients that currently use the Communications Server LPD.
- ▶ If you run both the Print Interface LPD and the Communications Server LPD on the same system, you must configure two TCP/IP stacks so that you can use all functions that Infoprint Server provides.

## Infoprint Server ports

The default port numbers that Infoprint Server uses are:

- ▶ Print Interface LPD: port 515
- ▶ Print Interface IPP Server: port 631
- ▶ z/OS SNMP agent: ports 161 and 162

## Port 515 usage

TCP/IP ships in hlq.SEZAINST(SAMPPROF) a sample configuration file for TCP/IP. The sample profile includes a definition for port 515: `515 TCP LPSERVE;LPD Server`. If you use the sample profile for TCPIP, you must change the port definition to: `515 TCP OMVS`, comment it out, or remove it to enable Infoprint Server LPD use of port 515. See *z/OS Communications Server IP Configuration Guide*, SC31-8775 for additional considerations.

## Port 631 usage

Port 631 is the well-known port for communication between IPP clients and IPP Servers. If you specify a port other than 631, make sure that the port is not used by any other service on the z/OS system. This attribute is optional.

## Only Print Interface LPD on a z/OS system

To run only the Print Interface LPD, and not the Communications Server LPD, make the following changes in the TCP/IP profile data set definitions:

- ▶ For all functions that Infoprint Server provides, the Print Interface LPD must listen at port 515. Therefore, you must make sure that the Communications Server LPD does not reserve port 515.
- ▶ Do not start the Communications Server LPD. Remove the LPSERVE procedure name from the AUTOLOG statement in the TCPIP profile data set.
- ▶ Printer definitions in the Printer Inventory must be created with the same name as the printers that are defined with SERVICE statements in the Communications Server LPD Server configuration data set.
- ▶ If you want the Print Interface LPD to function comparably to the Communications Server LPD, specify the Print Interface LPD compatibility filter in the printer definitions for PSF-controlled printers.

## Print Interface LPD and Communications Server LPD on same z/OS

To run both the Print Interface LPD and the Communications Server LPD on the same z/OS system, you can either run each LPD on a separate TCP/IP stack or run both LPDs on the same TCP/IP stack. IBM recommends that you run each LPD on a separate TCP/IP stack so that you can use all of the functions that Infoprint Server provides.

## Running each LPD on a separate TCP/IP stack

You need to define multiple TCP/IP stacks so that the Print Interface LPD can run on one stack, while the Communications Server LPD runs on the other stack. Each TCP/IP stack has its own IP address, or its own host name that resolves to an IP address. When the Print Interface LPD and the Communications Server LPD each runs on its own stack, you can use all of the functions that Infoprint Server provides because you can assign port 515 to both the Print Interface LPD and the Communications Server LPD.



The Print Interface LPD must run on a z/OS UNIX System Services stack. You must identify the job name associated with the z/OS UNIX TCP/IP stack in the `_BPXK_SETIBMOPT_TRANSPORT` environment variable. Specify this variable before you start the Print Interface LPD.

For example, if you configure two stacks, one named TCPIP and another named TCPIPOE, and you want Infoprint Server to use the TCPIPOE stack, specify `_BPXK_SETIBMOPT_TRANSPORT=TCPIPOE`. You must set variable `_BPXK_SETIBMOPT_TRANSPORT` in the `aopstart EXEC`. If you do not specify this environment variable, the z/OS Communications Server uses its standard search mechanism to determine the TCP/IP stack to use.

### Running both LPDs on the same TCP/IP stack

The same TCP/IP stack can serve both the Print Interface LPD and the Communications Server LPD. However, you must assign different port numbers to each LPD to distinguish between them. Because you must assign port 515 to the Communications Server LPD, you must assign a port other than 515 to the Print Interface LPD. To do this, change the port number in the `lpd-port-number` attribute in the `aopd.conf` file.

### Limitations of two stacks

When you run both LPDs on the same TCP/IP stack, you *cannot use all* of the functions that Infoprint Server provides. These limitations apply if you select this TCP/IP configuration option:

- ▶ Remote users cannot use commands such as LPR and `enq` to submit print requests. This is because you cannot configure most LPR clients to send print requests to a port other than to the well-known port 515. The exception is the Infoprint Port Monitor for Windows, which allows you to configure the Port Monitor to send print requests to any port.
- ▶ IP PrintWay basic mode cannot transform data from and to the AFP data format using the resubmit for filtering option. This is because IP PrintWay basic mode resubmits data for transform to Print Interface at port 515. This limitation does not apply when you run IP PrintWay extended mode because IP PrintWay extended mode can transform data without resubmitting it to Print Interface for filtering.

### Using the LPD compatibility filter

The LPD compatibility filter, `lpd_compat.so`, provides support for some LPD command codes and parameters that the Print Interface LPD does not otherwise support. LPD command codes and parameters are specified in the LPD control file sent by the LPR client with each document to be printed.

## 4.19 Other hlq.PROFILE.TCPIP and JES considerations

- ❑ **TCPCONFIG statement**
  - **INTERVAL minutes SENDGARBAGE TRUE**
    - 10 minutes
  - **TCPRCVBFRSIZE buffersize => 32768**
  - **TCPSEENDBUFRSIZE buffersize => 32768**
- ❑ **AUTOLOG statement**
  - **When only Infoprint Server LPD is used - remove LPSERVE procedure name**
- ❑ **JES spool limits limit considerations**

Figure 4-19 Other hlq.PROFILE.TCPIP and JES considerations

### Editing the TCPIP profile data set

You can customize the TCPIP profile data set when you customize the Communications Server. The hlq.PROFILE.TCPIP data set contains statements that initialize the TCP/IP address space. Special considerations exist for Print Interface and IP PrintWay for these statements: TCPCONFIG, PORT, and AUTOLOG.

### TCPCONFIG statement

The TCPCONFIG statement in the TCPIP profile data set updates the TCP layer of TCP/IP.

```
TCPCONFIG INTERVAL 10 SENDGARBAGE TRUE TCPRCVBFRSIZE 32768 TCPSEENDBUFRSIZE 32768
```

This is a sample TCPCONFIG statement that you might want to include in the profile data set. An explanation of the parameters and considerations for both Print Interface and IP PrintWay follows.

#### ► **INTERVAL minutes SENDGARBAGE TRUE**

Number of minutes TCP waits after receiving a packet for a connection before it sends a keep-alive packet for that connection.

IP PrintWay relies on TCP to detect when a connection with the receiving system is no longer usable by sending keep-alive packets. Therefore, if you plan to use IP PrintWay, consider specifying a shorter interval than the default, which is approximately two hours.

For example, consider specifying an interval of 10 minutes. Note that the interval you specify applies to all TCP applications that direct TCP to send keep-alive packets.

Specify SENDGARBAGE TRUE if any target host requires that the keep-alive packet contain data.

► **TCPRCVBFRSIZE buffersize**

The TCP receive buffer size. IP PrintWay requires a buffer size of at least 32768.

► **TCPSENDBUFRSIZE buffersize**

The TCP send buffer size. IP PrintWay requires a buffer size of at least 32768.

You can also specify some of these parameters on other Communications Server statements. The settings from the last statement processed are used.

## **AUTOLOG statement**

The AUTOLOG statement tells TCP/IP which procedure names to start. If you want to run only the Infoprint Server LPD and not the Communications Server LPD, remove or comment out the LPSERVE procedure name from the AUTOLOG statement. Use a semicolon in the first column to indicate the line is a comment.

Example:

```
AUTOLOG
.
.
; LPSERVE                ; LPD Server
```

## **JES spool limit considerations**

Infoprint Server daemons that write data to the JES spool typically run for extended periods of time and write data to the spool for many print requests. Therefore, during normal processing, an Infoprint Server daemon can exceed the JES spool limit that you set in your installation. The spool limit is the cumulative amount of data that a program can write to the JES spool. Infoprint Server daemons that write data to the JES spool include these daemons:

- NetSpool daemon (aopnetd)
- Print Interface LPD (aoplpd)
- Print Interface IPP Server (aopipdpdxp)
- Print Interface subsystem (aopsubd)

The `_BPX_UNLIMITED_OUTPUT` environment variable lets z/OS UNIX programs that have an effective UID of 0 and run in their own address space (such as the Infoprint Server daemons) continue processing when they exceed their spool limit. In this case, the JES spool limit for each program is fixed at slightly less than 1 GB, with large spool limits for lines and AFP pages. Spool limits and JES actions specified in JES initialization statements or on the JOB JCL statement do not apply.

When the spool limit is reached, JES displays a warning message on the system console and lets the program continue. You can specify the interval at which the warning message is displayed in these JES initialization statements:

- JES2: ESTBYTES, ESTLINES, and ESTPAGES statements
- JES3: STANDARDS statement

The aopstart EXEC defines `_BPX_UNLIMITED_OUTPUT=YES` in the environment in which it starts Infoprint Server daemons. Do not remove or change the value of this environment variable.

## 4.20 Print Interface subsystem customization

- ❑ Print Interface subsystem processes output data created by z/OS batch jobs
  - Job specifies name of Print Interface subsystem in the SUBSYS parameter on the DD JCL statement
    - Default: AOP1
  - DD JCL Statement:
    - SUBSYS=(**subsystem\_name** [, 'printer\_definition\_name' ] [, 'attribute=value...' ]])
- ❑ aopd.conf configuration
  - start-daemons = { subd }
- ❑ Increasing maximum number of active transform daemons if batch jobs require transforms
  - Customizing aopxfd.conf

Figure 4-20 Customizing the Print Interface subsystem

### Customizing the Print Interface subsystem

The Print Interface subsystem can process output data created by z/OS batch jobs. To request that the Print Interface subsystem process an output data set, the request specifies the name of the Print Interface subsystem in the SUBSYS parameter on the DD JCL statement.

The Print Interface subsystem validates that a data set can print on the printer and calls the data stream transform, or other filter, requested in the printer definition. The subsystem then allocates an output data set on the JES spool using job attributes and JCL parameters specified on the DD JCL statement, the associated OUTPUT JCL statement, and values specified in the Allocation section of the printer definition. Each output data set is placed in a separate JES output group. The output data set can be printed on any printer, including printers that PSF and IP PrintWay control.

### Editing the aopd.conf configuration

Before you can use the Print Interface subsystem, you must update the start-daemons attribute in the aopd.conf configuration file. To start the Print Interface subsystem daemon, add subd to the values in this start-daemons attribute before you start the Print Interface subsystem daemon.

```
start-daemons = {subd}
```

To start the subd daemon, issue **aopstart** command in the z/OS OMVS shell or START AOPSTART operator command.

## Name of Printer Inventory

The *inventory* keyword in the *aopd.conf* file specifies the name of both the Printer Inventory and the Print Interface subsystem. The name must contain exactly four letters or numbers. The first character must be an uppercase letter. The job submitter specifies this name in the SUBSYS parameter on the DD JCL statement.

Because this name is also the name of the Printer Inventory, if you change this name after the Printer Inventory Manager has been started, you must stop and restart all Infoprint Server daemons. You must also stop and restart all other components and products that specify this Inventory name in their startup procedures. You must stop and restart the NetSpool task and IP PrintWay basic mode. If the PSF startup procedure specifies the name of the Printer Inventory, you must also stop and restart PSF.

Default: `inventory = AOP1`

## JCL parameters for the Print Interface subsystem

The SUBSYS DD JCL statement parameter can be used to request that the Print Interface subsystem process a data set. The *subsystem\_name* subparameter specifies the name of the Print Interface subsystem. This name must be the same as the Inventory name that is specified in the Infoprint Server configuration file. The Printer Inventory name is usually AOP1.

```
SUBSYS=(subsystem_name['printer_definition_name'] [, 'attribute=value...'])
```

This example shows how to submit output to the Print Interface subsystem named AOP1:

```
//DD1 DD SUBSYS=(AOP1,'myprinter')
```

If errors are made coding the SUBSYS DD statement parameter, the job will be failed by the MVS Converter/Interpreter. Error messages show what kind of mistake was found. For example:

```
IEFC746I SUBSYSTEM subsystem_name DOES NOT EXIST
AOP001E Printer 'printer_definition_name' is not defined.
```

## Increasing the maximum number of active transform daemons

The batch jobs that use the Print Interface subsystem can also use transforms. If an installation limits the number of transform daemons that can be active at one time, you might need to specify a higher number if jobs that use the Print Interface subsystem take a long time to complete or hang until they are cancelled by the operator. For example, these situations can occur:

- ▶ If you limit the number of AFP to PCL transform daemons to one, a job that writes AFP output to two data sets in the same job step will not complete if both data sets require an AFP to PCL transform daemon.
- ▶ If you limit the number of AFP to PCL transform daemons to two, two jobs that each write AFP output to two data sets in the same job step might not complete if each job reserves an AFP to PCL transform daemon.

Also, other jobs that require the AFP to PCL transform will not be able to complete.

## 4.21 Transform configuration file

- ❑ **Infoprint Server Transform Manager configuration**
    - **Default - /etc/Printsrv/aopxfd.conf**
  - ❑ **Determine maximum-active - minimum-active transforms**
    - **A value of 2 or 3 is suitable if you transform small print jobs and do not transform many print jobs at same time**
- ```
transform transform name<_transformclass>
  start-command = command      # command to start transformer
  maximum-idle-time = seconds  # amount of time before shutting down an idle transformer
  minimum-active = number      # min of this class concurrently active (default = 0)
  maximum-active = number      # max of this class concurrently active (default = no max)
  environment = {              # environment variables to be set
    variable -> value ...      # before starting transformer
  }
;
```
- ❑ **Transform daemons are started when needed idle transforms may be shut down**

Figure 4-21 Managing transform daemons

The Infoprint Server Transform Manager requires this configuration file. The Transform Manager manages transforms, which convert data streams to another format. Create an entry for each transform. If you want to specify different environment variables for the same transform for different printing situations, you can create different classes of the transform. If you define transform classes, create a separate entry for each transform class.

Each transform entry consists of a set of attributes, starting with the transform attribute and ending with a semicolon. The following shows the general format of a transform entry and defines the attributes in the transform entry that are common for all transforms:

```
transform name<_transformclass>
  start-command = command      # command to start transformer
  start-directory = directory  # directory in which to start transformer
                                # (default = current directory)
  maximum-idle-time = seconds  # amount of time allowed to pass before
                                # shutting down an idle transformer
                                # (default = no shutdown)
  minimum-active = number      # minimum of this class concurrently active
                                # (default = 0)
  maximum-active = number      # maximum of this class concurrently active
                                # (default = no maximum)
  environment = {              # environment variables to be set
    variable -> value ...      # before starting transformer
  }
;
```

<b>minimum-active</b>	<p>The number of transform daemons that the Transform Manager starts. Also the minimum number of transform daemons that the Transform Manager keeps active (that is, not shut down) even when the maximum-idle-time expires for an idle transform daemon. This attribute is optional.</p> <p>Default: minimum-active = 0</p>
<b>maximum-active</b>	<p>The maximum number of transform daemons that the Transform Manager keeps active. The number must be greater than 0 and greater than or equal to the number specified in minimum-active. When this number is reached, the Transform Manager does not start a new transform daemon to do the transform. Therefore, the transform request waits until a transform daemon is available. This attribute is optional.</p> <p>Default: No maximum number.</p>
<b>maximum-idle-time</b>	<p>The number of seconds before the Transform Manager shuts down an idle transform daemon and system resources are freed. The number you specify must be greater than 0. This attribute is optional. However, to avoid having many transform daemons active, consider specifying either this attribute or the maximum-active attribute, or both.</p> <p>Default: An idle transform daemon is not shut down.</p>

**Recommendations:** If you use the Print Interface subsystem in your installation, do not specify too low a value because jobs that use the subsystem and write multiple data sets might not be able to complete. To avoid having many transform daemons active, consider specifying either maximum-active or maximum-idle-time, or both.

Transform daemons are started when needed.

## Maximum-active considerations

Edit the maximum-active attributes in the transform configuration file, aopxfd.conf. To increase the maximum number of active transforms, consider setting a higher number. Very large print jobs might take several minutes or even an hour to transform. As a result, a transform daemon might not be available to service other transform requests. If the other transform requests are delayed long enough, the requesting programs might timeout and fail.

## Environment variables

The environment variables override environment variables with the same names that you set in the aopstart EXEC. The aopstart EXEC also picks up selected environment variables from the run-time environment.

You can specify the `_BPX_JOBNAME` environment variable in each transform entry to assign a different job name to each class of transform daemon. By default, the job name of each transform daemon is AOPXFD. When you assign a different job name to each class of transform daemon, the operator can manage the transform daemons more effectively.

## Updating transforms

If you modify a current transform, you should stop and restart the Transform Manager daemon to pick up the changes.

## 4.22 Infoprint Server Transform Manager and transforms

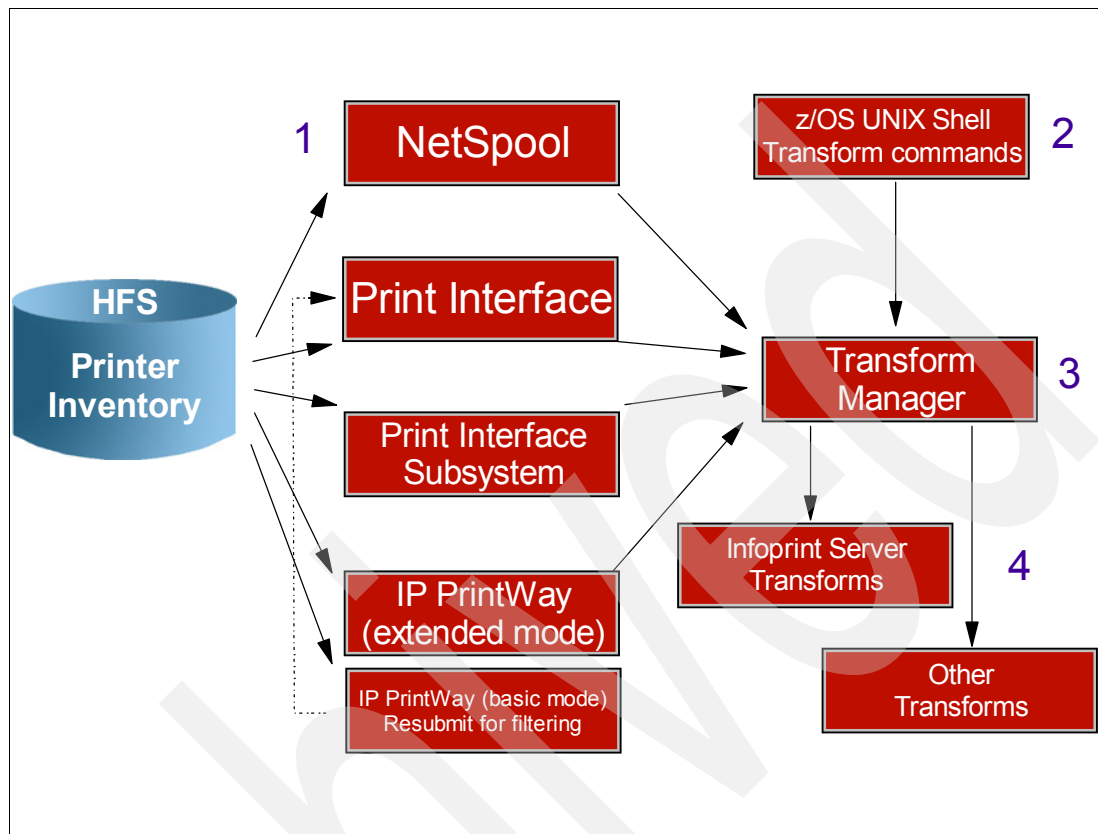


Figure 4-22 Customizing the Transform Manager and transforms

### Infoprint Server Transform Manager and transforms

The Transform Manager component of Infoprint Server controls the transform daemons provided with Infoprint Server Transforms and other optional transform products. The Transform Manager starts and stops the transform daemons using configuration information specified by the administrator in the `aopxfd.conf` file. For example, the administrator can limit the number of transform daemons that are active at a time. Transforms provided by IBM are implemented as dynamic link library (DLL) filters.

Users of transforms:

1. Before writing data to the JES spool, Print Interface, Print Interface Subsystem, NetSpool, and IP PrintWay extended mode can use transform filters that Infoprint Server Transforms and other optional transform products provide to transform print data formats. To use transforms automatically, an administrator must specify the transform filter for each data format in the printer definition. Administrators and job submitters can specify transform filter options to customize the transforms.

IP PrintWay basic mode can *resubmit* data sets to Print Interface to transform data. The administrator must specify the transform and select the **Resubmit for filtering** option in the printer definition.

2. z/OS UNIX transform commands provided with Infoprint Server Transforms and other optional transform products let users transform data format without printing it.
3. The Transform Manager manages the transform daemons and controls how many transform daemons are active at one time. The Transform Manager does not manage the



SAP to AFP transform and the Coax Printer Support feature of Infoprint Server Transforms because they are not implemented as daemons.

4. The Transform Manager can use transforms that separate, licensed IBM program products provide:
  - IBM Infoprint Server Transforms which transform data to and from the AFP data format.
  - IBM Infoprint XML Extender for z/OS, which transforms XML data to AFP or PDF format.
  - IBM Infoprint XT Extender for z/OS, which transforms Xerox files to AFP format. The Xerox files can be line-conditioned data streams (LCDS) or metacode data streams. XT is the IBM Xerox Transform technology.

### Infoprint Server Transforms features

Infoprint Server Transforms also provides the following features:

- ▶ Kanji AFP Print feature. This feature provides the Heisei Kaku Gothic W5 and Heisei Mincho W3 fonts. After these fonts are installed on the z/OS system, the PDF to AFP and the PostScript to AFP transforms can use these fonts and map some other commonly used Japanese fonts, including Ryumin-Light and Gothic BBB-Medium, to these two Heisei fonts. No additional customization is required.
- ▶ Coax Printer Support feature. This feature lets IP PrintWay print on VTAM-controlled printers.

### Using transforms

Print Interface can automatically invoke a transform before writing data to the JES spool. The administrator must specify the transform in the printer definition.

IP PrintWay basic mode can resubmit data sets to Print Interface to transform data. The administrator must specify the transform and select the Resubmit for filtering option in the printer definition.

IP PrintWay extended mode can automatically invoke a transform before sending data to the printer or e-mail destination. The administrator must specify the transform in the printer definition.

NetSpool can automatically invoke a transform after converting SCS and 3270 data to either the line or PCL data format. The administrator must specify the transform in the printer definition.

NetSpool can convert SCS and 3270 data to the line or PCL data format without using transforms that Infoprint Server Transforms and other transform products provide. The administrator simply selects either the Convert to line or Convert to PCL option in the printer definition.

Job submitters can invoke transforms directly using the z/OS UNIX transform commands, such as **pc12afp** and **ps2afp**.

## 4.23 Infoprint Server transforms

- ❑ Infoprint Transforms to AFP V2.1 for z/OS (5655-N60)
- ❑ Infoprint Transform for AFP to HP PCL V2.1 for z/OS (5655-P19)
- ❑ Infoprint Transform for AFP to Adobe PDF V2.1 for z/OS (5655-P20)
- ❑ Infoprint Transform for AFP Adobe PostScript V2.1 for z/OS (5655-P21)
- ❑ Infoprint Coaxial Printer Support V2.1 for z/OS (5655-N62)
- ❑ Infoprint Server Transforms V1.1 (5697-F51) transforms:
  - AFP to PCL PCL 5, 5e, or 5c
  - AFP to PDF PDF 1.2
  - AFP to PostScript PostScript Language Level 2
  - **No longer in service -- Continues to work**
- ❑ These transform products work with Infoprint Server. However, you can also run them as stand-alone transforms:
  - IBM Infoprint XML Extender for z/OS (5655-J66)
  - IBM Infoprint XT Extender for z/OS (5655-J65)
- ❑ Infoprint Manager for AIX (5785-E42)
- ❑ Infoprint Manager for Windows (5639-I27)

Figure 4-23 Infoprint Server Transforms

### Infoprint Transforms

Infoprint Transforms V2.1 consists of these separately priced products:

- ▶ Infoprint Transforms to AFP V2.1 for z/OS (5655-N60). It provides these transforms:
  - PCL to AFP transform converts Printer Control Language (PCL) 6 (XL, 5, 5c, 5e) data streams to AFP (MO:DCA-P) data streams.
  - PDF to AFP transform converts PDF 1.4 data streams to AFP (MO:DCA-P) data streams.
  - PostScript to AFP transform converts PostScript Language Level 3 to AFP (MO:DCA-P) data streams.
  - SAP to AFP transform converts SAP R/3 SAPGOF Release 4.6C to AFP data streams. It converts Output Text Format (OTF) data streams to MO:DCA-P data streams and Advanced Business Application Programming (ABAP) data streams to line data streams.
- ▶ Infoprint Transform for AFP to Adobe PDF V2.1 for z/OS (5655-P20).
- ▶ This transform converts AFP data streams to PDF 1.4 data streams.
- ▶ Infoprint Transform for AFP Adobe PostScript V2.1 for z/OS (5655-P21)
- ▶ This transform converts AFP data streams to PostScript Language Level 3 data streams.
- ▶ Infoprint Coaxial Printer Support V2.1 for z/OS (5655-N62)
  - This product converts line data streams to Data Stream Compatibility/Data Stream Extended (DSC/DSE) and SNA Character String (SCS) data streams and, with IP

PrintWay, transmits the data to VTAM-controlled printers defined as VTAM LU0, LU1, or LU3 printers.

## Transforms to AFP

The Transforms to AFP feature and the Kanji AFP Print feature are available at no additional charge to customers who have purchased Infoprint Server. This feature consists of these transforms that convert data streams to monochrome Mixed Object Document Content Architecture for Presentation (MO:DCA-P) data streams, which can be printed on IBM AFP printers:

- ▶ PCL to AFP Transform
- ▶ This transform converts Printer Control Language (PCL) 5 or 5e data streams to MO:DCA-P data streams.
- ▶ PDF to AFP Transform
- ▶ This transform converts Adobe Portable Data Format (PDF) 1.2 data streams to MO:DCA-P data streams. (The PDF to AFP transform supports PDF 1.3 except for Compact Font Format support.)
- ▶ PostScript to AFP Transform
- ▶ This transform converts PostScript Language Level 3 data streams to MO:DCA-P data streams.
- ▶ SAP to AFP Transform
- ▶ This transform converts SAP R/3 SAPGOF Release 4.6C (and lower releases) data streams. It converts Output Text Format (OTF) data streams to MO:DCA-P data streams and Advanced Business Application Programming (ABAP) data streams to line data streams.
- ▶ Each transform provides:
- ▶ A z/OS UNIX command which you can use to transform data in a z/OS UNIX file or MVS data set without printing it. The z/OS UNIX transform command creates an output file, which you can later print or transmit to another system for viewing or printing. The z/OS UNIX commands are:

**afp2pcl, afp2pdf, afp2ps, pc12afp, ps2afp, pdf2afp, and sap2afp.**

A filter that lets Infoprint Server automatically transform data before sending it to the printer or e-mail destination.

## Other Infoprint transforms

IBM provides other Infoprint transform products that can work with Infoprint Server but can also work as stand-alone products. These separately priced transform products are:

- ▶ Infoprint XML Extender for z/OS (5655-J66)  
These transforms convert XML data streams to AFP or PDF data streams.
- ▶ Infoprint XT Extender for z/OS (5655-J65)  
This transform converts Xerox files to AFP data streams.

**Note:** Infoprint Server Transforms V1.1 for z/OS (5697-F51) is no longer in service. Although Infoprint Server continues to work with Infoprint Server Transforms V1.1, you should use the replacement transform products instead.

## 4.24 Customizing the Transform Manager

- ❑ Update the BPXPRMxx member of SYS1.PARMLIB
  - Transform Manager uses z/OS UNIX domain sockets
  - FILESYSTYPE TYPE(UDS) ENTRYPOINT(BPXTUINT)
- ❑ Create the transform configuration file (aopxfd.conf)
  - /usr/lpp/Printsrv/samples/aopxfd.conf -> /etc/Printsrv/aopxfd.conf
  - SYS1.SAMPLIB(AOPCPETC)
- ❑ Create transform configuration file (aoprxf.conf) - Optional
  - remote-transform-manager-ip-address = hostname  
[remote-transform-manager-port-number = portnumber]
- ❑ Edit the Infoprint Server configuration file (aopd.conf)
  - start-daemons = { xfd }
- ❑ Set environment variables for the Transform Manager
  - AOPXFD\_CONF and AOPRXF\_CONF
  - LANG, LIBPATH, MANPATH, NLSPATH, and PATH

Figure 4-24 Customizing the Transform Manager

### Customizing the Transform Manager

The following data format transforms are supported by the Infoprint Server transforms:

Transform	Daemon name	Managed
PCL to AFP	pcl2afpd	Yes
PostScript to AFP	ps2afpd	Yes
PDF to AFP	ps2afpd	Yes
AFP to PCL	afp2pcl	Yes
AFP to PDF	afp2pdfd	Yes
AFP to PostScript	afp2psd	Yes
XML to AFP	xml2afpd	Yes
XML to PDF	xml2pdfd	Yes
Xerox to AFP	x2afpd	Yes
SAP OTF and SAP ABAP to AFP		No

Infoprint Server lets you transform data formats remotely using Infoprint Transform Manager for Linux (5639-P51) transforms. You can use the new **remotexf** command to transform files remotely. Infoprint Transform Manager for Linux transforms convert files to AFP format from:

- ▶ Adobe Portable Document Format (PDF) 1.5
- ▶ Adobe PostScript Language Level 3
- ▶ Graphics interchange format (GIF)
- ▶ HP Printer Control Language (PCL) 6

- Joint Photographic Experts Group (JPEG) file information format (JFIF)
- Tagged image file format (TIFF)

### Updating the BPXPRMxx member of SYS1.PARMLIB

Several Infoprint Server components, including the Transform Manager, use z/OS UNIX domain sockets. Therefore, you must have the FILESYSTYPE TYPE(UDS) statement in the BPXPRMxx member of SYS1.PARMLIB. This is an example of the FILESYSTYPE TUPE(UDS) statement:

```
FILESYSTYPE TYPE(UDS) ENTRYPPOINT(BPXTUINT)
      NETWORK DOMAINNAME(AF_UNIX)
              DOMAINNUMBER(1)
              MAXSOCKETS(200)
              TYPE(UDS)
```

You might need to increase the maximum number of allowed sockets specified in the BPXPRMxx member of SYS1.PARMLIB. Specify at least 200 in the MAXSOCKETS parameter on the NETWORK statement for AF\_UNIX file systems. Although a MAXSOCKETS value of 200 is suitable for most installations, you might need to specify a MAXSOCKETS value of 400 to make sure that UNIX sockets are always available when needed.

### Creating the transform configuration file (aopxfd.conf)

The transform configuration file contains information that the Transform Manager uses to manage transform daemons. You must create this file before you start the Transform Manager.

You do not need to create this file in order to use Infoprint Transform Manager for Linux or Infoprint Manager for AIX/Windows transforms.

In the transform configuration file, you can specify options and environment variables that affect the transform. These options and environment variables apply whether Print Interface invokes the transform automatically or whether the user invokes the transform with a transform command.

To create and edit the transform configuration file you must be running with effective UID of 0.

- If you change the configuration file while the Transform Manager is running, restart the Transform Manager to pick up the changes:

```
aopstop -d xfd
aopstart
```

The sample configuration file is in /usr/lpp/Printsrv/samples/aopxfd.conf.

### Creating the remote transform configuration file (aoprxf.conf)

The remote transform configuration file contains information that Transform Interface uses to access Infoprint Transform Manager for Linux (5639-P51). If you use Infoprint Transform Manager for Linux, the aoprxf.conf configuration file is required.

The remote transform configuration file, aoprxf.conf, format:

```
[ #comment ]
remote-transform-manager-ip-address = hostname
[ remote-transform-manager-port-number = portnumber ]
```

**remote-transform-manager-ip-address = hostname**

The host name or dotted-decimal IP address of the Infoprint Transform Manager for Linux transform server. This attribute is required.

Default: None

**remote-transform-manager-port-number = portnumber**

The port number of the Infoprint Transform Manager for Linux transform server. This attribute is optional.

Default: remote-transform-manager-port-number = 6986

The sample configuration file is in /usr/lpp/Printsrv/samples/aoprxf.conf.

### **Editing the Infoprint Server configuration file (aopd.conf)**

To start the Transform Manager, you must add or edit the start-daemons attribute in the Infoprint Server configuration file, aopd.conf:

- ▶ start-daemons = { xfd }

Add xfd to the values in this attribute before using the **aopstart** command.

If you change this attribute while Infoprint Server is running, stop and restart all Infoprint Server daemons to pick up the change. Also, restart the z/OS HTTP Server if you run Infoprint Central to pick up the change.

### **Setting environment variables for the Transform Manager**

The AOPXFD\_CONF environment variable affects the behavior of the Transform Manager. If you need to set this environment variable, set it in:

- ▶ The aopstart EXEC
- ▶ The /etc/profile file

The Transform Manager environment variables are:

- ▶ **AOPRXF\_CONF**

The full path name of the remote transform configuration file, aoprxf.conf. This environment variable is optional. If you did not create this configuration file, or if you created it in the default location, you do not need to set it.

Default: /etc/Printsrv/aoprxf.conf

- ▶ **AOPXFD\_CONF**

Full path name of the transform configuration file. If the transform configuration file is in /etc/Printsrv/aopxfd.conf, you do not need to set this environment variable.

Default: /etc/Printsrv/aopxfd.conf

- ▶ Transform Interface also uses the LANG, LIBPATH, MANPATH, NLSPATH, and PATH environment variables.

## 4.25 Transform classes

- ❑ Each transform can have one or more **transform classes** in the **aofxfd.conf** file - (optional)
  - Transform classes can specify different sets of options and environment variables for one transform
    - `transform transformname[ _transformclass ]`
  - Example: transform `pcl2afp_letter_300`
  - To specify the **letter\_300** transform class on the `pcl2afp` command, type:
    - `pcl2afp -c letter_300`
- ❑ filter program - A filter is a program that modifies the input data before it is sent to the printer

Figure 4-25 Transform classes

### Transform classes

For each transform, you can define one or more transform classes in the transform configuration file (optional). Transform classes let you transform data with different transform options and environment variables. For each transform class, you can specify a different set of options and environment variables.

For example, if you want to transform PCL data to AFP format for printers that have different resolutions or paper sizes, you would define a transform class for each combination of printer resolution and paper size and specify the resolution and paper size in environment variables that are specific for the transform class. The sample transform configuration file, `/usr/lpp/Printsrv/samples/aopxfd.conf`, defines several sample transform classes.

To use a transform class, the administrator or the job submitter must specify the transform class:

- ▶ The administrator specifies the transform class in the printer definition in the `-c filter` option. For example, in the Filter field for the PCL data format, specify:  
`pcl2afp.d11 -c letter_300`
- ▶ The job submitter specifies the transform class on the z/OS UNIX transform command in the `-c` option. For example:  
`pcl2afp -c letter_300 -o myfile.afp myfile.pcl`

- The job submitter specifies the transform class in the filter-options job attribute on the **lp** command or AOPPPRINT procedure. For example:

```
lp -d myprinter -o "filter-options='-c letter_300'" myfile.pcl
```

## Filter programs

In Infoprint Server, a filter is a program that can add, delete, or modify input data. Infoprint Server provides support for two types of filter programs:

- **DLL filters:** A DLL filter is one that resides in a dynamic link library (DLL) and that can inspect input data and perform various functions, such as convert data formats, add header pages, and specify job attributes.
- **UNIX filters:** In UNIX operating systems, a program that obtains data from standard input (STDIN) and returns the results to standard output (STDOUT).

Some transforms provide a DLL filter program that converts the data. A filter is a program that modifies the input data before it is sent to the printer. In a printer definition, you can associate a filter with an input data format (line data, MO:DCA-P, PCL, PDF, PostScript, SAP, XML, text, or other).

When you associate a filter with a data format, Print Interface, NetSpool, or IP PrintWay extended mode automatically calls the associated filter when the document to be printed contains that data format. IP PrintWay basic mode can also call an associated filter using the resubmit for filtering function.

## Format of a transform entry

Each transform entry in the aopxfd.conf file consists of a set of attributes, starting with the transform attribute and ending with a semicolon.

```
# comment
transform transformname[_transformclass]
  start-command = "daemon [ option ]..."
  [ environment = {name -> value [ name -> value ]...} ]
  [ maximum-active = number ]
  [ maximum-idle-time = seconds ]
  [ minimum-active = number ]
;
```

### transformname

The name of the transform, for example, pcl2afp or ps2afp.

### transformclass

The name of a transform class. Specify from 1 to 63 characters, including letters, numbers, or special characters.

**Default:** If you omit the transform class from the transform entry, this transform entry is used when no transform class option (-c) is specified either in the printer definition or by the job submitter.



## 4.26 Format of a PCL to AFP transform entry

**NOTE:** Currently, each instance of the transform can output one paper size and resolution, so it is necessary to configure a transformer class for each combination.

**ENVIRONMENT VARIABLES:**

<b>AOP_PAGE_HEIGHT</b>	page height given in inches (i) or millimeters (m)	<b>Default: 11i</b>
<b>AOP_PAGE_WIDTH</b>	page width given in inches (i) or millimeters (m)	<b>Default: 8.5i</b>
<b>AOP_RESOLUTION</b>	output resolution given in pixels	<b>Default: 240</b>
<b>AOP_HORIZONTAL_MARGINS</b>	left/right margins given in inches (i) or millimeters (m).	<b>Default:0.167</b>
<b>AOP_VERTICAL_MARGINS</b>	top/bottom margins given in inches (i) or millimeters (m).	<b>Default:0.167</b>

**EXAMPLE:**

```
transform pcl2afp
  start-command = pcl2afpd
  minimum-active = 1
  maximum-active = 3
  maximum-idle-time = 1200 # 1200 seconds (20
minutes)
  environment = { _BPX_JOBNAME -> PCL2AFPD
    AOP_PAGE_HEIGHT -> 11i
    AOP_PAGE_WIDTH -> 8.5i
    AOP_RESOLUTION -> 240
    AOP_HORIZONTAL_MARGINS -> 0.167i
    AOP_VERTICAL_MARGINS -> 0.167i }
;
```

Figure 4-26 Format of a PCL to AFP transform entry

### Format of a PCL to AFP transform entry

In the **aopxfd.conf** file, create one transform entry for the PCL to AFP transform. If you want to define transform classes, create a separate transform entry for each transform class.

```
transform pcl2afp[_transformclass]
  start-command = "daemon [ option ]..."
  [ environment = {name -> value [ name -> value]... } ]
  [ maximum-idle-time = seconds ]
  [ minimum-active = number ]
  [ maximum-active = number ]
;
```

**start-command = "daemon [ option ]..."**

This attribute names the transform daemon and option. Enclose the value in single or double quotation marks if you specify the option.

```
start-command = "pcl2afpd [-m nnn {K|M}]"
```

**pcl2afpd** The name of the PCL to AFP transform daemon. If the transform daemon is not in a directory identified in the PATH environment variable, specify the full directory path name of the daemon.

**-m nnn{K | M}** The number of bytes of memory the transform daemon can use to do transforms. Specify the number of bytes in either kilobytes or megabytes. The

amount required depends on the compression type, the image being compressed, fonts, and so on.

**daemon** The name of the transform daemon, for example, pcl2afpd or ps2afpd. If the transform daemon is not in a directory identified in the PATH environment variable, then specify the full directory path name of the daemon.

**option** One or more options supported by the transform daemon.

**environment = {name -> value [ name -> value ]... }**

Specify environment variables that define the transform environment. Enclose the environment variables in braces. The values in these environment variables override values of environment variables with the same names that were set when the aopstart command was used.

Example:

```
environment = {AOP_RESOURCE_PATH -> /usr/lpp/Printsrv/ps2afp}
```

You can define the **\_BPX\_JOBNAME** environment variable in each transform entry to assign a different job name to each class of transform daemon. By default, the job name is the user ID of the user who starts Infoprint Server. Assigning a different job name to each class of transform daemon lets the operator manage the transform daemons more effectively. You can specify a job name of 1 to 8 alphanumeric characters. Incorrect job names are ignored.

## 4.27 PostScript and PDF to AFP transform

### NOTE:

An initialization file is required. The start-command line option "-i" is used to specify the initialization file. The default initialization file (ps2afpe.ps.bin) is installed in the transform resources directory.

### ENVIRONMENT VARIABLES:

**AOP\_RESOURCE\_PATH** - Directories containing the transform resources. If the transform resources are in more than one directory, list all directories separated by colons; the directories are searched in the order they are listed. You do not need to list the default location for transform resources files, /usr/lpp/Printsrv/ps2afp because the default directory is always searched last.

DEFAULT: /usr/lpp/Printsrv/ps2afp

### EXAMPLE:

This example illustrates the use of AOP\_RESOURCE\_PATH env variable to specify a directory containing configurable resources, e.g. UserInit and preload.ps (see "Infoprint Transforms to AFP for z/OS" G550-0443-01 for info on PostScript/PDF-to-AFP transform configuration).

```
transform ps2afp
start-command = "ps2afpd -i ps2afpe.ps.bin"
minimum-active = 1
maximum-active = 3
maximum-idle-time = 1200 # 1200 seconds (20 minutes)
environment = {
  AOP_RESOURCE_PATH -> /usr/lpp/Printsrv/ps2afpv2/lib:/usr/lpp/Printsrv/ps2afpv2/Resource:
  /usr/lpp/Printsrv/ps2afpv2/fonts }
;
```

Figure 4-27 PostScript to AFP and PDF to AFP transform

## PostScript to AFP and PDF to AFP transform

Format of a PostScript and PDF to AFP transform entry

```
transform ps2afp[_transformclass]
start-command = "ps2apfd -i initializationfile [-M nnnnn{K|M}]"
[ environment = {name -> value[ name -> value]... } ]
[ maximum-idle-time = seconds ]
[ minimum-active = number ]
[ maximum-active = number ]
;
```

If you want Print Interface to automatically call the PostScript to AFP transform or the PDF to AFP transform before writing data to the JES spool, the administrator must specify the transform DLL in the Filter field of printer definitions in the Printer Inventory. In the printer definition, the administrator can also specify transform options, such as the type of AFP images to create; the resolution, length, and width of the output images; and the transform class to use. The user can specify the same transform options (1) on the ps2afp and pdf2afp commands and (2) in the filter-options job attribute when submitting a print request.

PostScript data is usually in ASCII representation. However, the PostScript to AFP transform can also process PostScript data in EBCDIC representation. Print Interface converts an EBCDIC data stream that starts with %! (X'6C5A') from EBCDIC code page IBM-1047 to ASCII code page ISO8859-1. You cannot customize these code pages.

## Ensuring sufficient memory is available to start the transform

The region size must be at least 10M larger than the number of bytes specified in the -M option in the transform configuration file. The default for the -M option is 32M. Therefore, the minimum region size is 42MB. However, IBM recommends a region size of 256 MB or more so that you can transform large or complex data streams.

**Note:** The Infoprint Transforms to AFP for z/OS transform (5655-N60) does not use the PostScript initialization file UserInit and preload.ps

Environment variables for the PDF to AFP and PostScript to AFP transforms:

► **\_BPX\_JOBNAME**

The job name for this transform.

► **\_CEE\_DMPTARG**

The directory where Language Environment (LE) writes a CEEDUMP. The transform's current working directory is base-directory/xfd/ps2afp.#.d.

► **AOP\_FAIL\_ON\_ERROR**

Specifies whether the transform stops processing when an error occurs during the transform.

► **AOP\_FONT\_SUBSTITUTION\_MESSAGES**

Indicates whether the transform writes a message (AOP2500W) in the transform's stderr file when it substitutes fonts in a document if no other errors occurred.

► **AOP\_RESOURCE\_PATH**

The directories that contains fonts and other transform resources. If the fonts and other resources are in more than one directory, list all directories separated by a colon.  
Default: AOP\_RESOURCE\_PATH -> /usr/lpp/Printsrv/ps2afpv2/lib:

/usr/lpp/Printsrv/ps2afpv2/Resource:/usr/lpp/Printsrv/ps2afpv2/fonts

## Setting up security

Security checking done in the transform requires that the user identifier (UID) of the executable file for the transform, ps2afpd, not be 0 (zero). When it is installed, file ps2afpd has a UID of 0. Therefore, you must change the owner of the file. The new owner must have a UID that is not 0 and not the default UID. Also, the set-user-ID flag for the file must be turned on.

## 4.28 Security for the PostScript and PDF to AFP transform

- ❑ UID of the executable ps2afpd file must not be 0 (zero)
  - Define a group to RACF
    - ADDGROUP (NOGROUP) OMVS(GID(nogroup-gid))
  - Define a user to RACF as a z/OS UNIX user
    - ADDUSER (NOBODY) OMVS(UID(nobody-non-zero-uid))
  - Connect the user to the group.
    - CONNECT (NOBODY) GROUP(NOGROUP)
  - Switch to an effective UID of 0
  - Assign the user as the owner of file ps2afpd
    - chown NOBODY /usr/lpp/Printsrv/bin/ps2afpd
  - Turn the set-user-ID flag on for file ps2afpd
    - chmod u+s /usr/lpp/Printsrv/bin/ps2afpd
  - Verify the result
    - ls -l /usr/lpp/Printsrv/bin/ps2afpd
    - -rwsr-x--- 1 NOBODY ... ..

Figure 4-28 Security for the PostScript and PDF to AFP transform

### Security for the PostScript and PDF to AFP transform

Security checking done in the transform requires that the user identifier (UID) of the executable file for the PostScript and PDF to AFP transform, ps2afpd, not be 0 (zero). When it is installed, ps2afpd has a UID of 0. Therefore, you must change the owner of the file. The new owner must have a UID that is not 0 and not the default UID. Also, the set-user-ID flag for the file must be turned on.

First use the Resource Access Control Facility (RACF), or another program that follows system authorization facility (SAF) protocol, to create a user and group profile for the owner of ps2afpd. Then, change the owner of ps2afpd and turn on the set-user-ID flag.

#### Define a group to RACF

The group profile must have an OMVS segment and a group identifier (GID). You can use any group name. IBM recommends that you do not give this group any authority to the z/OS file system.

For example, this RACF command defines group NOGROUP. For nogroup-gid, specify an integer that is different from other GIDs in your installation:

```
ADDGROUP (NOGROUP) OMVS(GID(nogroup-gid))
```

If you have already done this task for another transform, you might have already defined group NOGROUP. You do not need to define it again.

## Define a user to RACF as a z/OS UNIX user

This user will be the owner of ps2afpd. The transform, as well as PostScript jobs being transformed, run with the UID of this user.

The user profile must have an OMVS segment. Its **UID must (1) not be 0 and (2) not be the default UID, which is defined in the BPX.DEFAULT.USER profile in the RACF FACILITY class**. You can use any user name. For example, you can use NOBODY.

IBM recommends that you do not give this user authority to the z/OS file system. However, if any PostScript jobs to be transformed require access to certain files, you can give this user access to the required files or connect this user to another group (or groups) that has access to the required files.

For example, this RACF command defines user NOBODY. For nobody-uid, specify an integer that is different from other UIDs in your installation:

```
ADDUSER (NOBODY) OMVS(UID(nobody-non-zero-uid))
```

If you have already done this task for another transform, you might have already defined user NOBODY. You do not need to define it again.

## Connect the user to the group

Connect the user defined in step 2 to the group defined in step 1. For example, this RACF command connects user NOBODY to the NOGROUP group:

```
CONNECT (NOBODY) GROUP(NOGROUP)
```

## Switch to an effective UID of 0

To use the **su** command, you must be permitted to the BPX.SUPERUSER profile in the FACILITY class in RACF. This only has to be done if you are not a superuser as you need this to be able to issue the **chown** command.

### ► Assign the user as the owner of file ps2afpd

Use the **chown** command to assign the user defined in step 2 as the owner of the executable file ps2afpd. Issue this command on the z/OS UNIX command line to assign user NOBODY as the owner:

```
chown NOBODY /usr/lpp/Printsrv/bin/ps2afpd
```

### ► Turn the setuid flag on for file ps2afpd

The **chown** command turns off the setuid flag. Therefore, use the **chmod** command to turn this flag on again. For example, type this command on the z/OS UNIX command line:

```
chmod u+s /usr/lpp/Printsrv/bin/ps2afpd
```

After doing these steps, you can use the **ls** command to list the owner of ps2afpd and to verify that the setuid flag is on. For example, type on the z/OS UNIX command line:

```
ls -l /usr/lpp/Printsrv/bin/ps2afpd
```

Assuming that you assigned user NOBODY as the owner of ps2afpd, output from the **ls** command should look like this:

```
-rwsr-x--- 1 NOBODY ... ..
```

## 4.29 AFP to PCL, AFP to PDF, and AFP to PostScript transforms

- ☐ Edit the transform configuration file (aopxfd.conf)
  - Required
- ☐ Set up security for AFP resource libraries
  - Required
- ☐ Add paper sizes
  - Optional
- ☐ Customize the font-mapping table
  - Optional
- ☐ Scale 240-pel to 300-pel fonts using the AOXCF30 program
  - Optional

Figure 4-29 Customizing AFP to PCL, AFP to PDF, and AFP to PostScript transforms

### Editing the transform configuration file (aopxfd.conf)

The transform configuration file contains information that the Infoprint Server Transform Manager uses to manage the *from* AFP transforms. If this configuration file does not already exist, create it before you start the Transform Manager. If you change the configuration file while the Transform Manager is started, stop and restart the Transform Manager to pick up the changes.

### Setting up security for AFP resource libraries

The user ID that starts Infoprint Server must have RACF READ access to all system AFP resource libraries and user AFP resource libraries that are specified for the transforms. Job submitters must have RACF READ access to user AFP resource libraries.

- ▶ System AFP resource libraries are libraries specified in the transform configuration file.
- ▶ User AFP resource libraries are libraries specified in the (1) Resource library field in the printer definition, (2) USERLIB parameter of the OUTPUT JCL statement, or (3) resource-library job attribute.

**Recommendation:** To limit access to AFP resources, use the AOPSTART JCL procedure, instead of the aopstart command, to start Infoprint Server because you can associate a specific user ID with the AOPSTART JCL procedure. Then that user ID can be the sole member of the AOOPER group.

Job submitters must use a job submission method that authenticates their z/OS user IDs. Job submission methods that can authenticate z/OS user IDs include:

- ▶ `lp`, `afp2pcl`, `afp2ps`, and `afp2pdf` commands
- ▶ SAP R/3, using the Infoprint Server SAP Output Management System (OMS)
- ▶ Windows SMB protocol
- ▶ AOPPRINT and AOPBATCH JCL procedures
- ▶ z/OS JCL that submits a print job to the Infoprint Server subsystem or to IP PrintWay extended mode

Job submitters who use other job submission methods, such as the Infoprint Port Monitor for Windows, can only use user AFP resource libraries that have universal READ access. Infoprint Server writes a message (AOP092E) to the common message log and job submitter (if the job submission method allows messages to be returned) when the user does not have READ access to the AFP resource libraries. However, Infoprint Server suppresses RACF messages related to failed access checks unless you request RACF notification by specifying the RACF NOTIFY parameter (on the RDEFINE or RALTER command) for the profiles that protect the AFP resources.

### Adding paper sizes

You can add a new paper size to the paper sizes that the transforms support. To add a new paper size, you must add an entry with the name and dimensions of the new paper size to the AOPPAPER table. The source code for the AOPPAPER table is in the AOXPAPER member of SYS1.SAMPLIB data set.

To add a new paper size, code the PAPERGEN macro in AOXPAPER. In the PAPERGEN macro, you must specify:

- ▶ The name of the paper entry. The name can contain 1 to 8 letters or numbers.
- ▶ The width and length of the physical page in 300 d.
- ▶ The origin (x and y coordinates) of the printable area in 300 dpi.
- ▶ The width and length of the printable area in 300 dpi. The printable area is typically smaller than the size of the physical medium because many printers cannot print edge-to-edge.
- ▶ The origin (x and y coordinates) of the logical page. Only the AFP to PCL transform uses this value.
- ▶ The width and length of the logical page. Typically, the width of the logical page is equal to the width of the physical page minus 71 dots (at 300 dpi) at each end for European page formats or minus 75 dots for US page formats, and the length of the logical page is equal to the length of the physical page. Only the AFP to PCL transform uses this value.
- ▶ The PCL paper type ID. See your printer manufacturer's documentation. Only the AFP to PCL transform uses this value.

Assemble and link the modified AOXPAPER module, and install the AOPPAPER table using SMP/E. IBM provides these sample usermods to install the AOPPAPER table:

- ▶ SYS1.SAMPLIB(AOXPCLPJ) - AFP to PCL
- ▶ SYS1.SAMPLIB(AOXPDPFJ) - AFP to PDF
- ▶ SYS1.SAMPLIB(AOXPSPJ) - AFP to PostScript

Specify the new paper name in the AOP\_PAPER environment variable in the transform configuration file and restart the Transform Manager.



## Customizing the font-mapping table

All three transforms use the same default internal table to do the font-mapping function. This font-mapping table supports all single-byte character sets included in the IBM AFP Font Collection V2 (program number 5648-B33), which are available in both raster and outline formats. The sample font-mapping table provided in SYS1.SAMPLIB(AOXFONTS) is the same as the default internal font-mapping table.

The default font-mapping table is suitable for most installations.

*Outline fonts:* In single- and double-byte outline fonts the character shapes are represented by mathematical expressions. Because the font shape is defined without regard to size, outline fonts are scalable; therefore, you only have to store one version of an outline font. This increases your system storage space and enhances printing performance.

*Raster fonts:* A raster font is a font created by a series of pels arranged to form an image. Raster fonts are created in a specific point size, so if you want to use different sizes of the same raster font, you have to store multiple versions of the same font. Raster fonts can have 240-pel or 300-pel formats:

- ▶ 240-pel fonts 240-pel raster fonts can be bounded-box or unbounded-box. All IBM AFP printers except the 3800-3 use bounded-box fonts.
- ▶ 300-pel fonts Many IBM AFP printers print documents at 300-pel resolution. 300-pel fonts are provided in the optional Compatibility Fonts feature of PSF, or in the AFP Font Collection, Program Number 5648-B33. Your system programmer can also convert any single-byte 240-pel font in the bounded-box format to a 300-pel font by using the font-conversion program distributed with PSF. This program, APSRCF30, is described in PSF for z/OS: Customization.

## Scaling 240-pel to 300-pel fonts using the AOXCF30 program

The AOXCF30 font-conversion program provided with Infoprint Server Transforms lets you scale (that is, convert) your single-byte, bounded-box, 240-pel fonts to 300-pel fonts. This program places the scaled fonts into a partitioned data set, which you can then use with the transform. You must scale 240-pel fonts to 300-pel fonts if you do not already have 300-pel fonts and either of these conditions apply:

- ▶ You use the AFP to PCL transform. This transform requires 300-pel fonts.
- ▶ You use the AFP to PostScript or AFP to PDF transform, documents to be transformed reference raster fonts, and you do not map raster fonts to outline fonts.

## 4.30 AFP to PCL, AFP to PostScript, or AFP to PDF transform entry

- ❑ Create a separate transform entry for:
  - AFP to PCL transform
  - AFP to PDF transform
  - AFP to PostScript transform
- ❑ Create a separate transform entry for each transform class
- ❑ Transform entry format:

```
transform afp2xxx[_transformclass]
start-command = afp2xxxd
[ environment = {name -> value [ name -> valueâ... } ]
[ maximum-idle-time = seconds ]
[ minimum-active = number ]
[ maximum-active = number ]
```
- ❑ Transform example:
  - transform afp2pcl\_us
- ❑ Using this transform class with afp2pcl command, type:
  - afp2pcl -c us

Figure 4-30 AFP to PCL, AFP to PostScript, or AFP to PDF transform entry

### AFP to PCL, AFP to PostScript, or AFP to PDF transform entry

A separate transform entry for each of the transforms and transform classes that you plan to use must be created in the aopxpd.conf file for:

- ▶ AFP to PCL transform
- ▶ AFP to PDF transform
- ▶ AFP to PostScript transform.

Format of the transform entry:

```
transform afp2xxx[_transformclass]
start-command = afp2xxxd
[ environment = {name -> value [ name -> valueâ... } ]
[ maximum-idle-time = seconds ]
[ minimum-active = number ]
[ maximum-active = number ]
;
```

The entry provides the name of the transform and, as an option, the name of the transform class.

**transform** Indicates the beginning of a transform entry.

- afp2xxx** The name of the transform that Infoprint Server Transforms provides. Valid values are:
- afp2pcl: The AFP to PCL transform
  - afp2pdf: The AFP to PDF transform
  - afp2ps: The AFP to PostScript transform
- transformclass** The name of an optional transform class. Specify from 1 to 63 characters, including letters, numbers, or special characters.
- Default: No transform class is defined.
- start-command** afp2xxxd is the name of the transform daemon. If the transform daemon is not in a directory identified in the PATH environment variable, specify the full directory path name of the daemon. This attribute is required. Valid values are:
- afp2pcld: The AFP to PCL transform daemon
  - afp2pdfd: The AFP to PDF transform daemon
  - afp2psd: The AFP to PostScript transform daemon
- environment** Environment variables that define the transform environment for the transform. Enclose the environment variables in braces. The values in these environment variables override environment variables with the same name that were set when the **aopstart** command was used. Enclose the value in single or double quotation marks if the value contains special characters or spaces.

**Transform class example:**

```
transform afp2pcl_us
```

To use this transform class on the **afp2pcl** command, type:

```
afp2pcl -c us
```

## 4.31 AFP to PCL, AFP to PostScript, or AFP to PDF environment variables

<b>AOP_TRAYID</b>	<b>Specifies the tray ID for PCL or PostScript printer</b>
<b>AOP_PAPER</b>	<b>Specifies the paper size for each AFP input tray number</b>
<b>AOP_CHARS</b>	<b>The default coded font</b>
<b>AOP_ENCRYPT</b>	<b>Indicates whether you want to encrypt documents</b>
<b>AOP_FORMDEF</b>	<b>Default form definition used to format the input data</b>
<b>AOP_PAGEDEF</b>	<b>Default page definition when not specified elsewhere</b>
<b>AOP_FONTLIB</b>	<b>AFP system resource libraries that contain fonts</b>
<b>AOP_FORMDEFLIB</b>	<b>AFP system resource libraries that contain form definitions</b>
<b>AOP_OVERLAYLIB</b>	<b>AFP system resource libraries that contain overlays</b>
<b>AOP_PAGEDEFLIB</b>	<b>AFP system resource libraries that contain page definitions</b>
<b>AOP_PAGESEGLIB</b>	<b>AFP system resource libraries that contain page segments</b>
<b>AOP_FONTMAP</b>	<b>Indicates whether the transform maps fonts</b>
<b>AOP_OUTLINES</b>	<b>AFP to PDF transform processing for outline fonts</b>
<b>AOP_ANNOTATIONS</b>	<b>AFP to PDF transform produces annotations in the PDF file</b>
<b>AOP_COLOR</b>	<b>Indicates whether the transform is to produce color output</b>
<b>AOP_CUTSHEET</b>	<b>Transform is to prepare the output for cutsheet printer</b>
<b>AOP_FLATE</b>	<b>AFP to PDF transform is to use the Adobe Flate compression</b>
<b>AOP_MSGFORMDEF</b>	<b>Form definition used to format transform errors</b>
<b>AOP_MSGPAGEDEF</b>	<b>Page definition used to format transform errors</b>
<b>AOP_PCL</b>	<b>Indicates whether the printer accepts all PCL 5 commands</b>

Figure 4-31 AFP to PCL, AFP to PostScript, or AFP to PDF environment variables

### Specifying paper trays and paper sizes

In the transform entry, you can change how the transform maps AFP input-tray numbers to actual paper trays installed in your PCL and PostScript printers. You can also specify the size of paper installed in each paper tray. You specify this information in two environment variables:

**AOP\_TRAYID** This variable specifies the tray ID in the PCL or PostScript printer that the transform should use for each AFP input tray number. The AFP input tray number for each data set is specified in the input-tray-number job attribute, INTRAY JCL parameter, or AFP form definition.

The position (one through nine) of each tray ID corresponds to the AFP input tray number. The tenth position represents any AFP input tray number greater than 9, for example:

AOP\_TRAYID -> "1 4 1 1 1 1 1 1 1 2"

If the data set being transformed specifies:

- ▶ AFP input tray 1, the transform uses tray ID 1.
- ▶ AFP input tray 2, the transform uses tray ID 4.
- ▶ AFP input tray 3 through 9, the transform uses tray ID 1.
- ▶ AFP input tray M or any other input tray, the transform uses tray ID 2.

## Transforms and TRAY ID

For the AFP to PCL transform, specify the PCL tray ID. The transform codes the PCL tray ID in the PCL Paper Source command. PCL tray IDs usually have these meanings:

1. Feed paper from a printer-specific tray
2. Feed paper from manual input
3. Feed envelope from manual input
4. Feed paper from lower tray
5. Feed paper from optional paper source
6. Feed envelope from optional envelope source

For more information about PCL tray IDs, see the description of the Paper Source command in Hewlett Packard's PCL documentation. Because the implementation of paper tray IDs can vary from printer to printer, also see the documentation for your printer. They may differ as follows:

- ▶ PCL tray IDs do not match the tray numbers embossed on the actual printer trays.
- ▶ If the usual PCL tray IDs do not work, specify different tray IDs in the AOP\_TRAYID variable until the printer selects paper from the desired paper tray.

For the AFP to PDF transform, specify 1 if you want the AFP input tray number to be available for selection. Specify 0 if you do not want it to be available.

For the AFP to PostScript transform, specify the ID used by the PostScript printer to select each tray. This value, minus 1, corresponds to an entry in the Priority array in the InputAttributes dictionary for the PostScript printer. Printer-specific values are described in the PostScript PPD file for the printer.

**AOP\_PAPER** This variable specifies the paper size for each AFP input tray number. The AFP input tray number for each data set is specified in the input-tray-number job attribute, INTRAY JCL parameter, or AFP form definition.

Paper names: a3, a4, a4ee, a5, c5, com10, dl, executiv, legal, letter, letteree, and monarch

The position (one through nine) of each paper name in this variable represents the number of the AFP input tray. The tenth position represents any AFP input tray number greater than 9.

Example:

```
AOP_PAPER -> "letter legal letteree letter letter letter letter letter letter  
letter"
```

If the data set being transformed specifies:

- ▶ AFP input tray 1, the transform formats the output for letter size paper.
- ▶ AFP input tray 2, the transform formats the output for legal size paper.
- ▶ AFP input tray 3, the transform formats the output for letteree size paper.
- ▶ APF input tray M or any other tray, the transform formats the output for letter size paper.

The AFP to PDF transform formats all output for the same paper size, using the paper size for the first AFP input tray ID selected.

## 4.32 Customization of AFP resources

- ❑ **AFP resources are collections of:**
  - Data and control information used by transforms to create the PCL, PostScript, or PDF output
- ❑ **Each transform entry using environment variables**
  - Specify default AFP resources (font, form definition, and page definition)
- ❑ **Printer definitions**
  - Specify same resources which then override default resources in the transform entry
- ❑ **AFP resources:**
  - **AOP\_CHARS - AOP\_FORMDEF - AOP\_PAGEDEF**

Figure 4-32 Specifying AFP resources using environment variables

### Specifying default AFP resources

AFP resources are collections of data and control information that the transforms use to create the PCL, PostScript, or PDF output. You can specify default AFP resources (font, form definition, and page definition) in each transform entry. You can also specify the same default AFP resources in the printer definitions for the target printers. Default resources specified in the printer definition override default resources specified in the transform entry.

In most situations, specify the same default AFP resources as you currently specify either in your PSF startup procedures (in the PRINTDEV statements) or in your PSF FSA definitions in the Printer Inventory. In the transform entry, specify the font, form definition, and page definition that apply to most of the target printers. If one or more printers require different defaults, specify those defaults in the printer definitions for the printers. In a printer definition, you can also specify other AFP resources for the transforms to use, such as an overlay for the front or back side of a page.

### Environment variables

You specify default AFP resources in the following environment variables.

#### AOP\_CHARS

This variable identifies the default font used for transform error messages and for line data and AFP data that does not specify another font. You can specify only one font in this variable. The default font can be either a raster or outline font. Recommendations are as follows:

- For the AFP to PDF and the AFP to PostScript transforms, specify an outline font because outline fonts provide higher quality output for printing and viewing.
- For the AFP to PCL transform, specify a raster font because this transform requires raster fonts.

### **AOP\_FORMDEF**

This variable identifies the default form definition used when no other form definition is specified.

### **AOP\_PAGEDEF**

This variable identifies the default page definition used when no other page definition is specified.

### **AOP\_FORMDEF considerations**

AFP resources can also be specified in these locations:

- ▶ Job attributes and JCL parameters
- ▶ Input data set
- ▶ JES initialization statements for the IP PrintWay FSA (fonts and page definitions only)
- ▶ Printer definitions in the Printer Inventory

**Hierarchy for selecting a form definition:** The transforms use search hierarchies to select resources when they are specified in more than one location. The transforms use this hierarchy to select the name of the form definition, as follows:

- ▶ The form definition specified in one of these:
  - Form-definition job attribute
  - FORMDEF JCL parameter
- ▶ The form definition in the Form definition field in the printer definition.
- ▶ The first inline form definition.
- ▶ The form definition in the AOP\_FORMDEF environment variable.
- ▶ Form definition F1CP0111. This default is coded in the transform.

**Tips:** If form definition name dummy is specified, the transform uses the first inline form definition.

After the transform determines the name of the form definition, the transform searches for the form definition first inline in the data set, and then in the user and system resource libraries.

**Hierarchy for selecting a page definition:** The transforms use this hierarchy to select the name of the page definition:

- ▶ The page definition specified in one of these:
  - page-definition job attribute.
  - PAGEDEF or FCB JCL parameter. The value in the PAGEDEF parameter overrides the FCB parameter.
- ▶ The default page definition supplied by JES to IP PrintWay basic mode.
- ▶ The Page definition field in the printer definition.
- ▶ The first inline page definition.
- ▶ The AOP\_PAGEDEF environment variable in the transform configuration file.
- ▶ Page definition P1P08682. This default is coded in the transform.

### **AOP\_PAGEDEF considerations**

If page definition name dummy is specified, the transform uses the first inline page definition.

After the transform determines the name of the page definition, the transform searches for the page definition first inline in the data set, and then in the user and system resource libraries.

JES can supply a default page definition for jobs submitted directly to IP PrintWay basic mode with JCL and not processed first by NetSpool or Print Interface. To prevent JES from supplying a default page definition to the IP PrintWay basic mode FSA:

- ▶ In JES2, do not specify the FCB parameter in the JES2 PRTnnnnn statement for the IP PrintWay FSA, and do not specify the NIFCB parameter in the JES2 PRINTDEF statement.
- ▶ In JES3, specify PDEFAULT=FCB in the JES3 DEVICE statement for the IP PrintWay FSA.

**Hierarchy for selecting a font:** The transforms use this hierarchy to select a font for line data and AFP data that does not specify a font:

- ▶ The font named in the page definition.
- ▶ The font specified in one of these locations:
  - chars job attribute.
  - CHARS or UCS JCL parameter. The value in the CHARS JCL parameter overrides the UCS parameter.
- ▶ The default font supplied by JES to IP PrintWay basic mode.
- ▶ The font in the Character set field in the printer definition.
- ▶ The font in the AOP\_CHARS environment variable. The transform prefixes X0 to the font named in the variable if you do not specify a prefix.
- ▶ Font X060D9. This default is coded in the transform.

JES can supply a default font for jobs submitted directly to IP PrintWay basic mode with JCL statements and not processed first by NetSpool or Print Interface. To prevent JES from supplying a default font to the IP PrintWay FSA:

- ▶ In JES2, specify UCS=0 in the JES2 PRTnnnnn statement.
- ▶ In JES3, specify PDEFAULT=CHARS in the JES3 DEVICE statement for the IP PrintWay FSA.

**Hierarchy for selecting a font for transform error messages:** The transforms use this hierarchy to select a font used for transform error messages:

- ▶ The font named in the page definition specified in the AOP\_MSGPAGEDEF environment variable.
- ▶ The font in the AOP\_CHARS environment variable. The transform prefixes X0 to the font named in the variable if you do not specify a prefix.
- ▶ Font X060D9. This default is coded in the transform.

### **Other ways to specify AFP resources and options:**

- ▶ You can specify AFP resources, AFP user resource libraries, and other AFP options in Infoprint Server job attributes and OUTPUT statement (JCL) parameters. Values specified in job attributes and OUTPUT parameters override the same values in the printer definition. For example, the form-definition job attribute and the FORMDEF JCL parameter override the Form definition field in the printer definition.
- ▶ You can specify AFP resources in the transform configuration file. The AFP resources specified in the printer definition override the same resources specified in the transform configuration file.
- ▶ You can specify AFP system resource libraries in the transform configuration file. The transform searches the AFP user resource libraries specified in the printer definition before the AFP system resource libraries specified in the transform configuration file.



## 4.33 Specify AFP resource libraries

- ❑ When customizing the transform entries:
  - Can specify up to eight AFP system resource libraries
    - For fonts, page definitions, form definitions, page segments, and overlays
  - AFP resource libraries can be specified in the printer definitions for the target printers
    - These resource libraries are searched before libraries specified in the transform entry
- ❑ Specify resource libraries using environment variables
  - AOP\_FONTLIB - AOP\_FORMDEFLIB - AOP\_OVERLAYLIB - AOP\_PAGEDEFLIB - AOP\_PAGESEGLIB

Figure 4-33 Customization of AFP resource libraries

### AFP resource libraries

In the printer definition, you can specify AFP resources (such as the form definition, page definition, and font) and AFP user resource libraries. In addition, you can specify other AFP options (such as the type of duplexing).

In the transform entries, you can specify up to eight AFP system resource libraries for fonts, page definitions, form definitions, page segments, and overlays. You can also specify the AFP resource libraries in the printer definitions for the target printers. Resource libraries specified in the printer definition are searched before libraries specified in the transform entry.

In most situations, specify the same default AFP resource libraries as you currently specify in your PSF startup procedures. In the transform entry, specify the libraries that apply to most of the target printers. If one or more printers require different resource libraries, specify those libraries in the printer definitions for the printers.

The environment variables that specify system resource libraries:

- |                       |  |
|-----------------------|--|
| <b>AOP_FONTLIB</b>    | This variable identifies the system font libraries.            |
| <b>AOP_FORMDEFLIB</b> | This variable identifies the system form definition libraries. |
| <b>AOP_OVERLAYLIB</b> | This variable identifies the system overlay libraries.         |
| <b>AOP_PAGEDEFLIB</b> | This variable identifies the system page definition libraries. |
| <b>AOP_PAGESEGLIB</b> | This variable identifies the system page segment libraries.    |

## Location of resource libraries

AFP resource libraries can also be specified in these locations:

- ▶ Job attributes and JCL parameters
- ▶ Printer definitions in the Printer Inventory

The user ID that starts Infoprint Server and the job submitter must be authorized to read AFP resource libraries.

## Hierarchy for searching AFP resource libraries

The transforms use this hierarchy when searching AFP resource libraries:

- ▶ Resource libraries specified by the job submitter in one of these locations:
  - resource-library job attribute
  - USERLIB JCL parameter
- ▶ Resource libraries specified in the Resource library field in the printer definition. These libraries are searched only when the job submitter does not specify any resource libraries.
- ▶ Resource libraries specified in environment variables in the transform configuration file (aopxfd.conf).
- ▶ Default resource libraries that are hard-coded in the transform. These libraries are searched only if no system resource libraries are specified in the transform configuration file.

## 4.34 Mapping AFP raster and AFP outline fonts

- ❑ Transforms can map single-byte AFP raster fonts to AFP outline fonts
- ❑ Transforms can map single-byte AFP outline fonts to AFP raster fonts (optional)
- ❑ Font mapping lets you migrate between type technologies while maintaining the same font appearance and document fidelity
- ❑ Default: font-mapping is enabled
  - Therefore, customization is required only if you want to:
    - Disable font-mapping
    - Modify the font-mapping table to add custom fonts used by your installation

Figure 4-34 AFP raster and outline font customization

### AFP raster and outline fonts

The transforms can map single-byte AFP raster fonts to AFP outline fonts and single-byte AFP outline fonts to AFP raster fonts (optional). Font mapping lets you migrate between type technologies while maintaining the same font appearance and document fidelity. By default, font-mapping is enabled. Therefore, customization is required only if you want to disable font-mapping or modify the font-mapping table to, for example, add custom fonts used by your installation.

To map fonts, these transforms use an internal font-mapping table. This table supports the IBM AFP Font Collection Expanded Core Fonts, which are available in both raster and outline formats. This font table can be customized for your installation.

### Transform font mapping

Each transform does different font mapping:

- **AFP to PCL Transform:** Maps outline fonts to raster fonts

This transform can map outline fonts to equivalent 300-pel raster fonts. The AFP to PCL transform can use only 300-pel raster fonts. If an input document uses an outline font and the transform cannot find an equivalent raster font in the mapping table, the transform generates an error and terminates printing.

Recommendation: Map outline fonts to raster fonts because this transform requires raster fonts.

► **AFP to PostScript Transform:** Maps AFP raster to AFP outline

This transform can map AFP raster fonts to equivalent AFP outline fonts. The AFP to PostScript transform can use either 300-pel raster fonts or outline fonts. If an input document uses a raster font, and the transform cannot find an equivalent outline font in the mapping table, the transform uses the raster font.

Recommendation: Map raster fonts to outline fonts because outline fonts provide superior print quality at higher printer resolutions.

► **AFP to PDF Transform:** Maps AFP raster to AFP outline

This transform can map AFP raster fonts to equivalent AFP outline fonts. The AFP to PDF transform can use either 300-pel raster fonts or outline fonts. If an input document uses a raster font, and the transform cannot find an equivalent outline font in the mapping table, the transform uses the raster font.

**Recommendation:** Map raster fonts to outline fonts because outline fonts provide superior viewing quality.

## Transform configuration file

In the transform configuration file, you can control whether font-mapping is enabled for each transform and how the AFP to PDF transform handles outline fonts. These environment variables in the configuration file let you control font mapping:

► **AOP\_FONTMAP**

This variable enables and disables font mapping. When selecting this value, consider:

- **AFP to PCL transform:** IBM recommends that you enable font mapping to map outline fonts to raster fonts because this transform requires raster fonts.
- **AFP to PostScript transform:** IBM recommends that you enable font mapping to map raster fonts to outline fonts because outline fonts provide better print quality. For the transform to map raster to outline fonts, your installation must have outline font libraries installed.
- **AFP to PDF transform:** IBM recommends that you enable font mapping to map raster fonts to outline fonts because outline fonts provide superior viewing quality. For the transform to map raster to outline fonts, your installation must have outline font libraries installed.

If your installation does not have outline font libraries installed, you do not need to disable font mapping for the AFP to PDF or AFP to PostScript transforms. If the transform does not find an outline font in the system font libraries, the transform automatically disables font-mapping for the raster font. The transform sends an error message in the transform error log once for each missing outline font. To avoid this error message, you can either remove the missing font from the font-mapping table or disable font mapping in this variable.

► **AOP\_OUTLINES** (for AFP to PDF transform only)

This variable controls whether the AFP to PDF transform includes outline fonts in the PDF output. The **yes** option (default), causes the transform to include outline fonts in the output data. The **built-in** option causes the transform to include only the names of outline fonts in the PDF output. Select the **built-in** option if you want to reduce the size of your PDF files.

## 4.35 Optional environment variables

### □ Environment variables that are optional:

- AOP\_ANNOTATIONS
- AOP\_COLOR
- AOP\_CUTSHEET
- AOP\_FLATE
- AOP\_MSGFORMDEF
- AOP\_MSGPAGEDEF
- AOP\_PJL

Figure 4-35 Optional environment variables

### Optional environment variables

The following environment variables are optional:

#### ► AOP\_ANNOTATIONS

Indicates whether the AFP to PDF transform is to produce annotations in the PDF file. The AFP to PDF transform can create an annotation on each page. The annotation contains printing instructions such as "Normal duplex Front side". For more information about annotations, see the online help provided with Adobe Acrobat® reader. This variable applies only to the AFP to PDF transform. Other transforms ignore this variable. Valid values are:

- **yes:** The AFP to PDF transform produces annotations.
- **no:** The AFP to PDF transform does not produce annotations.

#### ► AOP\_COLOR

Indicates whether the transform is to produce color output. Specify yes if the printer supports color. This variable applies only to the AFP to PCL and AFP to PostScript transforms. The AFP to PDF transform ignores this variable and always produces output in color. Valid values are:

- **yes:** The transform produces color output.
- **no:** The transform does color simulation.

► **AOP\_CUTSHEET**

Indicates whether the transform is to prepare the output for printing on a cutsheet printer. Valid values are:

- **yes**: The output is to be printed on a cutsheet printer. Therefore, the transform uses the cutsheet specification in the form definition to determine whether to send medium orientation information to the printer. For more information about the cutsheet specification in the form definition, see the description of the **CUTSHEET** command in IBM Page Printer Formatting Aid: User's Guide.

Select this option if your output is incorrectly printing in the down direction on a cutsheet printer.

- **no**: The output is not to be printed on a cutsheet printer. Therefore, the transform always sends medium orientation information to the printer.

► **AOP\_FLATE**

Indicates whether the AFP to PDF transform is to use the Adobe Flate compression algorithm to compress the output. This variable applies only to the AFP to PDF transform. The AFP to PCL and AFP to PostScript transforms ignore this environment variable. Valid values are:

- **yes**: The transform produces compressed output.
- **no**: The transform does not produce compressed output. However, throughput is higher.

► **AOP\_MSGFORMDEF**

Form definition used to format transform error messages. Specify the one to eight character form definition name, with or without the f1 prefix. If you omit the f1 prefix, the transform adds it. This form definition must be located in one of the libraries specified in the AOP\_FORMDEFLIB variable.

► **AOP\_MSGPAGEDEF**

Page definition used to format transform error messages. Specify the one to eight character page definition name, with or without the p1 prefix. If you omit the p1 prefix, the transform adds it. This page definition must be located in one of the libraries specified in the AOP\_PAGEDEFLIB variable. The transform formats messages for the first paper size defined in the AOP\_PAPER variable. IBM recommends page definition P1P08682 for letter size paper, and page definition P1Q09182 for A4 paper.

► **AOP\_PJL**

Indicates whether the printer accepts all PCL 5 commands, including PJL commands. All printers that support PCL 5 support PJL commands. Some printers that support only PCL 4 do not support PJL commands. Sometimes, a printer that does not support PJL commands prints a smiley face where a PJL command occurs in the data stream or prints the PJL command on the first page.

This environment variable applies only to the AFP to PCL transform. The AFP to PostScript and AFP to PDF transforms ignore this variable. Valid values are:

- **yes**: The printer accepts PJL commands.
- **no**: The printer does not accept PJL commands. Therefore, the AFP to PCL transform does not generate PJL commands.

## 4.36 Example afp2pcl transform class entry

```
# AFP to PCL transform with typical European paper sizes
# NOTE: An outline font library is needed for font mapping. If font mapping is wanted,
# add your outline font library (e.g. VAINI.FONT300) to the AOP_FONTLIB concatenation.
# Otherwise set AOP_FONTMAP to "no".
transform afp2pcl_fi
  start-command = afp2pcl
  min-active = 0
  max-active = 2
  maximum-idle-time = 300 # 5 minutes
  environment = {
    AOP_CHARS -> GT15
    AOP_COLOR -> no
    AOP_FONTLIB -> "vaini.font300 sys1.font300 sys1.fontlibb"
    AOP_FONTMAP -> yes
    AOP_FORMDEF -> F1A10111
    AOP_FORMDEFLIB -> "sys1.fdeflib"
    AOP_MSGFORMDEF -> F1CP0110
    AOP_MSGPAGEDEF -> P1P08682
    AOP_OVERLAYLIB -> "sys1.overlib"
    AOP_PAGEDEF -> P1V06683
    AOP_PAGEDEFLIB -> "sys1.pdeflib"
    AOP_PAGESEGLIB -> "sys1.pseglib"
    AOP_PAPER -> "a4 a3 a4 a4 a4 a4 a4 a4 a4 c5"
    AOP_TRAYID -> "1 4 0 0 0 0 0 0 2"
    _BPX_JOBNAME -> AFP2PCLD  }
;
```

Figure 4-36 Example afp2pcl transform class entry

### Example of an afp2pcl transform

Figure 4-36 shows a sample of an afp2pcl transform that is customized.

For this type of transform, do the following:

- ▶ Specify AFP 300-pel raster font libraries because the AFP to PCL transform requires raster fonts.
- ▶ Specify AFP outline font libraries if data to be transformed refers to outline fonts.

As shown in Figure 4-36, an outline font library is needed for font mapping. If font mapping is wanted, add your outline font library (VAINI.FONT300) to the AOP\_FONTLIB concatenation. Otherwise set the environment variable AOP\_FONTMAP to **no**.

### Transform Manager

When the Transform Manager starts, it starts two types of transform daemons, one to transform jobs that do not specify a transform class and another to transform jobs that specify a transform class such as fi, as shown in Figure 4-36 (afp2pcl\_fi). The Transform Manager keeps no more than two of each type of transform daemons active at any time. The Transform Manager shuts down any transform daemon that is idle and frees system resources after 5 minutes. The Transform Manager keeps at least one transform daemon of the first type active.

## 4.37 Enhanced PDF documents

- ❑ The AFP to PDF transform environment variables:
  - AOP\_INDEX: Creates bookmarks in PDF documents
  - AOP\_INDEX\_LANG: Specifies the language for converting text
  - AOP\_LINEARIZE: Optimizes PDF documents for fast viewing
  - AOP\_LINKS: Creates links in PDF documents for improved navigation
  - AOP\_ROTATE\_PDF: Specifies how to rotate PDF documents for easier viewing
- ❑ PDF encryption
  - AOP\_ENCRYPT: Enables encryption
  - AOP\_PASSWORD\_EXIT: The name of Password exit
  - AOP\_PROTECT: Restricted actions without password

Figure 4-37 Enhanced PDF documents

### Enhanced PDF documents

The AFP to PDF transform can create PDF documents that are easier to navigate and view. The transform provides the following environment variables so that you can create enhanced PDF documents:

- ▶ AOP\_INDEX: Creates bookmarks in PDF documents for improved navigation. Bookmarks are created when the AFP document contains Tag Logical Elements (TLE) structured fields.
- ▶ AOP\_INDEX\_LANG: Specifies the language for converting text in bookmarks.
- ▶ AOP\_LINEARIZE: Optimizes PDF documents for fast viewing from the Web.
- ▶ AOP\_LINKS: Creates links in PDF documents for improved navigation. Links are created when the AFP document contains Link Logical Elements (LLE) structured fields.
- ▶ AOP\_ROTATE\_PDF: Specifies how to rotate PDF documents for easier viewing.

To use these new functions, you specify the environment variables in the Infoprint Server configuration file.

### PDF encryption

The AFP to PDF transform can now encrypt PDF documents. Encrypting PDF documents provides enhanced security for sending PDF documents over a network. In addition, you can associate user and owner passwords with encrypted PDF documents to prevent



unauthorized access, and you can restrict users from modifying, printing, and doing other actions when they open encrypted PDF documents.

The transform supports the following job attributes:

- ▶ pdf-encryption-level: The level of encryption.
- ▶ pdf-owner-identifier: An optional owner identifier. The transform passes this identifier to the Password exit, which provides the owner password.
- ▶ pdf-protect: The actions, such as modifying or printing, that you want to restrict in encrypted PDF documents.
- ▶ pdf-user-identifier: An optional user identifier. The transform passes this identifier to the Password exit, which provides the user password.

The transform now supports new environment variables:

- ▶ AOP\_ENCRYPT: Enables encryption.
- ▶ AOP\_PASSWORD\_EXIT: The name of your installation's Password exit.
- ▶ AOP\_PROTECT: The actions (such as modifying and printing) that are restricted in PDF documents when no passwords are specified.

The transform also provides a sample Password exit and password database. The Password exit provides passwords to the transform. (For security reasons, job submitters cannot specify passwords during job submission.)

## 4.38 To use PDF encryption

- ❑ Install and customize the z/OS Open Cryptographic Services Facility (OCSF) feature
- ❑ Write a Password exit and create a password database or use an existing one (Optional)
  - Associate passwords with the encrypted PDF documents (Required)
- ❑ Enable encryption and specify options in environment variables
- ❑ (Optional) Update printer definitions to specify encryption options
  - This step is required if Infoprint Server job attributes cannot be specified when print jobs are submitted
- ❑ Use the afp2pdf command to:
  - Create an encrypted PDF document
  - Submit a print job to create an encrypted PDF document and send it to an e-mail destination

Figure 4-38 To use PDF encryption

### OCSF

To encrypt PDF documents, you must install and customize Open Cryptographic Services Facility (OCSF) with the Security Level 3 feature of OCSF. OCSF is part of z/OS Cryptographic Services.

OCSF requires that the user ID that starts Infoprint Server daemons have READ access to the following profiles in the RACF FACILITY class:

- ▶ BPX.SERVER
- ▶ CDS.CSSM
- ▶ CDS.CSSM.CRYPTO
- ▶ CDS.CSSM.DATALIB

Users who start Infoprint Server must be members of the AOOPER RACF group, or have a UID of 0. (AOOPER is the default group name for Infoprint Server operators. However, your installation can assign a different name to this group.) Therefore, you should give the AOOPER group READ access to these profiles.

### Password exit

To encrypt PDF documents with passwords, you must provide a Password exit to the AFP to PDF transform. (To encrypt PDF documents without passwords, you do not need a Password exit.)

The types of PDF passwords are:

- |                       |  |
|-----------------------|--|
| <b>User password</b>  | Lets someone open an encrypted PDF document. A user password is optional. If a user password is not specified, anyone can open the encrypted PDF document.   |
| <b>Owner password</b> | Lets you restrict actions in an encrypted PDF document. The owner password also lets someone open an encrypted PDF document and bypass restrictions. An owner password is optional. However, it is required to restrict actions. |

## Sample Password exit

IBM provides a sample Password exit and sample password database:

- ▶ `/usr/lpp/Printsrv/lib/aoppdfexit.dll`: This sample Password exit looks up passwords in the sample password database and returns a password for each identifier that is passed to it as input. It does not change the restricted actions.

The sample Password exit accepts the name of the password database as an argument. If no argument is specified for the sample exit in the `AOP_PASSWORD_EXIT` environment variable, the exit looks for the database in `/etc/Printsrv/aoppdfexit.db`.

- ▶ `/usr/lpp/Printsrv/samples/aoppdfexit.db`: A sample password database that maps identifiers to passwords. To use the sample database with the sample exit, copy it to `/etc/Printsrv/aoppdfexit.db` because the sample exit looks for the password database in that location.

IBM provides the source code for the sample Password exit (`aoppdfexit.dll`) in these files:

- ▶ `/usr/lpp/Printsrv/samples/aoppdfexit.h`: A header file that contains declarations and interface descriptions.
- ▶ `/usr/lpp/Printsrv/samples/aoppdfexit.c`: The source code for the sample Password exit.

## Enable encryption and specify encryption options

Environment variables that specify encryption options in the transform configuration file:

- ▶ `AOP_ENCRYPT`: Enables encryption. When you enable encryption, the transform encrypts documents if either of these conditions is met:
  - The job submitter specifies a user or owner identifier in a job attribute, or submits a print job to a printer definition that specifies a user or owner identifier.
  - The `AOP_PROTECT` environment variable is specified in the AFP to PDF transform class.
- ▶ `AOP_PASSWORD_EXIT`: The name of your Password exit and optional arguments. The transform calls this exit to obtain PDF owner and user passwords when an owner or user identifier is specified.

This environment variable is used if encryption is enabled and a user or owner identifier is specified. Otherwise it is ignored.

- ▶ `AOP_PROTECT`: The actions to be restricted in all PDF documents when no identifiers are specified. You can restrict copying, printing, and updating PDF documents.

This environment variable is ignored if either a user or owner identifier is specified.

Consider specifying the `AOP_PROTECT` environment variable in a separate transform class. Use this transform class only for documents you want to restrict actions on.

## Update printer definitions to specify encryption options

In the printer definition, you can specify options that the AFP to PDF transform uses when it encrypts PDF documents with passwords.

IP PrintWay Printer Definition Processing panel AFP to PDF Transform Encryption fields:

**AFP to PDF Transform Encryption:**

User identifier . (extend)

Owner identifier. (extend)

Encryption level 2 1. Low (40-bit key) 2. High (128-bit key)

Protected actions:

Restrict print. . . 2 1. Yes 2. No

Restrict copy . . . 2 1. Yes 2. No

Restrict update . . 2 1. Yes 2. No

The printer definition fields (and the corresponding PIDU attributes) that let you specify encryption options are as follows:

- ▶ **Encryption level (pdf-encryption-level):** The level of encryption used to encrypt PDF documents.
  - **Low** (40-bit key): A low level of encryption (a 40-bit encryption key) is used. Select this value if you e-mail PDF documents to countries that do not use 128-bit encryption, or for users with Adobe Acrobat Reader 3.0 - 4.x.
  - **High** (128-bit key): A high level of encryption (a 128-bit encryption key) is used. Select this value for sensitive PDF documents.
- ▶ **Owner identifier (pdf-owner-identifier):** The identifier of the owner of an encrypted PDF document. The owner identifier is associated with a password that is stored in a separate database. An owner password is required to restrict actions in the Protected actions field.
- ▶ **Protected actions (pdf-protect):** The actions that are restricted in encrypted PDF documents:
  - **Copy:** Users cannot:
    - Copy or extract content to another document
    - Extract content for accessibility
  - **Print:** Users cannot:
    - Print at low resolution (150 dpi)
    - Print at high resolution
  - **Update:** Users cannot:
    - Change the document
    - Assemble (insert, rotate, and delete pages)
    - Add comments
    - Fill in form fields or sign
    - Create template pages
- ▶ **User identifier (pdf-user-identifier):** The identifier of the user of an encrypted PDF document. The user identifier is associated with a password that is stored in a separate database. The user password is required to open encrypted PDF documents.

You can enter from 1 - 256 characters, with any combination of letters, numbers, blanks, and special characters that the Password exit allows.

This step is required if job submitters cannot specify Infoprint Server job attributes when they submit a print job.

## Create an encrypted PDF document

Use the **afp2pdf** command to create an encrypted PDF document, or submit a print job to create an encrypted PDF document and send it to an e-mail destination. Specify encryption options in job attributes or submit the print job to a printer definition that contains encryption options. The job attributes for encrypting PDF documents are: pdf-encryption-level, pdf-owner-identifier, pdf-protect, and pdf-user-identifier.

## 4.39 Example - Create an encrypted PDF document

```
afp2pdf -c encrypt -o myfile.pdf myfile.afp
lp -o "filter-options='-c encrypt'" -d myprinter myfile.afp
```

```
transform afp2pdf_encrypt
start-command = afp2pdfd
minimum-active = 1
maximum-active = 2
maximum-idle-time = 300 # 5 minutes
environment = {
    _BPX_JOBNAME -> AFP2PDFD
    AOP_ANNOTATIONS -> yes
    AOP_CHARS -> 60D9
    AOP_CUTSHEET -> no
    AOP_ENCRYPT -> yes
    AOP_FLATE -> yes
    AOP_FONTLIB -> "sys1.font300 sys1.fontln"
    AOP_FONTMAP -> yes
    AOP_FORMDEF -> F1CP0110
    AOP_FORMDEFLIB -> "sys1.fdeflib"
    AOP_INDEX -> yes
    AOP_INDEX_LANG -> 037
    AOP_LINEARIZE -> yes
}
```

```
AOP_LINKS -> yes
AOP_MSGFORMDEF -> F1CP0110
AOP_MSGPAGEDEF -> P1P08682
AOP_OUTLINES -> yes
AOP_OVERLAYLIB -> "sys1.overlib"
AOP_PAGEDEF -> P1P08682
AOP_PAGEDEFLIB -> "sys1.pdeflib"
AOP_PAGESEGLIB -> "sys1.pseglib"
AOP_PAPER ->
    "letter letter letter letter letter letter letter letter letter letter"
AOP_PASSWORD_EXIT -> ->
    "/usr/lpp/Printsrv/lib/aoppdfexit.dll"
AOP_PROTECT -> "modify print select"
AOP_POSITIONING_METHOD -> cell
AOP_ROTATE_PDF -> no
AOP_TRAYID -> "1 1 1 1 1 1 1 1 1"
}
```

Figure 4-39 Create an encrypted PDF document

### Example - Transform configuration file entry for AFP to PDF

The transform entry shown in Figure 4-39 creates transform class "encrypt." To use this transform class, specify the class in the -c transform option as shown in these two examples:

```
afp2pdf -c encrypt -o myfile.pdf myfile.afp
lp -o "filter-options='-c encrypt'" -d myprinter myfile.afp
```

## 4.40 Scaling fonts using the AOXCF30 program

```
/* LICENSED MATERIALS - PROPERTY OF IBM - 5697-F51
/* (C) COPYRIGHT IBM CORP. 2000
/* AFP TRANSFORM (C) COPYRIGHT I-DATA INTERNATIONAL 1990,2000
/* AOXCF30 PROGRAM in SYS1.SAMPLIB(AOXCF30J)
/* CONVERTS SBCS AND DBCS BOUNDED BOX FONTS FROM 240 TO 300 DPI
/* USING A PROC INVOCATION OF THE AOXCF30 FONT SCALING PROGRAM.
/* THE SUPPLIED PROC STATEMENTS ARE PROVIDED AS A SAMPLE ONLY, AND
/* SHOULD BE TAILORED TO MATCH INSTALLATION REQUIREMENT FOR
/* SCALING OF FONTS TO 300 DPI.
/* BOX DRAWING CHARACTERS LOCATED IN FONTS CAN BE REPLACED USING
/* THE BOX PARAMETER, AND SUPPLIED BOX DRAW CHARACTERS (FONTB300)
/* PARM='BOX' USE SUPPLIED BOX CHARACTER SET (DEFAULT)
/* PARM='NOBOX' DO NOT USE THE SUPPLIED BOX CHARACTER SET
/* S2 EXEC PROC=SCALE,NAME=C0XXXXXX,PARM='BOX'
/* (FONT IMAGE MEMBER NAME)
//SCALE PROC HLQ=AOX,NAME=,PARM1=BOX
//S1 EXEC PGM=AOXCF30,REGION=5120K,PARM='&PARM1'
//SYSPRINT DD SYSOUT=*
/* SYSUT1 IS THE INPUT 240 DPI FONT LIBRARY
//SYSUT1 DD DISP=SHR,DSN=SYS1.FONTLIBB(&NAME.)
/* SYSUT2 IS THE OUTPUT 300 DPI FONT LIBRARY
//SYSUT2 DD DISP=SHR,DSN=SYS1.FONT300(&NAME.)
//FONTB300 DD DISP=SHR,DSN=&HLQ.SAOXCF30
// PEND
//C0XXXXXX EXEC PROC=SCALE,NAME=C0xxxxxx
//C0YYYYYY EXEC PROC=SCALE,NAME=C0yyyyyy
```

Figure 4-40 AOXCF30 program

### Scaling 240-pel to 300-pel fonts using the AOXCF30 program

The AOXCF30 font-conversion program provided with Infoprint Server Transforms lets you scale (that is, convert) your single-byte, bounded-box, 240-pel fonts to 300-pel fonts. This program places the scaled fonts into a partitioned data set, which you can then use with the transform.

You must scale 240-pel fonts to 300-pel fonts if you do not already have 300-pel fonts and either of these conditions apply:

- ▶ You use the AFP to PCL transform. This transform requires 300-pel fonts.
- ▶ You use the AFP to PostScript or AFP to PDF transform, documents to be transformed reference raster fonts, and you do not map raster fonts to outline fonts.

If you have previously scaled your 240-pel fonts using the PSF-supplied APSRCF30 program (required for the IBM 4028 printer), you do not need to use the AOXCF30 program to scale them again. Because IBM Core Interchange raster fonts are shipped in both 240-pel and 300-pel resolutions, you do not need to convert them. Outline fonts are resolution-independent and do not need to be converted.

### AOXCF30 program

The AOXCF30 program can convert 240-pel single-byte and double-byte raster fonts to 300-pel raster fonts. The AOXCF30 program converts a 240-pel raster font to a close

approximation of the font at 300-pel resolution. However, IBM does not warrant the quality of the resulting character pattern when using this program.

Specific fixed pitch fonts require the replacement of box characters. These are supplied in the AOX.SAOXCF30 library.

### Using the AOXCF30 program

- ▶ 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 can use the existing 240-pel font library as a model. However, the 300-pel font library requires approximately 30% more space than the 240-pel font library.

- ▶ Create a list of 240-pel fonts that require conversion.

Only the character set members, whose file names start with **CO**, are converted with AOXCF30. However, you can also specify coded font members, whose file names start with **XO**, and code page members, whose file names start with **TO**. The coded fonts and code pages are copied to the output file without conversion.

- ▶ \_ Modify the JCL supplied in SYS1.SAMPLIB(AOXCF30J).

Specify an entry in the JCL for each font member to be converted. The sample JCL is shipped with two dummy entries:

```
//COXXXXXX EXEC PROC=SCALE,NAME=COxxxxxx  
//COYYYYYY EXEC PROC=SCALE,NAME=COyyyyyy
```

Duplicate these statements for each font that is to be converted.

- ▶ Submit the JCL to run the AOXCF30 program.

This program can use a large amount of CPU resources, especially if you run it on a smaller S/370™ processor. To run the AOXCF30 program, you should have a minimum region size of 5 MB.

Archived



## IP PrintWay

IP PrintWay transmits output data sets from the JES spool to remote printers or print servers and to e-mail destinations.

You can run either IP PrintWay basic mode or IP PrintWay extended mode:

- ▶ IP PrintWay basic mode, the original mode of operation, uses the z/OS Functional Subsystem Interface (FSI) to obtain output data sets from the JES spool. IBM does not plan additional enhancements to IP PrintWay basic mode.
- ▶ IP PrintWay extended mode, starting with z/OS V1R5, uses the z/OS Sysout Application Programming Interface (SAPI) to obtain output data sets from the JES spool. It provides better performance, improved usability, and more function than IP PrintWay basic mode.

IP PrintWay basic mode and IP PrintWay extended mode use the same printer definitions in the Printer Inventory.

In addition:

- ▶ IP PrintWay can transmit output data sets to remote printers or host systems in a TCP/IP network. A print server can be running on the host system. The remote printer or host system must support one of these TCP/IP protocols:
  - LPR/LPD protocol
  - Internet Printing Protocol (IPP)
  - Direct socket printing protocol
- ▶ IP PrintWay can transmit output data sets to one or more e-mail addresses. Each output data set is sent as an e-mail attachment. If you install the AFP to PDF transform feature of Infoprint Server Transforms, IP PrintWay can transform line data and AFP data so that the e-mail recipients can view and print the output with Adobe Acrobat Reader.
- ▶ IP PrintWay basic mode can transmit output data sets to VTAM-controlled printers that are defined to VTAM. The printers must be defined as LU0, LU1, or LU3 printers. Supported output data streams are Data Stream Compatibility/Data Stream Extended (DSC/DSE) and SNA Character String (SCS). VTAM printer support in IP PrintWay requires the Coax Printer Support feature of Infoprint Server.

## 5.1 IP PrintWay

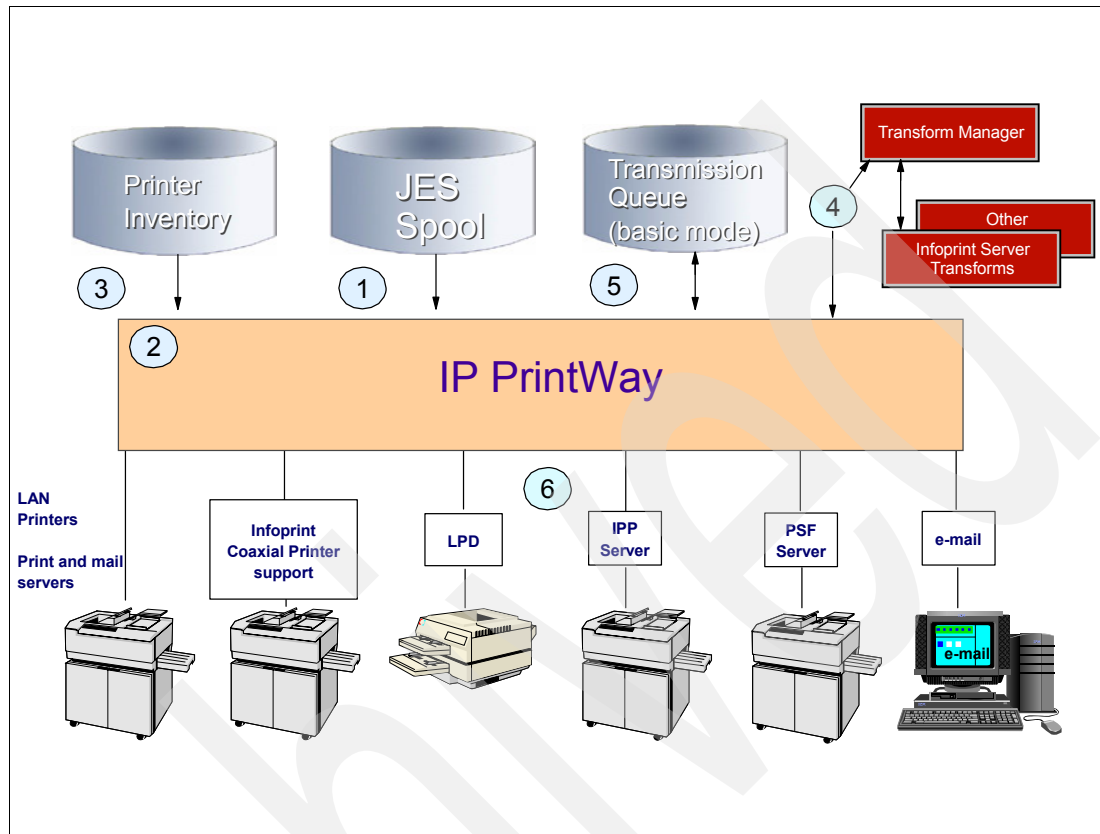


Figure 5-1 IP PrintWay processing steps

### IP PrintWay processing steps

Figure 5-1 shows the steps that occur from the time IP PrintWay selects output data sets from the JES spool until it transmits them to the printer or e-mail destination and then deletes them from the JES spool. An explanation of each step follows.

1. IP PrintWay selects output data sets from the JES spool. IP PrintWay can select data sets that were allocated on the JES spool by NetSpool or Print Interface, or data sets submitted from z/OS batch jobs using JCL. The data sets can contain line data, ASCII text data, or formatted data, such as PCL, PostScript, SAP, or MO:DCA-P (AFP) data.
2. IP PrintWay extended mode uses the SAPI interface to select output data sets from the JES spool. IP PrintWay basic mode runs as a JES functional subsystem (FSS). An IP PrintWay basic mode functional subsystem application (FSA) uses the FSI interface to select output data sets from the JES spool.
3. IP PrintWay uses information in printer definitions in the Printer Inventory to process data sets, select the transmission protocol, and obtain printer addresses. IP PrintWay can, instead, use the printer's IP address specified on the OUTPUT JCL statement or in an Infoprint Server job attribute.

IP PrintWay recognizes data sets allocated on the JES spool by Print Interface and NetSpool and does not convert data from EBCDIC to ASCII or format the data if Print Interface or NetSpool has already converted data to ASCII. For other data sets, IP PrintWay can convert data between EBCDIC and ASCII, can add a header to each page, and can format data using the carriage-control characters in line data, an FCB, or pagination attributes specified in the printer definition.

4. IP PrintWay extended mode can, in most cases, automatically detect the input data format and transform data into the format required by the printer or e-mail destination. IP PrintWay basic mode can transform data by resubmitting the data stream to Print Interface. To do transforms, IP PrintWay uses transforms that Infoprint transform products provide.
5. When you run IP PrintWay extended mode, operators can use Infoprint Central to work with output data sets that IP PrintWay is processing or has retained on the JES spool. In addition, Infoprint Central lets you work with data sets that IP PrintWay has not yet selected for processing.

When you run IP PrintWay basic mode, the IP PrintWay transmission queue data set contains information about output data sets that IP PrintWay is processing or has retained on the JES spool. Operators must use Infoprint Server ISPF panels to manage these data sets.
6. IP PrintWay transmits data sets to the printer, print server, or e-mail destination using the protocol selected in the printer definition: LPR, direct sockets, IPP, e-mail, or VTAM. IP PrintWay can also transmit LPD options and IPP job attributes to the printer's LPDs and IPP Servers. For example, IP PrintWay can transmit information that the LPD prints on a separator page.

## IP PrintWay functions

Additional functions that IP PrintWay provides include:

### ► Retry

IP PrintWay can retry an unsuccessful transmission a specified number of times at a specified interval. Retry limits and retry times can be specified in the printer definition and on an OUTPUT JCL statement. In addition to the requested retries, IP PrintWay retries an unsuccessful transmission automatically for a short period of time right after transmission.

If an error occurs in the middle of a print job, IP PrintWay restarts printing from the beginning of the print job. However, IP PrintWay extended mode can restart printing after the last page that printed successfully if the printer supports Hewlett Packard (HP) Printer Job Language (PCL) and the direct sockets printing protocol. The administrator must request in the printer definition that IP PrintWay extended mode restart printing after the last successful page.

### ► Retention of jobs on the JES spool

After successfully transmitting each data set, or after completing the requested number of transmission attempts, IP PrintWay can retain the data set on the JES spool forever or for a period of time. Retention periods can be specified in the printer definition or on an OUTPUT JCL statement.

### ► Transforms

IP PrintWay can transform line data to text format for printing on a network printer or e-mailing without using a separate transform product.

IP PrintWay can transform data to PCL, PDF, or PostScript format for printing or e-mailing.

### ► Validation of print requests

IP PrintWay extended mode can validate, with some exceptions, that the document can print as requested on the selected printer. For example, IP PrintWay extended mode can reject documents with data formats that the printer does not support or that are too large to print on the selected printer.

### ► Operator control

When you run IP PrintWay extended mode, the operator can use tools such as SDSF and Infoprint Central to work with output data sets (called print jobs) that are intended for IP

PrintWay or that IP PrintWay has retained. In addition, Infoprint Central lets you delete, hold, and move print jobs that IP PrintWay extended mode is processing. Infoprint Central can display additional status such as whether the print jobs have completed successfully, were retained due to failed transmission to LAN printers, or were deleted before printing.

When you run IP PrintWay basic mode, the operator must use Infoprint Server ISPF panels to work with output data sets that IP PrintWay processes.

- **Job selection rules**

When you run IP PrintWay extended mode, the administrator defines job selection rules in the Printer Inventory to specify which print jobs to select. The operator can use Infoprint Central to start and stop these job selection rules, or the administrator can automate the starting and stopping of job selection rules.

- **Printer selection using an OUTPUT JCL statement**

On an OUTPUT JCL statement, a user can select the printer definition by specifying either (1) the name of the printer definition in the FSSDATA parameter or (2) the DEST, CLASS, or FORMS parameter (or a combination of these parameters) associated with the printer definition.

Users can also specify the IP address for the printer directly on the OUTPUT JCL statement, thereby eliminating the need to create a printer definition for each printer in the Printer Inventory.

- **SMF accounting**

For each data set processed, IP PrintWay writes a System Management Facility (SMF) type 6 record. The SMF type 6 record identifies the job submitter and the address of the printer. It also contains the number of bytes that were transmitted to the printer. IP PrintWay extended mode can also record the number of pages that printed successfully if the printer supports Hewlett Packard (HP) Printer Job Language (PJM) and the direct sockets printing protocol.

- **Exits and filters**

IP PrintWay lets you write exits to customize IP PrintWay processing. You can also write filter programs to provide your own data stream transforms.

- **Transmission order**

IP PrintWay preserves the order of the data sets on the JES spool when transmitting data sets. IP PrintWay retains this order even if the transmission of the data sets must be retried. If a JES output group contains more than one output data set, IP PrintWay acquires all of the data sets in the output group before transmitting any of them and can transmit these data sets to the printer as a single file.

- **Printer commands**

The administrator can specify printer commands for IP PrintWay to send to the printer before or after the data to be printed. Printer commands can be used to change fonts or switch between simplex and duplex printing. Printer commands are specified in the printer definition or in the Begin Data Set exit.

- **IPsec data encryption**

The IP security function of z/OS Communications Server can encrypt data that IP PrintWay sends to printers and print servers that support Internet Protocol security (IPsec).

## 5.2 IP PrintWay modes of operation

### IP PrintWay Extended Mode:

- Uses the Process Sysout API (SAPI) to:
  - Obtain jobs from the JES spool - send them to remote TCP/IP-attached printers and e-mail destinations
- Provides better control of work selection criteria
- Enables administrators to more easily:
  - Cancel the currently printing job
  - Stop and restart printing to a printer
  - Restart retained jobs
- Cannot share printing information across systems in a sysplex

### IP PrintWay Basic Mode (FSS)

*Figure 5-2 IP PrintWay modes of operation*

### IP PrintWay modes

IP PrintWay extended mode uses the z/OS Sysout Application Programming Interface (SAPI) to obtain output data sets from the JES spool. This implementation results in better performance and improved usability. In addition, IP PrintWay extended mode provides additional functions that help you manage printers and print jobs.

IP PrintWay basic mode is the name used for the original IP PrintWay mode of operation, to distinguish the original mode from the IP PrintWay extended mode. You can run IP PrintWay in both basic mode and extended mode. After the introduction of IP PrintWay extended mode, IBM stopped producing enhancements for IP PrintWay basic mode.

IP PrintWay extended mode and IP PrintWay basic mode use the same printer definitions in the Printer Inventory.

### Comparing IP PrintWay extended mode to basic mode

IP PrintWay extended mode provides enhancements as compared to IP PrintWay basic mode:

- ▶ Better performance
  - Most jobs start printing sooner.
  - A printer problem or large job on one printer does not delay printing to another printer.

- IP PrintWay extended mode calls data stream transforms and other filters directly without resubmitting them to Print Interface for filtering. Calling transforms directly is more efficient because data is not written to the JES spool a second time.
- Improved usability
 

Operators can use Infoprint Central, instead of Infoprint Server ISPF panels, to work with IP PrintWay extended mode printers and print jobs.

Operators can use JES commands to work with print jobs that IP PrintWay extended mode has selected to process, is waiting to retry, or has retained on the JES spool. However, to work with print jobs that IP PrintWay is currently processing, operators must use Infoprint Central.

### **Extended mode sysplex support**

IP PrintWay extended mode cannot share printing information across multiple systems in a sysplex as IP PrintWay basic mode can:

- IP PrintWay basic mode keeps queue information in its transmission-queue data set and keeps messages in its message-log data set. Both data sets can be shared across multiple systems.
- IP PrintWay extended mode keeps queue information in the Printer Inventory and keeps messages in the common message log. The Printer Inventory and common message log cannot be shared across multiple systems.

Therefore, if you configure IP PrintWay extended mode to print on the same printers from different z/OS systems:

- You might see more time-out situations. A time-out situation occurs if a printer is busy when IP PrintWay sends data to it. However, while IP PrintWay extended mode waits for a printer to become available, printing on other printers is not affected.
- To see all print jobs on a printer's queue and all messages from IP PrintWay about a printer, Infoprint Central users must open separate instances of their Web browser so they can log on to multiple z/OS systems. This is because from each instance of a Web browser, you can log on to only one z/OS system and see print jobs, printer queues, and Infoprint Server messages for that one z/OS system only.

### **Extended mode versus basic mode**

IP PrintWay extended mode always uses the z/OS iconv utility to convert between EBCDIC and ASCII code pages. It does not use standard or customized TCP/IP translate tables as IP PrintWay basic mode can.

IP PrintWay extended mode does not call the Message exit and the Response Notification exit. Also, IP PrintWay extended mode does not pass some fields in the ANFUEXTTP control block to IP PrintWay exits, and it ignores requests from the exits to hold or delete data sets and write messages to the console.

IP PrintWay extended mode cannot print to VTAM-controlled printers in the SNA network.

IP PrintWay extended mode ignores the NOTIFY parameter on the OUTPUT JCL statement and the Notify field in the printer definition.

IP PrintWay extended mode prints all data sets in an output group on the printer selected for the first data set in the output group. In addition, retry values and retention periods for the first data set apply to all data sets in the output group.

## 5.3 SAPI process

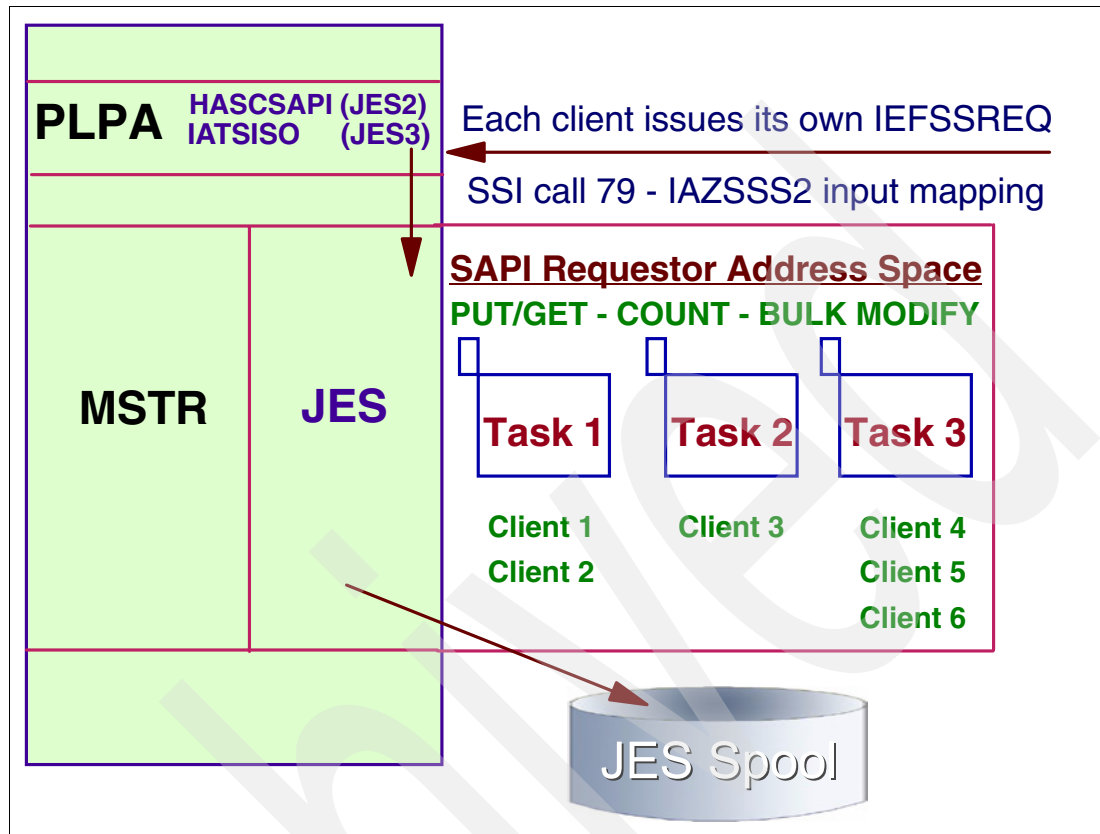


Figure 5-3 Sysout Application Programming Interface (SAPI) design

### SAPI processing

The SYSOUT Application Program Interface Data Set Call (SSI Function Code 79) allows a user-supplied program to access JES SYSOUT data sets independently from the normal JES provided functions (such as print, network). Users of this function are application programs operating in an address space external to JES.

### Client access to SAPI

The SYSOUT Application Program Interface supports multiple, concurrent requests from the application's address space. Each issuer of the IEFSSREQ macro is referred to as an application thread (client/server thread).

### Control block communication

The IAZSSS2 (SSOB Extension for SYSOUT Application Program Interface (SAPI) - SSS2) mapping macro is used as input to the IEFSSREQ request for SAPI processing. Fields in the SSS2 macro are differentiated into input, output, and disposition fields.

- ▶ An issuer's application thread sets input fields upon each IEFSSREQ invocation.
- ▶ JES manages output fields. JES updates the output-defined fields in response to each IEFSSREQ invocation.
- ▶ An issuer's application thread sets the disposition fields on an obtain data set request to inform JES of the disposition processing to occur for the data set returned on the prior obtain data set request.

## SYSOUT Application Program Interface Request Types

An application thread can make three types of requests with SAPI. Each is independent of, and mutually exclusive with the others:

### ► PUT/GET

Requests data set selection, and optionally can provide disposition processing for the data set returned in the previous SAPI PUT/GET call.

PUT/GET request processing occurs when an application thread issues the IEFSSREQ macro to initiate data set selection. The input Subsystem Options Block (SSOB) and SSS2 control blocks, provided by the application thread, specifies the selection criteria used to select a data set. The application thread can use a wide variety of selection criteria to select a SYSOUT data set to be processed.

Once the application thread receives a data set from the JES, you must allocate (through a dynamic allocation with the data set name that is returned from SSS2DSN) the data set to process it. During this allocation, dynamic allocation requires browse token (DALBRTKN) text unit. JES performs the initialization of this text unit. The application thread must move the address from field SSS2BTOK into a text unit pointer field for the JES-provided DALBRTKN text unit. The actual processing of the SYSOUT data set depends upon your specific application. After your application thread has completed processing of the data set, it then unallocates the data set with the text unit of DUNDDNAM specifying the DDNAME of the returned data set from the original allocation. The allocation/unallocation of the data set must occur once per returned data set.

The PUT processing occurs when the application thread subsequently issues a following IEFSSREQ macro to select another data set. You can use fields in the optional disposition section of the SSS2 to change certain attributes of the previously obtained data set from the prior IEFSSREQ call.

### ► COUNT

Returns the count of entries that can be scheduled without returning a particular data set. JES counts the number of schedulable elements (OSEs/JOEs) matching the input selection criteria and returns the count to the application thread in field SSS2CDS. An application thread does not receive a data set in the SAPI COUNT call. Included in the information returned are the total byte count, record count, line count, and page count.

### ► BULK MODIFY

Modifies selected attributes of one or more data sets.

With a BULK MODIFY request, the application thread can select SYSOUT data set(s) for modifications. Modification of data sets matching the input selection criteria occurs with the setting of information in the SSS2:

- Class update - The class is changed to the specified class in the SSS2.
- Delete processing - Each data set is deleted.
- Destination update - The destination is changed to the specified in the SSS2.
- Release processing - Each data set is moved to the WRITER queue in JES3, and marked non-held in JES2.

Release processing is applicable only to data sets on the JES3 Output Service HOLD queue, or for those data sets with dispositions of HOLD or LEAVE for JES2.



## 5.4 SAPI client request to JES

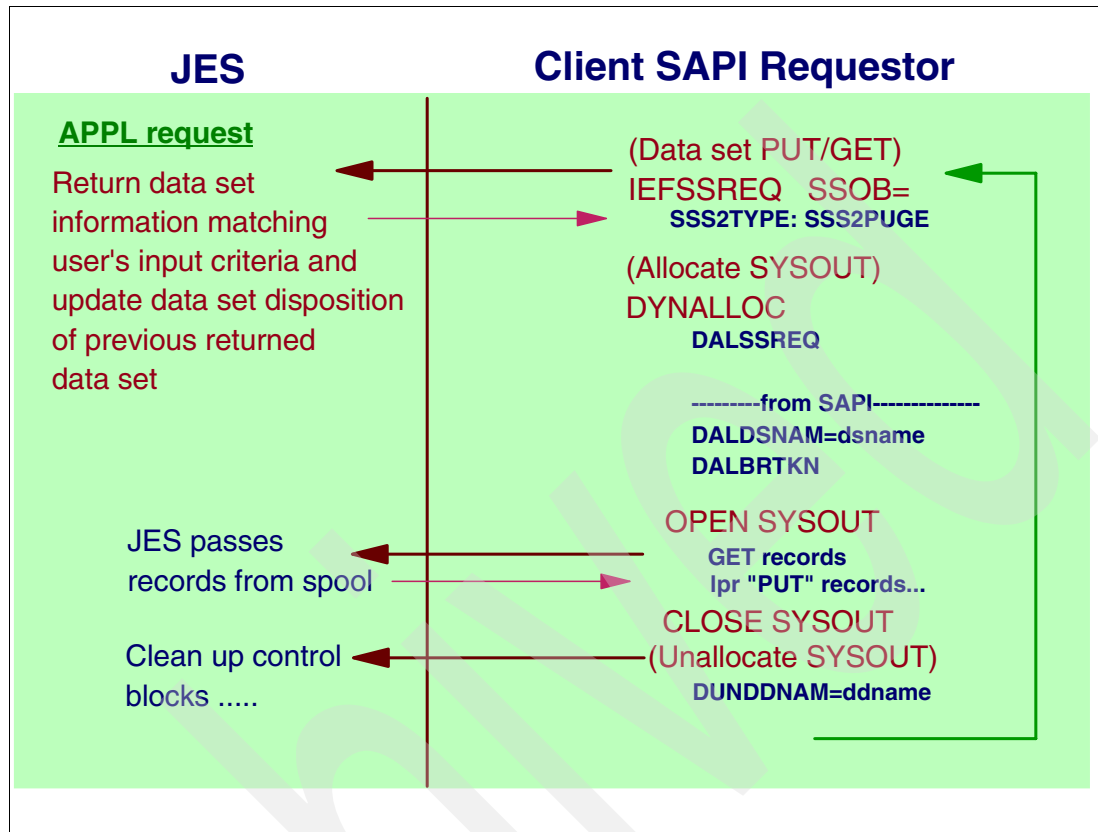


Figure 5-4 Client requesting a data set from JES

### SAPI client request to JES

PUT/GET request processing occurs when an application thread issues the IEFSSREQ macro to initiate data set selection. The input SSOB and SSS2 control blocks, provided by the application thread, specifies the selection criteria used to select a data set.

Information contained within the SYSOUT data set's scheduler work blocks (SWBs) can also be returned to the application thread. Much of the information contained within the SWB is normally not processed by JES, and therefore much more information about the data set can be retrieved from the SWB than is returned in fields of the SSS2.

The application thread needing to retrieve the SWB information, sets SSS2 flags when issuing a PUT/GET request. JES then provides the application thread the information that can be used when the application thread invokes the SJF services to retrieve this SWB information. These services are either SJFREQ REQUEST=RETRIEVE or SWBTUREQ REQUEST=RETRIEVE.

### Dynamic allocation

Once the application thread receives a data set from the JES, it allocates (through a dynamic allocation) the data set to process it. During this allocation, dynamic allocation requires DALBRTKN text unit. JES initializes this text unit and the application thread moves the address from SSS2 into a text unit pointer field for the JES-provided DALBRTKN text unit.

### **Process output data set**

JES passes the output data set to the application by reading the spool and passing the SYSOUT records. The actual processing of the SYSOUT data set depends upon your specific application.

### **Unallocate data set**

After the application thread has completed processing of the data set, it then unallocates the data set with the text unit of DUNDDNAM specifying the DDNAME of the returned data set from the original allocation. The allocation/unallocation of the data set must occur once per returned data set.

### **Return data set and get another**

The PUT processing occurs when the application thread subsequently issues a following IEFSSREQ macro to select another data set. You can use fields in the optional disposition section of the SSS2 to change certain attributes of the previously obtained data set from the prior IEFSSREQ call.

## 5.5 IP PrintWay extended mode SAPI processing

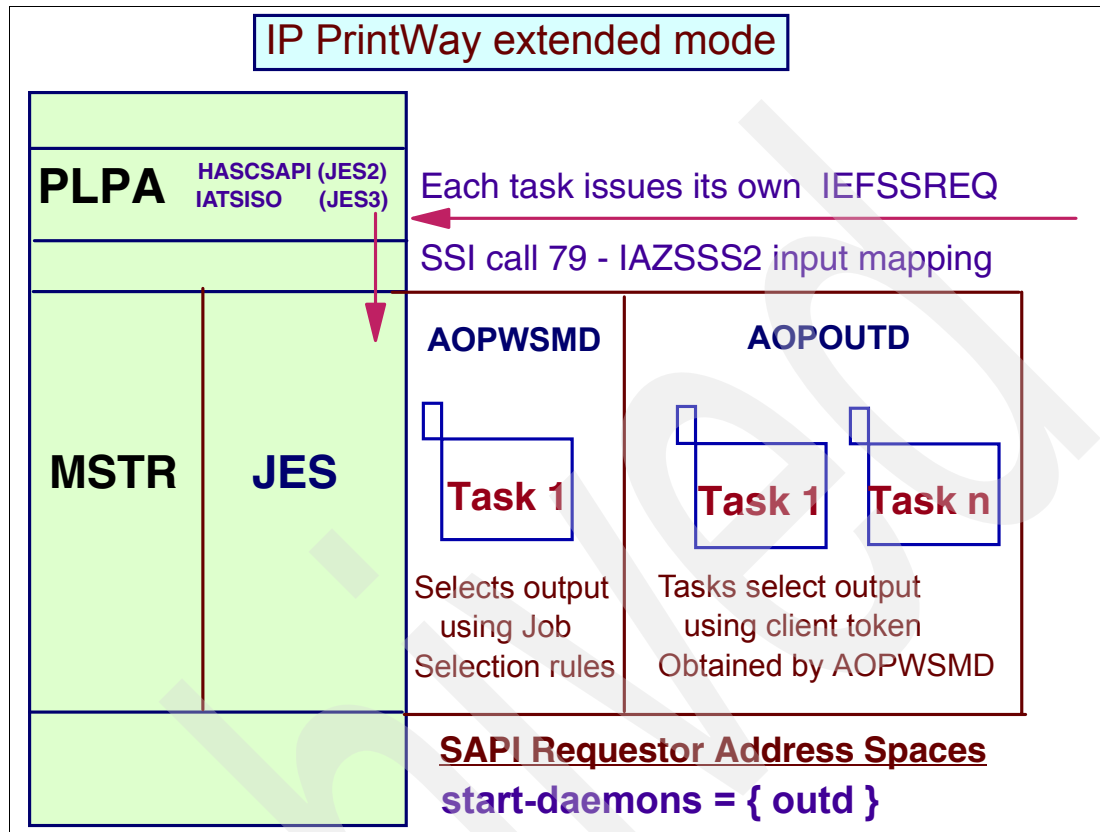


Figure 5-5 IP PrintWay extended mode SAPI requestors

### IP PrintWay extended mode SAPI requestors

The SAPI daemons for IP PrintWay extended mode run in z/OS UNIX address spaces called AOPWSMD and AOPOUTD. The daemons are started by the **aopstart** command when you specify in the configuration file **start-daemons=outd**.

The presence of **outd** indicates to the startup task to start the IP PrintWay extended mode daemons:

- aopwsmd** aopwsmd is the workselection client that manages and starts the work selection elements for getting jobs from the JES spool through the SAPI interface. aopwsmd uses the IP PrintWay job selection rules in the Printer Inventory to select output data sets from the JES spool.
- aopoutd** aopoutd is the output server that starts the output device threads and manages the queues for the output devices. aopoutd sends output to remote printers in the TCP/IP network and to e-mail destinations.

## 5.6 IP PrintWay extended mode printing

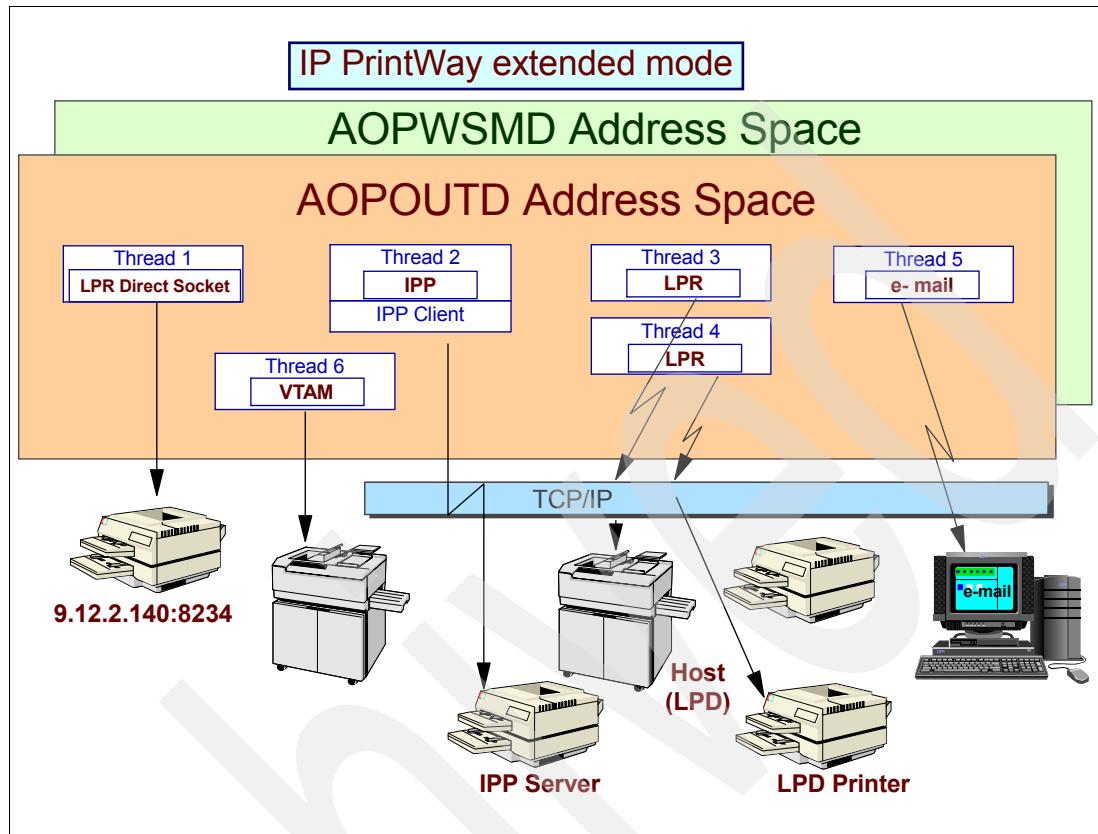


Figure 5-6 Printing with IP PrintWay extended mode

### Printing with IP PrintWay extended mode

The start-daemons = { outd } aopd.config file specification starts the major components (daemons) of IP PrintWay extended mode:

**aopwsmd** An IP PrintWay extended mode daemon. It uses the IP PrintWay job selection rules in the Printer Inventory to select output data sets from the JES spool.

**aopoutd** An IP PrintWay extended mode daemon. It sends output to remote printers in the TCP/IP network and to e-mail destinations. The aopoutd daemon runs in an z/OS UNIX address space called AOPOUTD. The major functions of the aopoutd daemon include:

- ▶ JobReceiverManager - There can be several tasks (threads) called JobReceiver.
- ▶ OutputServer - There can be many OutputQueue tasks (that creates an OutputDevice). The OutputDevice is responsible for actually delivering the job to the target. It uses the correct protocol to accomplish delivery.
- ▶ PostPrintHandler - The PostPrintHandler retains or removes the job from the spool and the inventory. It updates the output-device object in the printway inventory and updates the printway-job object with the status of the job. If the job is a Print Interface job or a NetSpool job, then the job object in the printer inventory is updated with the final status of the job.

## 5.7 IP PrintWay basic mode

- ❑ **IP PrintWay Basic Mode:**
  - Uses IP PrintWay FSSs and FSAs to:
  - Output from JES spool and send them to TCP/IP-attached
    - LPR to LPD
    - Direct socket printing
    - Internet Printing Protocol (IPP)
    - VTAM-controlled printers: LU0 (DSC/DSE), LU1 (SCS), LU3 (DSC/DSE)
    - Electronic mail (e-mail) destinations
- ❑ **Administration:**
  - Through JES commands and SDSF or EJES
  - IP PrintWay Transmission Queue ISPF interface
- ❑ IP PrintWay basic mode can share printing information across systems in a sysplex
- ❑ Maximum data sets size 2 gigabytes

Figure 5-7 IP PrintWay basic mode

### IP PrintWay basic mode

IP PrintWay basic mode, the original mode of operation, uses the z/OS Functional Subsystem Interface (FSI) services to obtain output data sets from the JES spool. *IBM does not plan additional enhancements to IP PrintWay basic mode.*

The FSI services are actually JES and FSS/FSA supplied routines that allow interaction between JES and the FSS/FSA. FSI services fall into three categories:

- ▶ Communication services
- ▶ Data access services
- ▶ Control services.

### Basic mode support

The IP PrintWay basic mode component of Infoprint Server transmits output data sets from the JES spool to remote printers or print servers and to e-mail destinations, using one of these transmission protocols:

- ▶ **LPR:** The LPR protocol is a TCP/IP protocol defined by RFC 1179. An LPD that adheres to RFC 1179 must be running in the remote printer or system.
- ▶ **Direct-sockets printing:** The direct sockets printing protocol is a TCP/IP protocol in which data is transmitted directly to a designated port. The remote printer or print server must support direct sockets printing.

- ▶ **Internet Printing Protocol (IPP):** IPP is a standard TCP/IP protocol for printing over the Internet. An IPP server must be running in the remote printer or system.
- ▶ **Virtual Telecommunications Access Method (VTAM):** IP PrintWay basic mode supports printing to any printer that is defined to VTAM as LU type 0, LU type 1, or LU type 3. Supported output data streams are SNA character string (SCS) and Data Stream Compatible/Data Stream Extended (DSC/DSE). To print on VTAM-controlled printers, Infoprint Coaxial Printer Support for z/OS is required.
- ▶ **E-mail:** IP PrintWay can use the z/OS UNIX `sendmail` to send your print output to one or more e-mail addresses. IP PrintWay sends the output, which can be in any data format, as an e-mail attachment.

### IP PrintWay transmission queue and message-log

IP PrintWay basic mode keeps queue information in its transmission-queue data set and keeps messages in its message-log data set. Both data sets can be shared across multiple systems.

- ▶ The transmission-queue data set contains an entry for each data set that IP PrintWay is processing. IP PrintWay basic mode creates the entries in the transmission-queue data set. You cannot add entries to it. However, you can use the Infoprint Server ISPF panels to browse, modify, and delete entries.
- ▶ IP PrintWay basic mode writes messages that track data-set transmissions in an IP PrintWay message-log data set. The message log is organized as a circular queue. When a message extends to the end of the data set, the next message starts at the beginning. In this way, the size of the message data set remains stable, with old messages automatically overlaid by newer messages.

### Data set size with basic mode

IP PrintWay basic mode cannot transmit data sets larger than two gigabytes to a remote LPD. Also, depending on the IP PrintWay options the administrator selects in the printer definition, IP PrintWay might not be able to transmit data sets larger than two gigabytes to a remote printer that uses the IPP or direct-sockets printing protocol, or to e-mail destinations.

## 5.8 IP PrintWay basic mode FSS/FSA

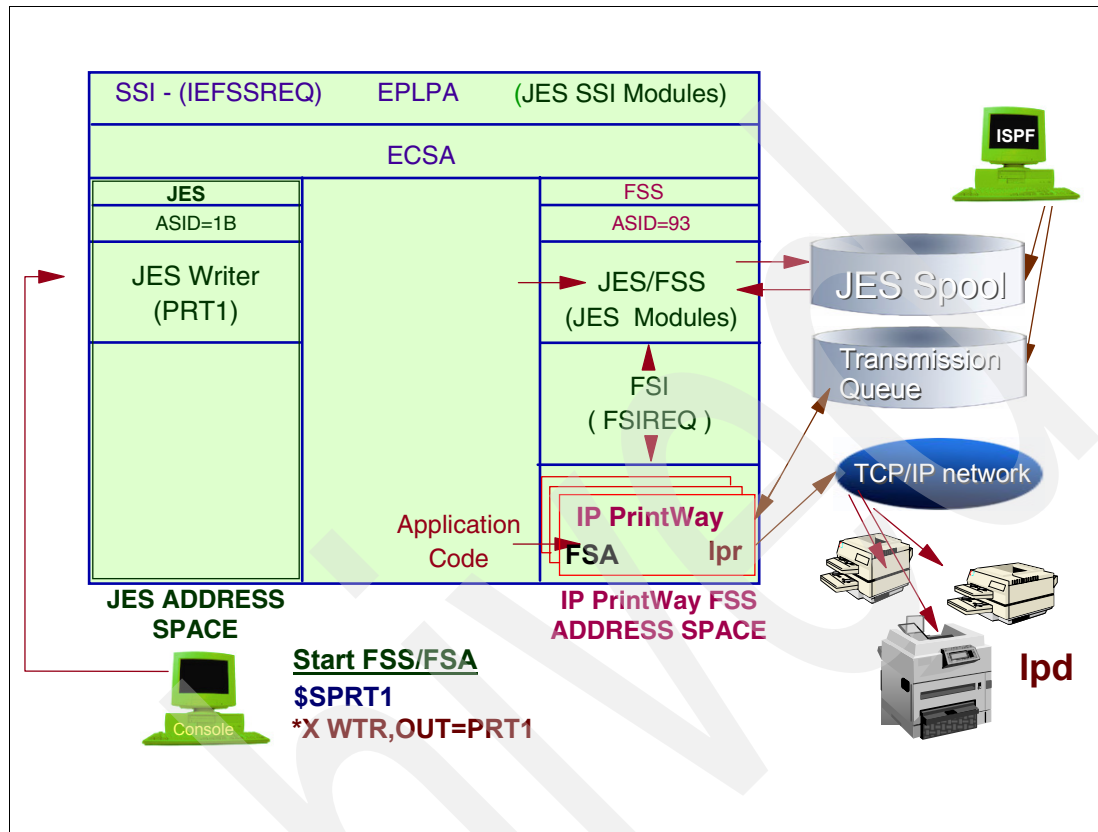


Figure 5-8 IP PrintWay basic mode FSS/FSA

### IP PrintWay basic mode FSS/FSA

A functional subsystem is an extension of JES that runs in its own address space separate from the JES address space. An FSS provides support for a function peripheral to JES processing, such as a peripheral device or other component. A functional subsystem application (FSA) executes within the functional subsystem address space and is defined to provide application-specific support to peripheral functions. An FSA allows devices to operate outside of direct JES control.

In Figure 5-8, the JES address space starts the FSS address space when an operator starts the first FSA writer. The writer is defined in the JES initialization statements and actually becomes the FSA portion of the FSS address where IP PrintWay executes.

FSI services are actually JES and FSS/FSA supplied routines that allow interaction between JES and the FSS/FSA. FSI services fall into three categories: communication, data access, and control services. These services are described in the following sections.

#### Communication services

The functions of the individual FSI communication services are:

**FSI CONNECT** The FSS and FSA invoke the FSI CONNECT service to establish the functional subsystem interface to JES. FSI CONNECT processing tells JES that the FSS/FSA is started. It also identifies to the FSI the addresses of FSS/FSA routines that are to receive control when JES

	issues the FSIREQ macro and the addresses of JES routines that are to receive control when the FSS/FSA issues the FSIREQ macro.
FSI DISCONNECT	The FSS and FSA invoke the FSI DISCONNECT service to terminate connection with JES.
FSI ORDER	JES invokes the FSI ORDER service to issue orders to the FSS/FSA. When an operator issues a JES command that requires the participation of an FSS/FSA, JES converts that command into an order. An order represents a unit of work known to both JES and the FSS/FSA. The FSS/FSA performs the actions associated with the order and then responds to JES with the required information.

### ***Data access services***

The functions of the individual FSI data access services are:

FSI GETDS	The FSA invokes the FSI GETDS service to request access to a JES spool data set and its characteristics. The GETDS service is functionally equivalent to allocating and opening a data set.
FSI GETREC	The FSA invokes the FSI GETREC service to obtain one or more records from a data set obtained by use of the FSI GETDS service.
FSI FREEREC	The FSA invokes the FSI FREEREC service to free one or more logical records that it previously acquired with a GETREC request.
FSI RELDS	The FSA invokes the FSI RELDS service to release a data set previously obtained by the FSI GETDS service. The RELDS service is functionally equivalent to closing and unallocating a data set.
FSI CHKPT	The FSA invokes the FSI CHKPT service to request JES to record checkpoint information for the JES spool data set currently being processed on the FSA device.

### ***Control services***

The FSI POST service is the only FSI control service. JES invokes the FSI POST service to signal completion of asynchronous requests.

### **FSS Startup processing**

JES starts the FSS either during JES initialization or in response to an operator command to start a printer under control of the FSS. JES gets information from the FSS initialization statement to use in the MVS START command. JES then issues the MVS START command causing the creation of the FSS address space.

Once the FSS address space is created, the FSS performs initialization. When initialization is complete, the FSS responds to JES with an FSIREQ CONNECT request. Successful completion of FSS CONNECT processing signals JES to issue a START FSA order. JES issues the START FSA order to the FSS order routine. The FSS order routine receives the order and then the FSS attaches an FSA task to perform FSA and device initialization. When FSA initialization is complete, the FSA responds to JES with an FSIREQ CONNECT request. Successful completion of FSA CONNECT processing signals JES to issue a START DEVICE order.

JES issues the START DEVICE order to the FSA order routine. The START DEVICE order indicates to the FSA that JES is ready to receive GETDS requests. The FSIREQ SEND request notifies JES that the FSA has completed the order. At this point, the FSA can issue GETDS requests.



## **FSI data set processing**

The FSA issues an FSIREQ GETDS request to JES to obtain a JES spool data set and its attributes for processing. The GETDS service is functionally equivalent to allocating and opening a data set. If work is available, JES immediately satisfies the GETDS request. JES assigns a data set to the FSA and returns data set related information in the GETDS parameter list.

Once the FSA has obtained access to a SYSOUT data set, it uses the data set identifier returned to issue a FSIREQ GETREC request to JES to obtain logical records for the data set. When JES receives the GETREC request, it obtains one or more logical record pointers using an index table. JES then returns a pointer to the index in the GETREC parameter list to the FSA.

The FSA processes the records associated with the index and then issues a FSIREQ FREEREC request to release the storage associated with these logical records. Storage resources are a fixed quantity. After all of the records in a data set have been processed or when end-of-file is reached, the FSA issues a FSIREQ RELDS request to return the data set to JES.

When JES receives the RELDS request, it closes the data set and deallocates the storage resources associated with it. If the FSA indicated that valid checkpoint information exists for the data set, JES writes the final checkpoint record to spool. JES then waits for the next GETDS request.

## **FSS shutdown**

When an operator issues a command to either drain a specific device or to shut down JES cleanly, JES issues a STOP device order to the FSA order routine for the FSA controlling that device.

The FSA order routine processes the order and the appropriate FSA task stops the printer device. When the printer is stopped, the FSA must issue an FSIREQ SEND request to notify JES that the FSA has completed the order and that the printer is stopped. At this point, JES can pass another order to the FSA. After the FSA notifies JES that a device was stopped, JES issues a STOP FSA order to the FSS order routine.

The STOP FSA order causes the FSA to perform cleanup processing and then terminate itself by issuing an FSA-level DISCONNECT to JES. When JES receives the FSA-level DISCONNECT it validates the information and then issues a message to the operator. An FSS receives a STOP FSA order for every active FSA that it controls. After all active FSAs are stopped, JES issues the STOP FSS order to the FSS order routine.

The STOP FSS order causes the FSS to perform cleanup processing and then terminate itself by issuing an FSS-level DISCONNECT to JES. JES validates the FSS information and terminates the FSS address space.

## 5.9 IP PrintWay basic mode printing

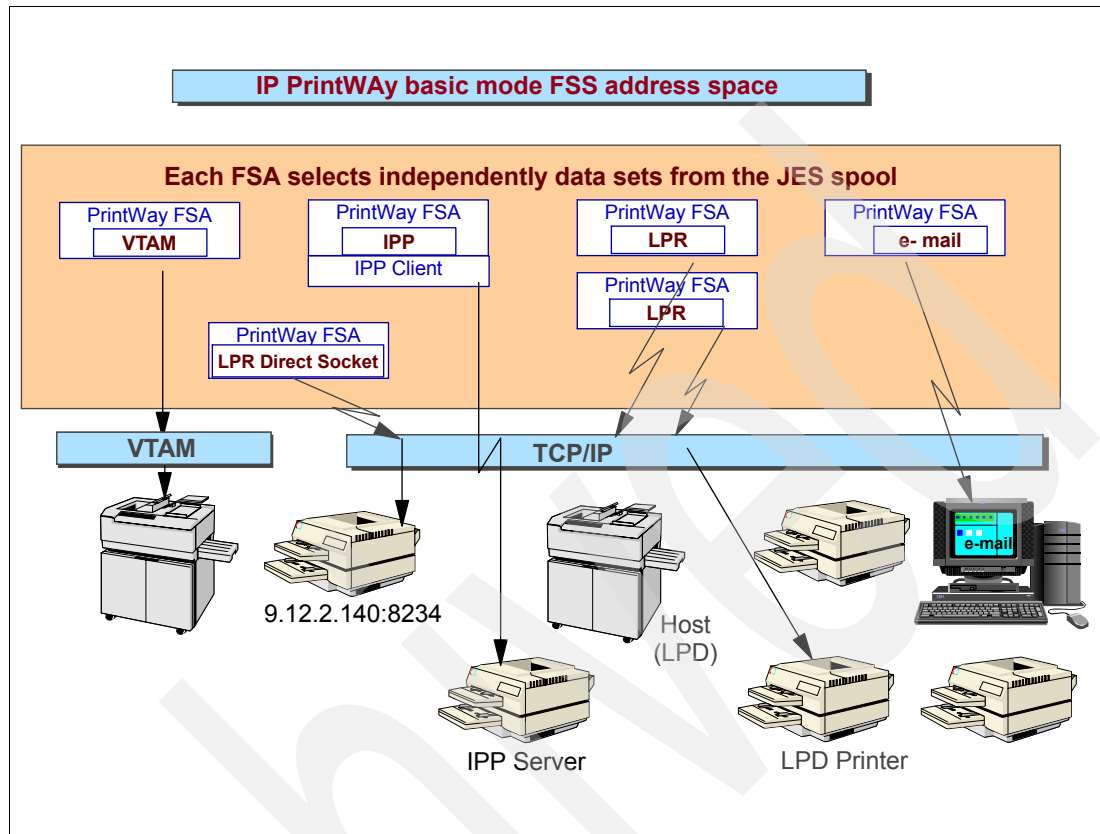


Figure 5-9 Printing with IP PrintWay basic mode

### Printing with IP PrintWay basic mode

In an IP PrintWay printer definition, you can select the transmission protocol that IP PrintWay uses to transmit output data sets from the JES spool to the target system. The target system can be a printer, a print server, or an e-mail destination. IP PrintWay basic mode supports the following protocols:

- ▶ LPR
- ▶ Direct sockets
- ▶ IPP
- ▶ e-mail
- ▶ VTAM

**Note:** Each of the FSAs can print *any* type of data sets.

## 5.10 One IP PrintWay basic mode FSS/JES and several FSAs

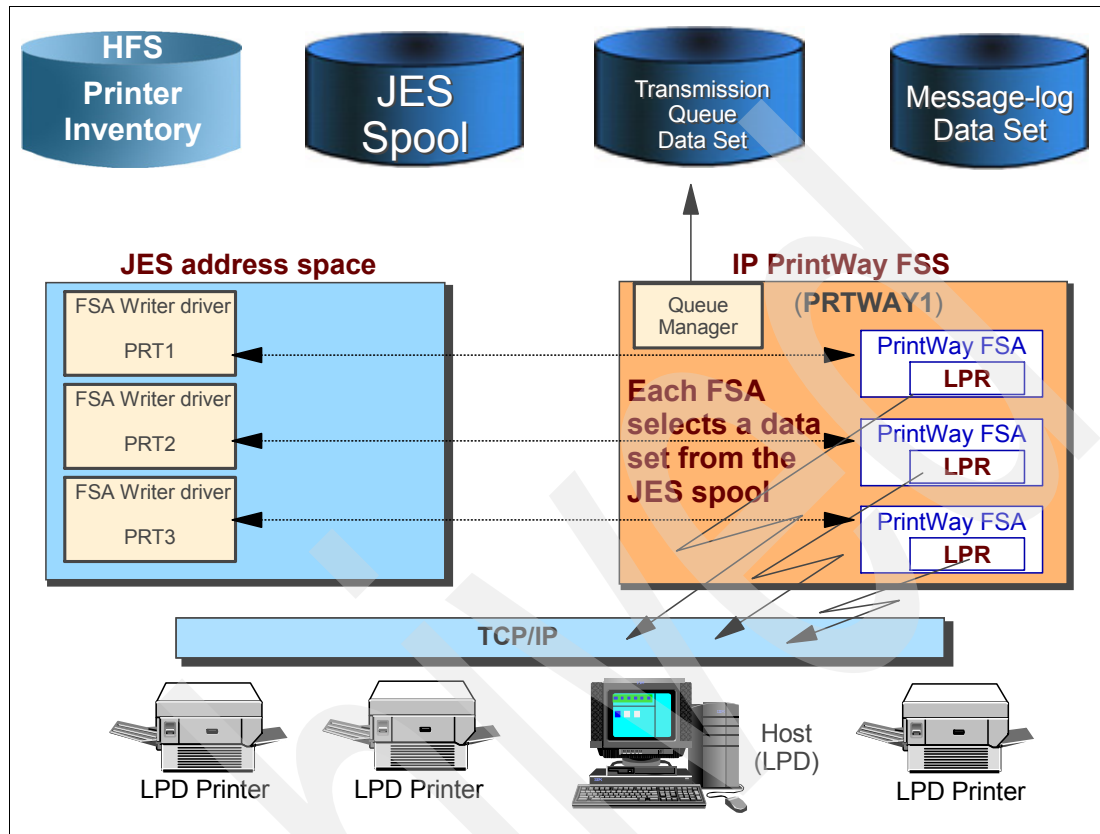


Figure 5-10 One IP PrintWay basic mode FSS and JES

### One IP PrintWay basic mode FSS and JES

Several IP PrintWay FSAs can run in the same FSS address space. Figure 5-10 shows three FSAs that have been started in the FSS named PRTWAY1. Each FSA, once started, selects a data set to print by requesting the data set from JES. The data set is then read from the JES spool by the FSA running in the FSS address space and sent to the destination. IP PrintWay sends data to printers through TCP/IP.

IP Printway keeps track of the work in process in the transmission queue data set. The IP Printway transmission queue is a VSAM cluster and contains an entry for each data set that IP PrintWay is transmitting to the remote systems or that IP PrintWay is retaining on the JES spool.

Only one IP PrintWay FSS is required. However, for improved throughput and more efficient use of system resources up to 2000 FSSs can be defined. If you define more than one IP PrintWay FSS, specify the same transmission queue data set in the startup procedure for each FSS to ensure that data sets are always transmitted in the same order as IP PrintWay acquires them from the JES spool.

## 5.11 Multiple IP PrintWay basic mode FSS address spaces

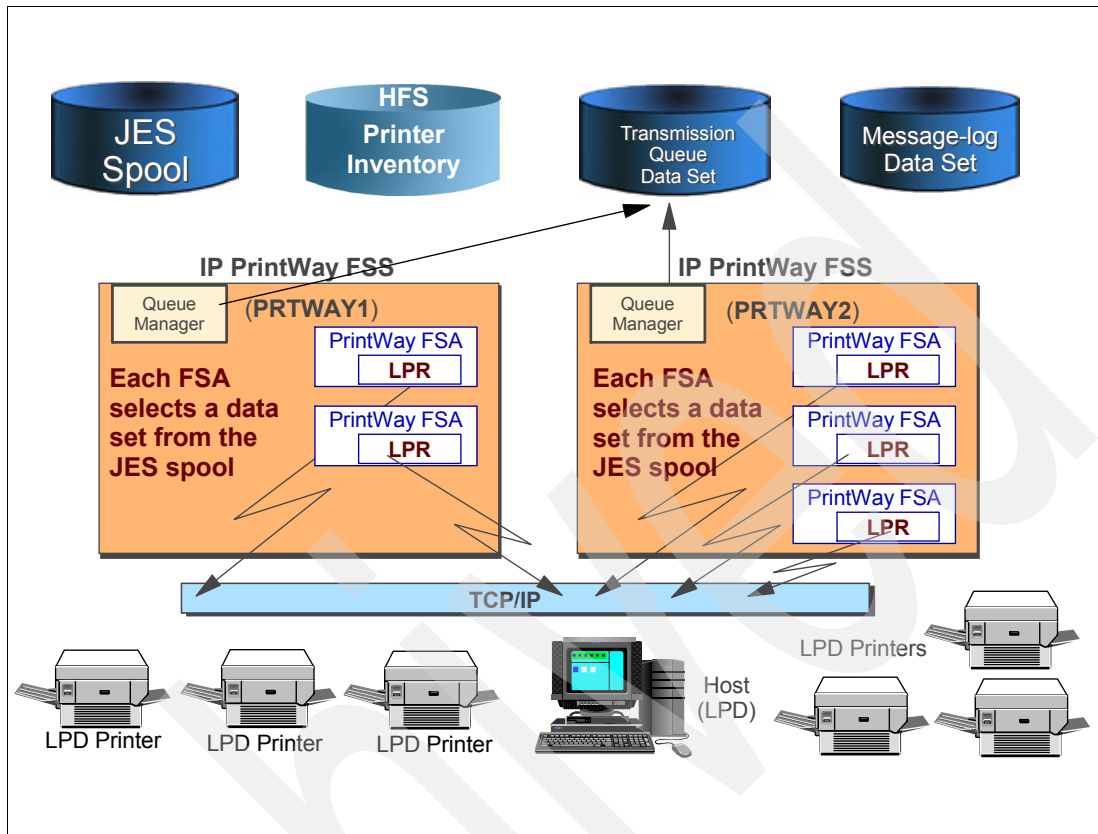


Figure 5-11 Multiple IP PrintWay FSS address spaces

### Multiple IP PrintWay FSS address spaces

When multiple IP PrintWay FSSs share a transmission queue data set, as shown in Figure 5-11, you must allocate the data set with SHAREOPTIONS (4 3). When you define the FSSs in this way, each FSS reads all of the transmission queue entries and ensures that data sets are sent in the right order.

Each IP PrintWay FSA can transmit data sets, one data set at a time, to many printers or print servers. When an FSA selects a data set from the JES spool, the IP address determines which printer the FSA passes the data set to.

For improved throughput and more efficient use of system resources, you should define more than one FSA for each IP Printway FSS. IBM recommends that you do not create more than 35 FSAs per FSS. You can define up to 64 FSAs in an FSS. As a guideline for the number of FSAs, you should estimate how many jobs must print concurrently and specify at least that many FSAs.

## 5.12 TCP/IP customization tasks for IP PrintWay basic mode

- ❑ **Customize z/OS Communications Server**
  - **TCP/IP TCPIP.DATA statements**
    - LOOKUP
    - TCPIPJOBNAME
    - DATASETPREFIX
    - SYSTCPD DD statement for TCPIP.DATA data set
    - IP PrintWay jobname as HLQ of TCPIP.DATA data set
  - **PROFILE.TCPIP data set**
  - **z/OS UNIX sendmail**
- ❑ **IP PrintWay FSS startup procedure - SYS1.PROCLIB(ANFWPROC)**

Figure 5-12 TCP/IP customization tasks for IP PrintWay basic mode

### z/OS Communications Server customization

To print on TCP/IP-attached printers or to send output to e-mail destinations, you must customize the TCP/IP component of z/OS Communications Server. The TCP/IP TCPIP.DATA data set contains configuration information for TCP/IP. This topic describes only those statements that have special consideration for IP PrintWay basic mode or IP PrintWay extended mode:

#### LOOKUP

Specifies the order in which TCP/IP uses the DNS and the local host files to resolve host names. We recommend that to significantly improve performance when you run IP PrintWay extended mode, specify: LOOKUP DNS

**Note:** The LOOKUP TCPIP.DATA statement primarily affects IP PrintWay extended mode.

#### TCPIPJOBNAME

Specifies the name of the TCP/IP program that IP PrintWay basic mode uses. You can specify the TCP/IP name either in this statement or in the IP PrintWay basic mode FSS definition in the Printer Inventory. If you do not specify a name in either this statement or the FSS definition, the name of the TCP/IP program must be TCPIP. If you specify a TCP/IP name in both places, IP PrintWay basic mode uses the name in the FSS definition.

**DATASETPREFIX** Specifies the high-level qualifier for TCP/IP data sets. IP PrintWay basic mode uses this high-level qualifier when searching for TCP/IP translation tables.

### TCPIP.DATA data set

For IP PrintWay basic mode you can create a unique TCPIP.DATA data set for use by IP PrintWay basic mode. If you do this, you must either:

1. Identify that data set in the IP PrintWay basic mode startup procedure with a SYSTCPD DD statement, or
2. Use the jobname of the IP PrintWay basic mode startup procedure as the high-level qualifier of the TCPIP.DATA data set name. To find the TCPIP.DATA data set to use, IP PrintWay basic mode follows this search sequence. IP PrintWay basic mode reads each instance of TCPIP.DATA it can find, in order, until it finds values for each TCP/IP configuration statement (TCPIPDATA and DATASETPREFIX). It uses the first value it finds for each statement. If an allocation fails, the data set does not exist, or the data set is not available, IP PrintWay basic mode searches the next data set in the following sequence:
  - a. The data set that the SYSTCPD DD statement refers to in the IP PrintWay basic mode startup procedure
  - b. *jobname*.TCPIP.DATA  
*jobname* is the job name of the IP PrintWay basic mode startup procedure.
  - c. SYS1.TCPPARMS(TCPDATA)
  - d. *tcpnlq*.TCPIP.DATA  
*tcpnlq* is the value in a DATASETPREFIX statement found earlier in the search sequence or the system default value of TCPIP.

### Customizing the PROFILE.TCPIP data set

Print Interface, IP PrintWay extended mode, and the SNMP subagent support IPv6. If you enable or disable IPv6 while Infoprint Server daemons are running, you must stop and restart the LPD (aoplpd), the IPP Server daemon (aopippd), the IP PrintWay extended mode daemons (aopoutd and aopwsmd), and the SNMP subagent daemon (aopsnmpd).

Special considerations exist for Print Interface and IP PrintWay for these statements: TCPCONFIG, PORT, and AUTOLOG.

### Customizing z/OS UNIX sendmail

When the e-mail protocol is selected in a printer definition, IP PrintWay uses z/OS UNIX sendmail to prepare and send e-mails to the recipients listed in the printer definition. Sendmail is a mail transfer agent provided with z/OS Communications Server that provides enhanced SMTP support.

If you have not already customized z/OS UNIX sendmail, you must do so before IP PrintWay can send output to e-mail destinations. For information about how to customize sendmail, see:

- *z/OS Communications Server: IP Configuration Guide* describes how to customize sendmail and the other components of z/OS Communications Server that sendmail requires.
- The industry-accepted book *Sendmail*, published by O'Reilly & Associates, Inc. (ISBN 1-56592-222-0), describes the sendmail configuration file (*sendmail.cf*), how to set up sendmail aliases files, how to ensure security, and how to debug problems with your sendmail configuration.

- [www.sendmail.org](http://www.sendmail.org), a Web site maintained by the Sendmail Consortium as a resource for the freeware version of sendmail.

### IP PrintWay FSS startup procedure (basic mode)

IP PrintWay provides a startup procedure in SYS1.PROCLIB(ANFWPROC). You can use the same startup procedure to start more than one IP PrintWay FSS, but only if you want all IP PrintWay FSSs to share the same IP PrintWay transmission-queue and message-log data sets. You must specify the name of this startup procedure in the JES initialization statement for each FSS.

The SYS1.PROCLIB(ANFWPROC):

```
//ANFWPROC PROC  HLQ=ANF,
//      TCPHLQ=TCPIP
//* -----
/* this is a sample procedure for starting IP PrintWay
/* -----
//IEFPROC  EXEC  PGM=ANFFIEP,REGION=4M,TIME=NOLIMIT,PARM='INV=AOP1'
//*-----
/* AOP1 is the default name of the printer inventory
/* server. This must be modified if this is not the
/* name being used for the inventory server
/*-----
//ANFQUEUE DD DSN=&HLQ..QUEUE,DISP=SHR
//SYSTCPD  DD DSN=&TCPHLQ..TCPIP.DATA,DISP=SHR
//ANFMMSG  DD DSN=&HLQ..MSGFILE,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSABEND DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
```

## 5.13 IP PrintWay basic mode customization steps

- ❑ **APF-authorize SYS1.IMAGELIB**
  - **Link-list and APF authorize SYS1.IMAGELIB**
- ❑ **Define the IP PrintWay functional subsystem - (FSS)**
  - **Define one or more FSAs**
  - **Multiple FSS address spaces can be defined**
- ❑ **Create FSS and FSA definitions in the Printer Inventory**
  - **The FSS and FSAs can be defined using ISPF**
- ❑ **Customize the IP PrintWay startup procedure**
- ❑ **Customize PSF for sharing network printers**

Figure 5-13 Customizing the functional subsystem for basic mode

### APF-authorizing SYS1.IMAGELIB (basic mode)

If you plan to use FCB support in IP PrintWay basic mode, you must link-list and APF authorize the SYS1.IMAGELIB data set. To temporarily authorize this data set, use the SETPROG APF operator command. To permanently authorize this data set:

- ▶ Update the PROGxx member of SYS1.PARMLIB to include the data sets in the LNKLIST statement. If the LNKAUTH parameter that is put in the IEASYSxx member of SYS1.PARMLIB does not specify LNKLIST, also add the data sets to the APF statement in the PROGxx member.
- ▶ As an alternative to updating the PROGxx member, you can update the LNKLISTxx and IEAAPFxx members of SYS1.PARMLIB. However, IBM recommends updating the PROGxx member.

### Defining the IP PrintWay functional subsystem (basic mode)

IP PrintWay basic mode operates as a JES functional subsystem (FSS).

In the IP PrintWay FSS, you define one or more IP PrintWay functional subsystem applications (FSAs), which use the support facilities of the FSS to communicate with JES. Several IP PrintWay FSAs can run in the same FSS.

**Note:** There is no practical limit to the number of FSSs that you can define, nor is there a limit to the number of FSAs that you can define to run under an individual FSS.



You can define one or more IP PrintWay functional subsystem applications (FSA) in the JES initialization statements for a IP Printway FSS.

## Creating FSS and FSA definitions in the Printer Inventory (basic mode)

You can specify IP PrintWay basic mode configuration information in these objects in the Printer Inventory:

### FSS definition

IP PrintWay uses values specified in this definition when the IP PrintWay FSS starts. In an IP PrintWay FSS definition, you can specify configuration information that applies to an IP PrintWay FSS and to all FSAs in that FSS. An IP PrintWay FSS definition is required in these situations:

- ▶ If any of the FSAs in the FSS will transmit output to VTAM-controlled printers. In this case, you must specify the name of the VTAM APPL statement you defined for the FSS.
- ▶ If you need to change the IP PrintWay configuration defaults. If an IP PrintWay FSS definition does not exist when IP PrintWay starts, IP PrintWay uses default values.

### FSS example

Figure 5-14 shows an Infoprint Server ISPF IP PrintWay FSS browse panel.

```
AOIPSA1                                IP PrintWay FSS
Command ==>

FSS name. . . PRINTWAY
Description . JES3 PrintWAY FSS in SC65                                (extend)

    Old-style translation
    Hiperspace blocks . . 524288
    TCP/IP job name . . . TCPIP
    Document code page. .
    Applid. . . . . SC65PW01
    National language . . 1    1. English  2. Japanese
    Trace mode. . . . . 1    1. None    2. Internal  3. No printing  4. Full
        Trace prompt
        Trace table size . 32
```

Figure 5-14 IP PrintWay FSA ISPF panel

### FSA definition

IP PrintWay uses values specified in this definition when an IP PrintWay FSA starts.

You can create one IP PrintWay FSA definition for each IP PrintWay FSA that your installation has defined to JES. In each IP PrintWay FSA definition you can specify the type of tracing to start for the FSA.

Creating FSA definitions is optional. If an FSA definition does not exist, IP PrintWay uses the tracing value specified in the FSS definition. The FSA name must match the name of the FSA as defined to JES.

You can use either the Infoprint Server ISPF panels or the Printer Inventory Definition Utility (pidu) to create FSS and FSA definitions.

## FSA example

Figure 5-15 shows an Infoprint Server ISPF IP PrintWay FSA browse panel.

AOIPAAI	IP PrintWay FSA
Command ==>	
FSA Name. . . PRTWAY	
Description . IP PrintWay FSS/FSA	(extend)
Trace mode . . 1 1. None 2. Internal 3. No printing 4. Full	

Figure 5-15 IP PrintWay FSA ISPF panel

## Customizing the IP PrintWay startup procedure (basic mode)

To start an IP PrintWay basic mode FSS, you must use a cataloged startup procedure. The startup procedure identifies:

- ▶ The program name, region size, and time-out value
- ▶ Name of the Printer Inventory
- ▶ Location of the IP PrintWay transmission-queue and message-log data sets
- ▶ Environment variables

IP PrintWay provides a basic mode FSS startup procedure in SYS1.PROCLIB(ANFWPROC).

## Customizing PSF for sharing network printers

PSF is a separately orderable product that runs on z/OS and controls printing on AFP printers. When customizing PSF to send AFP data to IBM network printers, be sure to configure PSF for printer sharing if you also want to print on these printers from IP PrintWay. When configured for printer sharing, IP PrintWay can also send non-AFP data streams to the printer.

If you do not configure PSF for printer sharing, PSF does not release the printer to receive data from other sources.

## 5.14 ANFWPROC - IP PrintWay basic mode startup procedure

- ❑ Startup procedure in SYS1.IBM.PROCLIB(ANFWPROC)
  - Define to RACF STARTED class

```
//ANFWPROC PROC    HLQ=ANF ,
//                TCPHLQ=TCPIP
//* -----
/** this is a sample procedure for starting IP PrintWay
/** -----
//IEFPROC EXEC PGM=ANFFIEP,REGION=4M,TIME=NOLIMIT,PARM='INV=AOP1'
//*-----
/** AOP1 is the default name of the printer inventory
/** server. This must be modified if this is not the
/** name being used for the inventory server
//*-----
//ANFQUEUE DD DSN=&HLQ..QUEUE,DISP=SHR
//SYSTCPD  DD DSN=&TCPHLQ..TCPIP.DATA,DISP=SHR
//ANFMMSG  DD DSN=&HLQ..MSGFILE,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSABEND DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
```

Figure 5-16 ANFWPROC - IP PrintWay basic mode startup procedure

### ANFWPROC - IP PrintWay basic mode startup procedure

IP PrintWay provides a startup procedure in SYS1.IBM.PROCLIB(ANFWPROC). You can use the same startup procedure to start more than one IP PrintWay FSS but only if you want all IP PrintWay FSSs to share the same IP PrintWay transmission-queue and message-log data sets. You must specify the name of this startup procedure in the JES initialization statement for each FSS. You can use the same startup procedure to start more than one IP PrintWay FSS but only if you want all IP PrintWay FSSs to share the same IP PrintWay transmission-queue and message-log data sets. You must specify the name of this startup procedure in the JES initialization statement for each FSS.

### Editing the procedure

If the high-level qualifier for the TCP/IP data sets is not TCPIP, change the TCPHLQ parameter on the PROC statement.

If the Printer Inventory name is not the default name, AOP1, specify the inventory name in the PARM parameter of the EXEC statement.

If you run many FSAs in the IP PrintWay FSS, you might need to increase the size in the REGION parameter of the EXEC statement.

If you do not use the default names for the IP PrintWay transmission-queue and message-log data sets, change the data set names in the procedure.

## 5.15 RACF for IP PrintWay basic mode startup procedure

- ❑ Create a user ID to assign to the IP PrintWay startup procedure

```
ADDUSER AOPSTC OMVS(UID(uid ) HOME('/u/aopstc')  
PROGRAM('/bin/sh')) DFLTGRP(AOPOPER) NOPASSWORD
```

- ❑ Give the user ID RACF update access to the IP PrintWay transmission-queue, message-log data sets, and TCPIP.DATA

```
PERMIT ANF.QUEUE ACCESS(UPDATE) ID(AOPSTC)  
PERMIT ANF.MESSAGE ACCESS(UPDATE) ID(AOPSTC)  
PERMIT SYS1.TCPPARMS ACCESS(READ) ID(AOPSTC)
```

- ❑ Defining the IP PrintWay startup procedure to RACF

```
RDEFINE STARTED ANFWPROC.* STDATA(USER(AOPSTC) GROUP(AOPOPER))  
SETROPTS RACLIST(STARTED) REFRESH
```

Figure 5-17 IP PrintWay basic mode startup procedure

### RACF definitions for the IP PrintWay basic mode startup procedure

The user ID assigned to the IP PrintWay startup procedure must be defined to RACF with access to z/OS UNIX.

Create a user ID with an OMVS segment, a home directory, and a default group that has an OMVS segment with a group identifier (GID).

You can select any name for the user ID. Use the RACF **ADDUSER** command to create a new user ID. Use the RACF **ALTUSER** command to alter an existing user ID to give it access to z/OS UNIX. For the default group, you can select any group that has an OMVS segment and a GID. The user ID must be connected to the default group.

For example, these RACF commands define user AOPSTC with default group AOPOPER. For user-identifier, specify an integer that is different from other UIDs in your installation. The NOPASSWORD operand defines the user ID as a protected user ID. Protected user IDs can not log on to the z/OS system, and they can not be revoked due to incorrect password attempts.

```
ADDUSER AOPSTC OMVS(UID(user-identifier) HOME('/u/aopstc')  
PROGRAM('/bin/sh')) DFLTGRP(AOPOPER) NOPASSWORD
```

### Access to data sets

Give the user ID RACF update access to the IP PrintWay transmission-queue and message-log data sets. Also, give the user ID RACF read access to the TCP/IP TCPIP.DATA

data set. These RACF commands give user ID AOPSTC access to data sets ANF.QUEUE, ANF.MSGFILE, and TCPIP.TCPIP.DATA:

```
PERMIT ANF.QUEUE ACCESS(UPDATE) ID(AOPSTC)
PERMIT ANF.MESSAGE ACCESS(UPDATE) ID(AOPSTC)
PERMIT SYS1.TCPPARMS ACCESS(READ) ID(AOPSTC)
```

### **RACF STARTED class**

You must define a profile for the IP PrintWay startup procedure in the RACF STARTED class. These RACF commands define a profile for the ANFWPROC procedure. The following example assumes that the user ID you defined for use with this procedure is AOPSTC and that AOPSTC is connected to group AOPOPER.

```
RDEFINE STARTED ANFWPROC.* STDATA(USER(AOPSTC) GROUP(AOPOPER))
SETROPTS RACLIST(STARTED) REFRESH
```

## 5.16 IP PrintWay transmission-queue and message-log

### ❑ IP PrintWay basic mode

- Allocate and initialize the transmission-queue data set
  - Restrict access to the transmission-queue data set
  - Multiple FSS address spaces can share
  - Specify SHAREOPTIONS (4 3)
  - Sample JCL is in SYS1.SAMPLIB ANFDEAL and ANFQINIT
- Allocate and initialize IP PrintWay message-log data set
  - Message exit (ANFUXMSG) to suppress messages
  - Sample JCL is in SYS1.SAMPLIB(ANFMIJCL)



Figure 5-18 Customizing IP PrintWay basic mode

### Allocating and initializing the transmission-queue data set (basic mode)

The transmission-queue data set is a required data set that contains an entry for each data set that IP PrintWay basic mode is processing. IP PrintWay maintains the entries in the transmission-queue data set. You cannot add entries to it. However, you can use the Infoprint Server ISPF panels to browse, modify, and delete entries.

You can allocate one transmission-queue data set to be shared by all IP PrintWay functional subsystems (FSSs). Or, you can create a separate data set for use by each IP PrintWay FSS. If more than one FSS can transmit output data sets to the same print queue, IBM recommends sharing the transmission-queue data set so that data sets are printed in the correct order. The transmission-queue data set can be shared with other instances of IP PrintWay running on other z/OS systems.

Sample JCL to allocate the transmission-queue data set is in SYS1.SAMPLIB(ANFDEAL). The sample JCL deletes and allocates the VSAM transmission-queue data set needed for IP PrintWay.

The sample SYS1.SAMPLIB(ANFQINIT) job initializes the VSAM transmission-queue data set.

**Note:** When multiple FSSs share a transmission-queue data set, you must allocate the data set with **SHAREOPTIONS (4 3)**.

### **Restricting access to the transmission-queue data set (basic mode)**

All ISPF users can use the Infoprint Server ISPF panels to update the IP PrintWay basic mode transmission-queue data set, even if you have established a RACF administration group for the Printer Inventory. You can, however, use Resource Access Control Facility (RACF) or a similar program to restrict access to the data set.

If you have RACF protected the transmission-queue data set, the Infoprint Server ISPF panels display only the actions that a user is allowed to do. For example, if a user has read-only access to the transmission-queue data set, that user is allowed only to browse entries in the data set and is not allowed to modify or delete entries.

### **Allocating IP PrintWay message-log data set (basic mode)**

IP PrintWay basic mode writes messages that track data-set transmissions in an IP PrintWay message-log data set. This is a required data set. The message log is organized as a circular queue. When a message extends to the end of the data set, the next message starts at the beginning. In this way, the size of the message data set remains stable, with old messages automatically overlaid by newer messages.

You can create one message-log data set to be shared by all IP PrintWay functional subsystems (FSSs), or you can create separate data sets for each FSS. To allocate space for the data set and initialize it, use the sample JCL in SYS1.SAMPLIB(ANFMJCL) to run the ANFMFILE program. The message-log data set requires these specifications:

- ▶ Physical sequential unmovable organization (PSU).
- ▶ Fixed block format (FBA).
- ▶ Logical record length (LRECL) of 57 at a minimum. 80 is recommended.
- ▶ Block size (BLKSIZE) of about 10 records per block. 800 is recommended.

Secondary space is unnecessary because the ANFMFILE program formats all available space. To change the size of the IP PrintWay message-log data set, rerun the ANFMFILE program. The ANFMFILE program produces a D37 abend when it completes processing.

You can suppress selected messages that IP PrintWay basic mode writes to the message-log data set by writing a Message exit (ANFUXMSG). You can view messages in the message-log data set by using the Infoprint Server ISPF panels.

## 5.17 View IP Printway basic mode transmission-queue

- 
- Transmission Queue Data Set

```
A0PIPP1                               Infoprint Server: Printer Inventory Manager
Option ==> 5_

Printer Definitions
  1 Add                               Add a printer definition
  2 List                               List printer definitions
  3 Select                             Select printer definitions to list

Other Functions
  4 Other Definitions                 Manage FSS, FSA, pool, and job selection definitions
  5 PrintWay Queue                   View IP PrintWay basic mode transmission queue
  6 PrintWay Message                  View IP PrintWay basic mode message log
  7 Configure                         Change panel configuration
```

Figure 5-19 Using the transmission-queue

## Browsing the transmission-queue

You must display a list of transmission-queue entries, using Option 5 as shown in Figure 5-19, before you can do other functions, such as browsing or modifying an individual entry. You can either list all transmission-queue entries, or you can select the entries you want to list. IP PrintWay creates the entries in the transmission-queue data set.

During an ISPF session, you can work with multiple transmission-queue data sets. For example, if your installation has defined multiple IP PrintWay FSSs, you might need to access the transmission-queue data set for each FSS. To change the name of the data set during your ISPF session, change the name on the IP PrintWay Transmission Queue Selection panel, shown in Figure 5-20 on page 267.

## Use of the transmission-queue

For improved performance, put the transmission-queue data set on a volume that is not heavily used. A VSAM LISTCAT of the IP PrintWay transmission-queue data set might show the number of extents increasing over time. This is because all print jobs in the data set have ascending keys, and VSAM does not reuse or reclaim lower space (with lower keys) even though IP PrintWay deletes the print jobs from the data set. Therefore, you might need to periodically delete the transmission-queue data set and reallocate it to reclaim the space. Do this only when IP PrintWay is stopped.



## 5.18 IP PrintWay Transmission Queue Selection panel

```
ANFIPQS                      IP PrintWay Transmission Queue Selection
Command ===> _____

Enter optional display criteria below:

Printer Name:                ===> _____
or
DCF Routing Key:
  DEST                      ===> P0K45AN
  CLASS                     ===> J
  FORMS                     ===> STD

Jobname                      ===> _____
Arrival Date                 ===> _____ (YYYY/MM/DD)
Arrival Time                 ===> _____ (HH:MM:SS)
Transmission Status         ===> _____ (One or more of: A S F R H Z)
Port Number                  ===> _____ (100 to 65535)
Print Q Name/Logmode        ===> _____

Host Name/IP Address/
URL/LU Name/e-mail         ===> _____

Queue Data Set name         ===> PRINTWAY.QUEUE

Change Sort Criteria        ===> _
```

Figure 5-20 IP PrintWay Transmission Queue Selection panel

### Using the selection panel

The IP PrintWay basic mode transmission-queue contains an entry for each data set that IP PrintWay basic mode has selected from the JES spool to process. Each entry contains the status of the transmission of the data set, routing information, and transmission options.

After IP PrintWay basic mode selects a data set from the JES spool for transmission, you cannot use JES commands and other spool management products (such as SDSF) to manage the data set. You must use Infoprint Server ISPF panels to monitor, reroute, retransmit, and delete data sets that IP PrintWay has selected.

IP PrintWay initially creates a queue entry for each data set to be transmitted. IP PrintWay retains the queue entry until either:

1. IP PrintWay deletes the data set from the JES spool, or
2. The IP PrintWay FSA processing the data set ends normally or abnormally and releases the data sets it is processing back to JES so that another IP PrintWay FSA can process them.

**Note:** If all IP PrintWay FSAs stop processing, the transmission-queue should be empty. If any entries do remain, delete the entries or reallocate the data set before restarting IP PrintWay.

## 5.19 IP PrintWay basic mode message log view

```
005260          9.134.109.105
005261          ebopcl
005262 2008050713493964-ANFM604I Data set:
005263          WTSCPLX9.VAIN.VAIN4.JOB43370.D000000A.LIST#AA#
005264          The data set is being released to JES
005265 2008050816065054-ANFM699I FSS: PRINTWAY The queue manager has stopped
005266 2008050816065572-ANFM004I Using message table ENU
005267 2008050816065576-ANFM600I FSS: PRINTWAY The queue manager has been started
005268 2008050816065595-ANFM1120I The host name of the system where IP Printway is
005269          running: wtsc43
005270 2008050816071020-API0850I NETSPOOL Command accepted
005271 2008050816071786-API1021I NETSPOOL NetSpool initialization is complete.
005272 2008050816071786 <<<<=====>>>>
005273          PASS
005274 2004031214350165-ANFM604I Data set:
005275          WTSCPLX9.ROGERS03.ROGERS03.JOB13701.D00000B8.PS070154
005276          The data set is being released to JES
005277 2004031214350180-ANFM700I Data set:
005278          WTSCPLX9.FADEL.FADEL.JOB13701.D00000BA.PS070156
005279          The data set has been acquired by PRTWAY
005280 2004031214424433-ANFM601I Data set:
005281          WTSCPLX9.ROGERS03.ROGERS03.JOB13701.D00000B9.PS070155
005282          The data set was successfully transmitted to host and queue:
005283          9.12.14.94
005284          PASS
```

Figure 5-21 IP PrintWay basic mode message log view

### IP PrintWay basic mode message log view

IP PrintWay basic mode writes a message to the IP PrintWay message-log data set when it receives a data set from JES, when IP PrintWay successfully or unsuccessfully transmits the data set to the target printer or e-mail destination, and when IP PrintWay releases the data set to JES. Also, IP PrintWay writes other messages to this data set, and IP PrintWay installation exits can write messages to this data set.

The messages wrap around to the beginning when the data set becomes full. The time stamp preceding each message indicates when IP PrintWay wrote the message. IP PrintWay writes a string containing equal signs (====) at the end of the last message written.

### Viewing messages

To view the messages:

1. Start an Infoprint Server ISPF session.
2. On the main Infoprint Server ISPF panel, select **7** Configure. Make sure that the Message log field contains the correct name of the IP PrintWay message log.
3. On the main Infoprint Server ISPF panel, select **6** PrintWay Message.
4. To find the latest message, press the REFRESH function key on the Infoprint Server ISPF panels, or search for a string of equal signs.

## 5.20 VTAM APPL resource definition and LE run-time

- ❑ Create a VTAM APPL resource definition
    - VTAMLST data set
    - Printing to VTAM-controlled printers
      - Infoprint Coaxial Printer Support for z/OS is required
- PRINTWAY VBUILD TYPE=APPL Application major node**  
**PWAPPL01 APPL AUTH=ACQ Lets IP PrintWay acquire printer**  
**PWAPPL02 APPL AUTH=ACQ Lets IP PrintWay acquire printer**
- ❑ Change LE run-time options
  - To modify - create a CEEUOPT object file

Figure 5-22 Customizing IP PrintWay basic mode

### Creating a VTAM APPL resource definition (basic mode)

When IP PrintWay basic mode prints to VTAM-controlled printers, it runs as a VTAM application program. Therefore, if you plan to print on VTAM-controlled printers, you must create a VTAM APPL definition statement in the VTAMLST data set for each IP PrintWay FSS. A separate APPL statement is required for each FSS.

Create the APPL statement in a new or existing application major node definition, for example:

```
PRINTWAY VBUILD TYPE=APPL Application major node
PWAPPL01 APPL AUTH=ACQ Lets IP PrintWay acquire printer
PWAPPL02 APPL AUTH=ACQ Lets IP PrintWay acquire printer
```

**AUTH=ACQ** This parameter permits IP PrintWay to initiate a session with the printer. AUTH=ACQ is required.

**Note:** After you create the APPL statement, you must specify the name of the APPL statement in the FSS definition.

To print on VTAM-controlled printers, Infoprint Coaxial Printer Support for z/OS is required.

## Changing LE run-time options (basic mode)

The run-time options for the LE environment that PrintWay establishes are:

```
POSIX(ON)
ALL31(ON)
STACK(8K,64K,ANY,KEEP)
HEAP(8K,8K,,FREE)
ANYHEAP(8K,8K)
THREADHEAP(8,4K,ANY,FREE)
BELOWHEAP(8,4K,FREE)
LIBSTACK(8,0)
ENVAR(''_CEE_ENVFILE=//DD:STDENV'')
MSGFILE(,,,'ENQ)
```

If you want to change the LE run-time options, create a CEEUOPT object file. Then create an SMPE usermod to link-edit the object file with ANFICONV.

## 5.21 Define IP PrintWay basic mode functional subsystem

- ❑ IP PrintWay basic mode operates as a JES FSS
  - Up to 2000 FSSs
- ❑ IP PrintWay FSS has one or more IP PrintWay FSAs
  - Up to 64 FSAs
- ❑ Planning decisions to consider:
  - Define at least 2 FSAs but not more than 35
  - Determine number of concurrent jobs to print
  - For "critical" printers - separate FSS/FSA
- ❑ Determining work-selection criteria - WS parameters
  - Job submitter uses IP address of printer host name
  - JES3 - Uses form names and FCB as WS parm

Figure 5-23 Define IP PrintWay basic mode functional subsystem

### Define IP PrintWay basic mode functional subsystem

You do not need to create more than one IP PrintWay FSS. However, for improved throughput and more efficient use of system resources, you can define up to 2000 FSSs. If you define more than one IP PrintWay FSS, specify the same transmission-queue data set in the startup procedure for each FSS to make sure that data sets are always transmitted in the same order as IP PrintWay acquires them from the JES spool. When you define the FSSs in this way, each FSS reads all of the transmission-queue entries and makes sure that data sets are sent in the right order.

To avoid system abends that can occur because of out-of-storage conditions:

- Define additional FSSs if you have a large workload or experience network problems that cause IP PrintWay to retain data sets on the JES spool.
- Define FSAs (within the FSS) with the same JES work-selection criteria so that they can share the same workload.
- Ask your administrator to specify no retention time for successfully transmitted data sets, and as short a retention time as is reasonable for failed data sets.

System abends can occur because JES uses storage below the 16 MB line in the IP PrintWay FSS address space for control blocks for data sets that IP PrintWay has acquired from the JES spool. Acquired data sets include data sets that IP PrintWay is currently processing, as well as data sets that IP PrintWay has retained on the JES spool. JES uses about 4K bytes of storage below the 16 MB line for each acquired data set. JES releases the storage when the data set is deleted from the JES spool.

The amount of storage below the 16 MB line that is available in each IP PrintWay FSS address space is limited. The storage limit depends on the total amount of storage below the 16 MB line that is available. If an FSS requires more storage than is available, a system abend can occur. Typically, an abend occurs when an FSS retains a very high number of data sets on the JES spool. For example, in an installation with 7 to 8 megabytes of storage below the 16 MB line, an abend can occur if an FSS retains more than 300 to 500 data sets.

## IP PrintWay FSA

One PrintWay FSA can transmit data sets to many printers or e-mail destinations. However, for improved throughput and more efficient use of system resources, you might want to define several FSAs per FSS. JES lets you define up to 64 FSAs per FSS. Consider the following:

- ▶ Define at least two IP PrintWay FSAs so that when one FSA is waiting for a printer to respond, the second IP PrintWay FSA can print another document on a different printer.
- ▶ Do not define more than 35 FSAs per FSS.
- ▶ Determine how many jobs need to print concurrently (the burst rate), and specify at least that many FSAs. For example, some installations with low print volume can drive as many as 600 printers with 5 or 6 FSAs.
- ▶ To make sure that critical printers are available for printing at all times, define a separate FSS for each critical printer. In each JES FSA definition, specify unique work-selection criteria so that the FSA selects output data sets only for that one printer. For example, the work-selection criteria could be the DEST value (or a combination of DEST and FORMS values) specified in the printer definition for the printer.

## Work selection criteria

You define work-selection criteria for each IP PrintWay FSA during JES initialization. These criteria can determine which output data sets each FSA selects from the JES spool. You specify the work-selection criteria on the WS parameter of either the JES2 PRTnnnnn statement or the JES3 DEVICE statement. JES2 and JES3 let you specify a variety of work-selection criteria, which correspond to JCL parameters. Some work-selection criteria that you might consider for data sets to be processed by IP PrintWay are: output class, form name, and destination name. Some considerations related to work-selection criteria are:

- ▶ Do not specify destination as a work-selection criterion if job submitters in your installation plan to specify the printer host name or IP address in the DEST=IP: JCL parameter or in the printer-ip-address job attribute. The reason is that JES does not use the host name or IP address specified as an OUTPUT JCL parameter when determining whether an output data set meets the work-selection criteria defined for an FSA.
- ▶ In a JES3 environment, specify form as a work-selection criterion if your installation wants IP PrintWay to use the form name specified in a JCL statement to select any printer definitions in the Printer Inventory. The reason is that JES3 passes the form name specified as a JCL parameter to the IP PrintWay FSA only if form is a JES work-selection criterion. Although you specify form as a work-selection criterion, the IP PrintWay FSA selects jobs with all form names for processing.
- ▶ In a JES3 environment, specify FCB as a work-selection criterion if you want IP PrintWay to do FCB processing. The reason is that JES3 passes the FCB name specified as a JCL parameter to the PrintWay FSA only when FCB is a JES work-selection criterion.

IP PrintWay uses printer-selection criteria to select printers. IP PrintWay uses either the printer definition name or a combination of the DEST, CLASS, and FORMS values specified in the OUTPUT JCL statement to select printers.

## 5.22 IP PrintWay basic mode default font and page definitions

- ❑ Specifying a default font and page definition to JES
  - Define a default UCS and FCB
  - IP PrintWay uses:
    - UCS name as a default font name
    - FCB name as a default page definition
- ❑ Resubmit for filtering
  - Uses defaults from JES
- ❑ Default font and page definition for data transforms
  - JES2: (1) do not specify the NIFCB parameter in the PRINTDEF statement and (2) do not specify the FCB parameter in the PRTnnnnn statement
  - JES3: (1) PDEFAULT=FCB on the DEVICE statement (2) PDEFAULT=CHARS on the DEVICE statement

Figure 5-24 Specifying font and page definitions

### Default font and page definitions

When you define an IP PrintWay FSA to JES, you can specify a default UCS and a default FCB. JES supplies the default UCS and FCB to IP PrintWay when an output data set does not specify the UCS or FCB parameter on the OUTPUT JCL statement. IP PrintWay uses the default UCS name as the default font name if the output data set does not specify the CHARS parameter on the OUTPUT JCL statement. IP PrintWay uses the default FCB name as the default page definition name (in addition to the FCB name) if the output data set does not specify the PAGEDEF parameter.

You can specify a default font and page definition to JES in these ways:

- JES2** Specify a default font in (1) the NIUCS parameter on the JES2 PRINTDEF statement or (2) the UCS parameter on the JES2 PRTnnnnn statement for the IP PrintWay FSA. If you do not specify a default font, JES2 uses GF10 as the default font.
- Specify a default page definition in (1) the NIFCB parameter on the JES2 PRINTDEF statement or (2) the FCB parameter on the JES2 PRTnnnnn statement for the IP PrintWay FSA.
- JES3** Specify a default font in (1) the CHARS parameter on the JES3 OUTSERV statement or (2) the CHARS parameter on the JES3 DEVICE statement for the IP PrintWay FSA. If you do not specify a default font, JES3 uses font GS10 as the default font.

Specify a default page definition in (1) the CARRIAGE parameter on the JES3 OUTSERV statement or (2) the CARRIAGE parameter on the JES3 DEVICE statement for the IP PrintWay FSA. JES3 uses 6 as the default FCB name. (IP PrintWay adds a P1 prefix, so the default page definition is P16.)

### **Resubmit for filtering**

When you select the **resubmit for filtering** option in a printer definition, IP PrintWay basic mode passes the default font and page definition received from JES to the AFP to PCL, AFP to PDF, and AFP to PostScript transforms. This means that, when you specify a default font and page definition to JES, the transforms do not use the font and page definition the administrator specifies in the printer definition or the font and page definition you specify in the transform configuration file.

### **Default font and page definition for data transforms**

If you want the transforms to use a font and page definition specified in a printer definition or in the transform configuration file, do this:

**JES2** To prevent JES2 from supplying a default page definition name, (1) do not specify the NIFCB parameter in the PRINTDEF statement and (2) do not specify the FCB parameter in the PRTnnnnn statement.

To prevent JES2 from supplying a default font, specify UCS=0 in the PRTnnnnn statement. When UCS=0, JES2 ignores any font specified in the NIUCS parameter.

**JES3** To prevent JES3 from supplying a default page definition name, specify PDEFAULT=FCB on the DEVICE statement.

To prevent JES3 from supplying a default font specify PDEFAULT=CHARS on the DEVICE statement.

To prevent JES3 from supplying a default font or FCB, specify PDEFAULT=(CHARS,FCB) on the DEVICE statement.



## 5.23 JES FSS/FSA definitions for IP PrintWay basic mode

### JES2 example

```
FSS(PRTWAY1),PROC=ANFWPROC  
PRT1 FSS=PRTWAY1,MODE=FSS,START=NO,TRKCELL=YES,WS=(Q),CL=J  
PRT2 FSS=PRTWAY1,MODE=FSS,START=NO,TRKCELL=YES,WS=(Q),CL=J  
PRT3 FSS=PRTWAY1,MODE=FSS,START=NO,TRKCELL=YES,WS=(Q),CL=J
```

### JES3 example

```
FSSDEF,TYPE=WTR,FSSNAME=PRTWAY1,PNAME=AFNWPROC,  
DEVICE,DTYPE=PRTAFP1,JNAME=PRT1,FSSNAME=PRTWAY1,  
JUNIT=(,SC50,UR,,,SC49,UR),MODE=FSS,WC=(J),WS=(CL)  
DEVICE,DTYPE=PRTAFP1,JNAME=PRT2,FSSNAME=PRTWAY1,  
JUNIT=(,SC50,UR,,,SC49,UR),MODE=FSS,WC=(J),WS=(CL)  
DEVICE,DTYPE=PRTAFP1,JNAME=PRT3,FSSNAME=PRTWAY1,  
JUNIT=(,SC50,UR,,,SC49,UR),MODE=FSS,WC=(J),WS=(CL)
```

#### ☐ Blank truncation = no - required for ASCII data streams

- JES2 - BLNKTRNC=NO on the OUTCLASS( ) statement
- JES3 - TRUNC=NO on SYSOUT or BUFFER statement

Figure 5-25 JES FSS/FSA definitions for IP PrintWay

### JES definitions for IP PrintWay

To support the FSS/FSA environment for IP PrintWay basic mode, you need to do the following:

- ▶ Define one or more FSS address space definitions.
- ▶ Define devices in the JES initialization statements.

The JES initialization streams must include statements for printer devices that will be needed to start an FSA in a specified IP PrintWay FSS address space. In the example in Figure 5-11 on page 254 there are five JES printer devices started in the JES address space (not shown), which then starts the three FSAs in one FSS address space and two in the other.

- ▶ Define work selection criteria for the FSA to select data sets.
- ▶ Define one or more IP PrintWay procedures.

### Defining the IP PrintWay functional subsystem in JES2 (basic mode)

To define a functional subsystem for IP PrintWay in a JES2 environment, code these JES2 initialization statements:

- ▶ One FSS(fss\_name) statement to define each IP PrintWay functional subsystem (FSS)
- ▶ One PRTnnnnn statement to define each functional subsystem application (FSA) under control of the FSS

Also, the following JES2 initialization statements have considerations for IP PrintWay:

- SPOOLDEF** Specify TRKCELL=5 on the JES2 SPOOLDEF statement for improved performance. Also, specify TRKCELL=YES on the PRTnnnnn statement.
- OUTCLASS** Specify BLNKTRNC=NO on the JES2 OUTCLASS statements for all output classes that IP PrintWay basic mode processes. This prevents JES from removing EBCDIC blanks at the end of each record on the JES spool, which JES does by default to save spool space. Some data streams (for example, PCL, PostScript, and PDF data streams) might not print properly if JES removes blanks. The JES default is BLNKTRNC=YES.
- If you specify BLNKTRNC=YES or take the default, before IP PrintWay basic mode prints a data set, it restores the blanks that JES removed unless the data set contains line data or text data. For the best performance, however, you should always request that JES2 perform no blank truncation for all IP PrintWay output classes.
- PRINTDEF** The JES2 PRINTDEF statement lets you specify a default FCB and page definition in the NIFCB parameter and a default font in the NIUCS parameter.

### Defining the IP PrintWay functional subsystem in JES3 (basic mode)

To define a functional subsystem for IP PrintWay in a JES3 environment, code these JES3 initialization statements:

- ▶ One FSSDEF statement to define each IP PrintWay functional subsystem (FSS)
- ▶ One DEVICE statement for each functional subsystem application (FSA) under control of the FSS

Also, specify TRUNC=NO on the JES3 SYSOUT statements for all output classes that IP PrintWay basic mode processes. This prevents JES from removing EBCDIC blanks at the end of each record on the JES spool, which JES does by default to save spool space. ASCII data streams (for example, PCL, PostScript, and PDF data streams) might not print properly if JES removes EBCDIC blanks. The JES default is the value set in the TRUNC parameter of the JES3 BUFFER statement or, if none is specified, TRUNC=YES.

If you specify TRUNC=YES or take the default, before IP PrintWay basic mode prints a data set, it restores the blanks that JES removed unless the data set contains line data or text data. To obtain this support, you must run the z/OS V1R5 (or higher) level of JES3. For the best performance, however, you should always request that JES3 perform no blank truncation for all IP PrintWay basic mode output classes.

### IP PrintWay extended mode

These JES definitions are not used for IP PrintWay extended mode.

## 5.24 Infoprint Server extended mode versus basic mode

- ❑ Better performance
- ❑ Increased reliability
- ❑ Improved usability
- ❑ Enhanced printer management functions
- ❑ Enhanced print job management functions
- ❑ All messages written to common message log
- ❑ Automatically detect data format and verify job will print on selected printer

Figure 5-26 IP PrintWay extended mode versus basic mode

### Extended mode advantages

IP PrintWay extended mode offers the following advantages over the basic mode using FSSes and FSAs:

- ▶ Better performance in that print jobs start printing sooner.
- ▶ A printer problem or large job on one printer no longer delays printing to another printer because IP PrintWay calls data stream transforms directly without resubmitting them to Print Interface for filtering. IP PrintWay extended mode can restart printing after an error from the last page that the printer reported printed successfully.
- ▶ Increased reliability because IP PrintWay can retain an unlimited number of print jobs on the JES spool without running out of address space and abending with F02 (available storage exhausted) abends.
- ▶ Improved usability for operators to manage all IP PrintWay printers and print jobs from the web using Infoprint Central. Administrators can define job selection rules in the Printer Inventory to control which output data sets IP PrintWay extended mode selects to print.
- ▶ New printer management and print job management functions for operators using Infoprint Central.
- ▶ IP PrintWay extended mode writes all messages to the new common message log instead of to the IP PrintWay message data set.
- ▶ IP PrintWay extended mode uses automatic data format detection to verify that the selected printer has the capability to print the job prior to sending it to the printer.

## 5.25 IP PrintWay extended mode enhancements

- ❑ Using IP PrintWay extended mode, you can manage all IP PrintWay printers and print jobs from the Web using Infoprint Central
  - Stop and restart printers
  - Redirect print jobs from one printer to another
  - Delete and hold jobs
  - Delete and hold jobs that IP PrintWay extended mode is currently processing
- ❑ Call data stream transforms and other filters directly without resubmit for filtering
- ❑ Process print jobs larger than 2 gigabytes

Figure 5-27 IP PrintWay extended mode enhancements

### IP PrintWay extended mode enhancements

Operators can use Infoprint Central, instead of Infoprint Server ISPF panels, to work with IP PrintWay extended mode printers and print jobs, as follows:

- ▶ Stop printing to printers. Restart printing to stopped printers.
- ▶ Redirect all print jobs on a printer's queue (except for the print job that is currently processing), as well as all future print jobs, to an alternate printer. The alternate printer must also be an IP PrintWay printer. Redirections remain in effect until the operator restores the original printers or until IP PrintWay is restarted.
- ▶ Delete print jobs that IP PrintWay extended mode is currently processing. Hold print jobs that IP PrintWay extended mode is currently processing.
- ▶ Infoprint Central operators can use JES commands to work with print jobs that IP PrintWay extended mode has selected to process, is waiting to retry, or has retained on the JES spool.

IP PrintWay extended mode calls data stream transforms and other filters directly without resubmitting them to Print Interface for filtering. Calling transforms directly is more efficient because data is not written to the JES spool a second time.

IP PrintWay extended mode can print data sets larger than 2 gigabytes if space is available in the /var/Printsrv/ file system.

## 5.26 Enhanced functions with extended mode

- ❑ Process and retain more data sets on the JES spool without running out of address space
  - This reduces the possibility F02 abends
- ❑ Prints output data sets in priority order
- ❑ Print to printers that have IPv6 addresses
  - You must use the host name (instead of the colon-hexadecimal address) in the DEST=IP: JCL parameter, in Infoprint Server job attributes, and in printer definitions
- ❑ Writes the printer address for all protocol types in the SMF type 6 record
- ❑ Limitation:
  - Sysplex: IP PrintWay extended mode cannot share printing information across multiple systems in a sysplex

Figure 5-28 IP PrintWay extended mode enhanced functions and limitation

### IP PrintWay extended mode enhanced functions

IP PrintWay extended mode can process and retain more data sets on the JES spool without running out of address space. This reduces the possibility of ending abnormally with an F02 abend code.

IP PrintWay extended mode prints output data sets in priority order.

IP PrintWay extended mode can print to printers that have IPv6 addresses. However, you must use the host name (instead of the colon-hexadecimal address) in the DEST=IP: JCL parameter, in Infoprint Server job attributes, and in printer definitions.

IP PrintWay extended mode writes the printer address for all protocol types in the SMF type 6 record.

IP PrintWay extended mode has enhanced how it selects print jobs from the JES spool:

- ▶ It can select all print jobs that specify the DEST=IP: parameter on the OUTPUT JCL statement. You specify this job selection criterion in a job selection rule in the Printer Inventory.
- ▶ If print jobs request a printer that is not defined in the Printer Inventory, IP PrintWay extended mode releases the print job back to JES so that another program can select the print job for processing.

## Extended mode and JES definitions

IP PrintWay extended mode does not require that you customize JES to request no blank truncation as IP PrintWay basic mode does. This is because JES always restores any blanks that it removed before giving data to IP PrintWay extended mode. Therefore, you can request either that JES truncate blanks to save spool space, or that JES not truncate blanks to improve performance. For the unusual print jobs that do not print properly, you can specify the the AOP\_BLANK\_TRUNCATION\_CLASSES environment variable to request that IP PrintWay extended mode truncate blanks.

## Sysplex limitation

IP PrintWay extended mode cannot share printing information across multiple systems in a sysplex as IP PrintWay basic mode can:

- ▶ IP PrintWay basic mode keeps queue information in its transmission-queue data set and keeps messages in its message-log data set. Both data sets can be shared across multiple systems.
- ▶ IP PrintWay extended mode keeps queue information in the Printer Inventory and keeps messages in the common message log. The Printer Inventory and common message log cannot be shared across multiple systems.

## 5.27 IP PrintWay extended mode customization

- ❑ Edit the aopd.conf configuration file
  - **resolve-printway-printers = yes | no**
  - **smf-recording = yes | no**
  - **start-daemons = { outd }**
- ❑ Increase space in /var/Printsrv directory (>750 MB)
- ❑ Update a user profile of the daemon user ID
  - **PROCUSERMAX(70) and MAXPROCSYS**
- ❑ Create IP PrintWay job selection rules
- ❑ Authorize IP PrintWay to select JES spool data
- ❑ Set environment variables
- ❑ Customize z/OS Communications Server
  - **Resolver info LOOKUP = DNS ,TCPIP profile, sendmail**
- ❑ Customize PSF for sharing network printers
  - **JES spool-driven**
  - **VTAM RELREQ**
  - **Port switching**

Figure 5-29 IP PrintWay extended mode customization

### IP PrintWay extended mode customization

You must add or edit the Infoprint Server configuration file, aopd.conf, to customize IP PrintWay extended mode with the following parameters that are discussed in “Full format of the aopd.conf file” on page 77:

- ▶ **resolve-printway-printers = yes | no** - Indicates whether IP PrintWay extended mode looks up the full host names of printers in the Domain Name Server (DNS).

**yes** - IP PrintWay looks up the full host names of printers in the DNS. This lets administrators and job submitters specify printer addresses using all of these methods: dotted-decimal IP address, short host name, and full host name.

For example, assume that the administrator creates three printer definitions for the same printer. In each printer definition, the administrator specifies the printer's address in a different way: dotted-decimal IP address, short host name, and full host name. When IP PrintWay starts, it looks up the full host names of all printers in the DNS. It recognizes that these three printer definitions all refer to the same printer.

Therefore, IP PrintWay sets up only one internal print queue for the printer. This ensures that IP PrintWay sends print jobs to the printer in the order they were submitted and reduces contention within IP PrintWay for the printer. In addition, operators who use Infoprint Central see only one IP PrintWay printer.

**no** - IP PrintWay does not look up the full host names of printers in the DNS. This means that administrators and job submitters must use a consistent method to specify printer addresses. For example, in all printer definitions for the same printer the administrator

must specify the dotted-decimal IP address, the short host name, or the full host name of the printer so that IP PrintWay recognizes that multiple printer definitions refer to the same printer.

If you change this attribute while Infoprint Server is running, stop and restart the IP PrintWay extended mode daemons.

Default: `resolve-printway-printers = yes`

- **smf-recording = yes | no** - Indicates whether IP PrintWay extended mode is to write SMF type 6 records for data sets that it sends to a printer or e-mail destination. SMF type 6 records contain accounting information. If you change this attribute while Infoprint Server is running, stop and restart the IP PrintWay extended mode daemons.

If you want IP PrintWay extended mode to write System Management Facilities (SMF) type 6 records, you must authorize the user ID that starts Infoprint Server daemons to write SMF records. To do this, give the user ID that starts Infoprint Server daemons READ access to the BPX.SMF profile in the FACILITY class.

- **start-daemons = { outd }** - Specifies the daemons that start when you run the **aopstart** command or AOPSTART JCL procedure. To run IP PrintWay extended mode, add the outd value to any existing values in this attribute to start the aopoutd and aopwsmd daemons. Enclose all values in braces. If you add a value while Infoprint Server is running, restart Infoprint Server to start the new daemons. You do not need to first stop Infoprint Server daemons that are running. Also, restart the z/OS HTTP Server if you run Infoprint Central to pick up the change.

## File system space

IP PrintWay extended mode writes data to the **/var/Printsrv** directory before sending the data to the printer. Therefore, you might need to increase the amount of space available to the filesystem mounted at the **/var** or **/var/Printsrv** mount point.

## Daemon user ID OMVS parameters

When you use IP PrintWay extended mode, you should increase the number of active z/OS UNIX processes that the z/OS system allows. You can specify the number of allowed processes in these parameters in the OMVS segment of the user profile:

- **PROCUSERMAX** - The maximum number of processes that a single z/OS UNIX user ID can have concurrently active. Recommendation: Specify at least 70.
- 1. **PROCSYSMAX** - The maximum number of processes that the z/OS system allows. Recommendation: Add an additional 50 to the current MAXPROCSYS value. If you also run Print Interface or NetSpool, add an additional 200 for processes used by both NetSpool and Print Interface.
- **STEPLIBLIST** - The pathname of a hierarchical file system (HFS) file that contains a list of MVS data sets sanctioned by your installation for use as step libraries. Follow these guidelines:
  - In the HFS file, specify the name of each MVS data set on a separate line without quotation marks.
  - Give the HFS file execute permission for other. IBM recommends that UID 0 own the file and that the file's permissions be 755 or 555.

You can also use the BPXPRMxx parmlib member to specify the respective values: the maximum number of processes for a single user ID (MAXPROCUSER), the maximum number of processes that the z/OS system allows (MAXPROCSYS), and the pathname of the file containing a list of MVS data sets for use as step libraries (STEPLIBLIST).



## IP PrintWay job selection rules

IP PrintWay job selection rules determine which print jobs IP PrintWay extended mode selects from the JES spool. You define job selection rules in the Printer Inventory.

## JES spool data

If you activate the JESSPOOL class in your installation, you must give IP PrintWay extended mode UPDATE access to profiles in this class so that IP PrintWay extended mode can select output data sets from the JES spool.

## Environment variables

You must specify environment variables for IP PrintWay extended mode if any of these conditions apply:

- ▶ You have written IP PrintWay exits and the library that contains the exits is not in the system LNKST
- ▶ You use FCBs in your installation and the SYS1.IMAGELIB library is not in the system LNKST
- ▶ You send output to e-mail destinations and the z/OS UNIX sendmail command was not installed in the default directory
- ▶ You have more than one TCP/IP stack in your installation

To print on TCP/IP-attached printers or to send output to e-mail destinations, you must customize the TCP/IP component of z/OS Communications Server. The tasks related to TCP/IP customization include:

- ▶ Customizing the TCPIP.DATA data set
- ▶ Customizing the PROFILE.TCPIP data set
- ▶ Customizing z/OS UNIX sendmail

If the administrator selects one of the Printer Job Control (PJM) options in the printer definition (Restart printing after last successful page or Record pages printed successfully), that printer definition cannot be used to print documents that already contain PJM JOB commands. This is because a PJM JOB command in the document would conflict with the PJM JOB command that IP PrintWay adds to the document.

If you use Infoprint transforms, you might need to customize the AFP to PCL and AFP to PostScript transforms not to add a PJM JOB command to the transformed data stream. Set `environment = { AOP_PJM->no }` in the `aopxdf.conf` file for these transforms.

## 5.28 Job selection rules - IP PrintWay extended mode

- ❑ IP PrintWay job selection rules determine which print jobs are selected from the JES spool
  - **Job selection criteria in a job selection rule**
    - DEST - output destination of the print data sets
    - CLASS - SYSOUT class of the print data sets
    - FORMS - forms associated with print data sets
    - Creator - z/OS user ID associated with print data sets
    - Writer - external writer name
    - DEST IP address - indicates whether print data sets can specify the DEST=IP: parameter on the OUTPUT JCL statement
- ❑ Resource profile in the PRINTSRV class
  - **Controls Infoprint Central access to job selection rules**

Figure 5-30 Using job selection rules

### Job selection rules (extended mode)

Job selection rules for IP PrintWay extended mode are comparable to the JES work selection criteria you specify for IP PrintWay basic mode functional subsystem applications (FSAs). IP PrintWay extended mode uses the job selection rules defined in the Printer Inventory to determine which print jobs to select from the JES spool for printing. You must create at least one job selection rule for IP PrintWay to select print jobs. The attributes of the print job must match all of the values in a rule to be selected.

### Job selection rule parameters

You can use the Infoprint Server ISPF panels to create and manage job selection rules. You can also use the Printer Inventory Definition Utility (pidu) to create job selection rules. In each job selection rule in the Printer Inventory, you can specify the job selection criteria:

<b>DEST</b>	The destination name of the print jobs. This name must match the DEST parameter of the OUTPUT JCL statement and the DEST field of the printer definition. Wildcards are allowed.
<b>CLASS</b>	The JES output class of the print jobs. This class must match the CLASS parameter of the OUTPUT JCL statement and the CLASS field of the printer definition. You can specify 1- 36 classes.
<b>FORMS</b>	The forms name of the print jobs. This value must match the FORMS parameter of the OUTPUT JCL statement, the forms job attribute, and the FORMS field of the printer definition. You can specify 1 - 8 forms names. Wildcards are allowed.

<b>Creator</b>	The z/OS user ID associated with the print jobs. For print jobs submitted with JCL, the <b>1p</b> command, or the AOPPRINT JCL procedure, specify the user ID of the user who submits the print requests. For other print jobs that Print Interface or NetSpool create, specify the user's ID who started the Infoprint Server daemons. Wildcards are allowed.
<b>Writer</b>	The writer name for the print jobs. This value must match the WRITER parameter of the OUTPUT JCL statement and the Writer field of the printer definition. Wildcards are allowed.
<b>DEST IP address</b>	Indicates whether print jobs can specify the DEST=IP: parameter on the OUTPUT JCL statement: <ul style="list-style-type: none"> <li>► <b>Include:</b> Select only print jobs that specify the DEST=IP: parameter.</li> <li>► <b>Exclude:</b> Select only print jobs that do not specify the DEST=IP: parameter.</li> <li>► <b>Ignore:</b> Select print jobs that specify the DEST=IP: parameter and jobs that do not specify the DEST=IP: parameter.</li> </ul>

AOPIPWSL

Job Selection Rule

Command ==>

Rule name . . SAPI

Description . Job Selection rule for SAPI interface (extend)

Operator security profile

. . . POK.JOBSEL

DEST. . . \*

CLASS . . Y

FORMS . . STD

Creator .

WRITER. .

DEST IP address . 3 1. Include 2. Exclude 3. Ignore

Figure 5-31 Infoprint Server ISPF Job Selection Rule display panel

### Using job selection rules

If you specify multiple job selection criteria in a job selection rule, IP PrintWay selects print jobs that meet all the criteria specified in the rule. For example, if you (1) select Include in the DEST=IP field and (2) specify a class in the CLASS field, IP PrintWay selects only print jobs that specify the DEST=IP: JCL parameter and are in the specified output class. If you specify no job selection criteria in a job selection rule, IP PrintWay selects all print jobs from the JES spool. If IP PrintWay extended mode selects a print job for processing, but no printer definition exists for the printer in the Printer Inventory and the DEST=IP: parameter was not specified on the OUTPUT JCL statement, IP PrintWay releases the print job to JES and does not select it again unless the operator releases the print job. Another program, however, can select the print job from the JES spool. The operator can release print jobs using Infoprint Central or JES commands. For example, the operator might want to release the print job after the administrator creates a missing printer definition.

**Important:** If you want to run IP PrintWay extended mode and IP PrintWay basic mode at the same time, make sure that IP PrintWay extended mode does not select the same print jobs that IP PrintWay basic mode does.

## 5.29 Create IP PrintWay extended mode job selection rules

- ❑ Rules are defined in the Printer Inventory
- ❑ In Infoprint Server: Printer Inventory Manager ISPF dialog
  - Select Option 4 then 10 or 11 to define or view job selection rules
  - Job selection rules apply only to IP PrintWay extended mode
- ❑ JES3 complex considerations
  - IP PrintWay on each system must select different print jobs
  - Print Interface and NetSpool must allocate print jobs with job-selection criteria that match IP PrintWay running on the same system

```
AOPIPP1          Infoprint Server: Printer Inventory Manager
Option ==> 4_____

Printer Definitions
  1 Add          Add a printer definition
  2 List         List printer definitions
  3 Select       Select printer definitions to list

Other Functions
  4 Other Definitions  Manage FSS, FSA, pool, and job selection definitions
  5 PrintWay Queue    View IP PrintWay basic mode transmission queue
  6 PrintWay Message  View IP PrintWay basic mode message log
  7 Configure         Change panel configuration
```

Figure 5-32 Creating a job selection rule

### Job selection rules

IP PrintWay job selection rules determine which print jobs IP PrintWay extended mode selects from the JES spool. You define job selection rules in the Printer Inventory.

**Note:** In JES3 complex you should run Infoprint Server (Print Interface, NetSpool, and IP PrintWay) on only one system in the sysplex. However, if you want to run Infoprint Server on more than one system, consider the following:

- ▶ IP PrintWay on each system must select different print jobs from the JES spool. To accomplish this, you must define different job selection rules for IP PrintWay on each system. For example, IP PrintWay on system A could select print jobs in class P, while IP PrintWay on system B could select print jobs in class Q.
- ▶ Print Interface and NetSpool on each system must allocate print jobs with job selection criteria that match IP PrintWay running on the same system. For example, Print Interface and NetSpool running on system A could allocate print jobs in class P, while Print Interface and NetSpool running on system B could allocate print jobs in class Q.

The panel shown in Figure 5-32 manages the inventory of printer information that is used by NetSpool, the Print Interface, IP PrintWay, and PSF. These applications obtain information from the inventory about handling jobs on the JES spool and starting and stopping their processes. Option 4 on this panel directs you to the add, display, or update job selection definitions dialog.

## 5.30 Add IP PrintWay extended mode job selection rule

☐ Select Option 10 to add a new job selection rule

A0PIPP2

FSA, FSS, Pool, and Job Selection Rule Management

Option ==> 10

FSA

1 Add

Add an FSA

2 List

List FSAs

3 Select

Select FSAs to list

FSS

4 Add

Add an FSS

5 List

List FSSs

6 Select

Select FSSs to list

Pool of Printer Definitions

7 Add

Add a Pool

8 List

List Pools

9 Select

Select Pools to list

Job Selection Rule

10 Add

Add a Job Selection Rule

11 List

List Job Selection Rules

Figure 5-33 FSA, FSS, Pool, and Job Selection Rule Management panel

### FSA, FSS, Pool, and Job Selection Rule Management panel

The FSA, FSS, Pool, and Job Selection Rule Management panel lets you add, list, or select these types of definitions in the Printer Inventory:

- **Functional subsystem application (FSA)** - FSA definitions contain information about IP PrintWay basic mode and PSF FSAs. (The term “PSF” refers to PSF for z/OS and to the AFP Download Plus feature of PSF.)
- **Functional subsystem (FSS)** - FSS definitions contain information about the IP PrintWay basic mode and PSF FSSs in which the FSAs run.
- **Printer definition pool** - Printer definition pool definitions let a user select a group of printers to print to. The job is then sent to all the printers included in the pool.
- **Job selection rule** - IP PrintWay extended mode uses the job selection rules to determine which print jobs to select from the JES spool for printing.

To use this panel, type one of the listed option numbers in the Option field and press Enter. Or, if you have enabled point-and-shoot, move the cursor to the option and select it. The options are:

- Add** Add an FSA definition, an FSS definition, a pool definition, or a job selection rule.
- List** List all the definitions or rules, and then browse, copy, edit, or delete them.
- Select** Enter selection criteria to list definitions that match the criteria.

## 5.31 Job selection rules in the Printer Inventory

### ❑ Display of a job selection rule

- **POK.JOBSEL** is name of the RACF security profile in the PRINTSRV class

```
AOPIPWSL                               Job Selection Rule
Command ==> _____

Rule name . . SAPI_____
Description . Job Selection rule for SAPI interface_____ (extend)
Operator security profile
. . . POK.JOBSEL_____

DEST. . . *_____
CLASS . . Y-----
FORMS . . STD_____
Creator . _____
WRITER. . _____

DEST IP address . 3 1. Include 2. Exclude 3. Ignore
```

Figure 5-34 Job selection rules in the Printer Inventory

### Job selection rules in the Printer Inventory

The ISPF Job Selection Rule panel lets you add, browse, copy, or edit a job selection rule. This panel applies only to IP PrintWay extended mode.

IP PrintWay uses the job selection rules to determine which print jobs to select from the JES spool for printing. You must create at least one job selection rule for IP PrintWay to select print jobs. The attributes of the print job must match all of the values in a rule to be selected.

One job selection rule should be sufficient for many installations. You do not need to specify more than one job selection rule to obtain more efficient work load balancing. However, if you want IP PrintWay to process different print jobs at different times of the day, or if you want different operators to start and stop processing of different print jobs, you should create separate job selection rules. For example, you could create a separate job selection rule to process print jobs that are submitted to an output class reserved for overnight jobs in your installation. Operators could start this job selection rule in the evening and stop it during the day.

### Operator security profile

The Operator Security Profile Field in the job selection rule definition names of the RACF resource profile in the PRINTSRV class that controls who can work with this job selection rule using z/OS Infoprint Central for the Web.

## 5.32 Job selection rule security profiles

### ❑ Operator security profile

#### ➤ Define profiles to protect job selection rules

- Otherwise, any Infoprint Central user who is authorized to read the Printer Inventory can display, start, and stop job selection rules
- If you specify a profile name that is not defined to RACF, or if this field is blank, anyone can work with this printer

RDEFINE PRINTSRV POK.JOBSEL UACC(NONE)

Do not use AOP.JOBSEL

RDEFINE PRINTSRV (AOP.ADMINISTRATOR) UACC(NONE)

Display	PERMIT AOP.ADMINISTRATOR CLASS(PRINTSRV) ACCESS(READ) ID(userid group)
Start	PERMIT POK.JOBSEL CLASS(PRINTSRV) ACC(CONTROL) ID(userid group)
Stop	PERMIT POK.JOBSEL CLASS(PRINTSRV) ACC(CONTROL) ID(userid group)

Figure 5-35 Job selection rule security profiles

### Job selection rule security profiles

Some Infoprint Central actions require that users be authorized to read the Printer Inventory. These actions include viewing PSF printers, printer definitions, printer pool definitions, and IP PrintWay job selection rules.

To authorize users to read the Printer Inventory, using either Infoprint Central or Infoprint Server ISPF panels, give the AOP.ADMINISTRATOR profile in the PRINTSRV class universal READ access or give all Infoprint Central users READ access.

Profiles in the PRINTSRV class restrict who can work with printers. Profiles in the PRINTSRV class can apply to both IP PrintWay and PSF printers. The RACF **PERMIT** command you use determines what the administrator can do, display, start, or stop the rule.

You can define a separate profile to protect each printer, or you can define one profile to protect a group of printers or all printers. For example, if you want to authorize different users to work with printers in different locations, define separate profiles for printers in each location. If you want to authorize the same group of users to work with all printers, you need to define only one profile.

You specify the name of the profile that applies to each printer in the Printer Inventory:

- ▶ IP PrintWay printers: Specify the profile name in the printer's printer definition.
- ▶ PSF printers: Specify the profile name in the printer's FSA definition.

## 5.33 Authorizing JES spool and SMF for Infoprint Server

- ❑ **RACF JESSPOOL class active**
  - **IP PrintWay extended mode requires UPDATE access**
    - **PERMIT N1.\*.\*.D\*.\* CLASS(JESSPOOL) ID(AOPOPER) ACCESS(UPDATE)**
    - **SETROPTS RACLIST(JESSPOOL) REFRESH**
- ❑ **smf-recording=yes set in aopd.conf**
  - **Infoprint Server user ID must be authorized to write SMF records**
  - **FACILITY class BPX.SMF profile**
    - **PERMIT BPX.SMF CLASS(FACILITY) ID(AOPOPER) ACCESS(READ)**
    - **SETROPTS RACLIST(FACILITY) REFRESH**

Figure 5-36 Authorizing JES spool and SMF for Infoprint Server

### Authorizing JES spool for Infoprint Server

If you activate the JESSPOOL class in your installation, you must give IP PrintWay extended mode UPDATE access to profiles in this class so that IP PrintWay extended mode can select output data sets from the JES spool.

IBM recommends that you give the AOPOPER group access to profiles in the JESSPOOL class because users who start Infoprint Server must be members of the AOPOPER group or have a UID of 0. (AOPOPER is the default group name for Infoprint Server operators. However, your installation can assign a different name to this group.) If someone with a user ID of 0 who is not a member of the AOPOPER group can start Infoprint Server (for example, using the aopstart command), you must also give this user ID access to the profiles in the JESSPOOL class. If you start Infoprint Server in the /etc/rc file, give user ID ROOT access to the profiles, for example:

To define a broad profile to protect all print jobs on node N1, enter this RACF command:

```
RDEFINE JESSPOOL (N1.*.*.D*.*) UACC(NONE)
```

To give the AOPOPER group UPDATE access to the broad profile, enter these RACF commands:

```
PERMIT N1.*.*.D*.* CLASS(JESSPOOL) ID(AOPOPER) ACCESS(UPDATE) SETROPTS  
RACLIST(JESSPOOL) REFRESH
```



## JESSPOOL considerations

If you also define more specific profiles in the JESSPOOL class, give the AOOPER group access to those profiles as well.

If you permit the AOOPER group to profiles while IP PrintWay extended mode daemons (aopoutd and aopwsmd) are started, stop and restart them to pick up the changes. For information about how to stop and start Infoprint Server daemons, see *z/OS Infoprint Server Operation and Administration*.

Do not activate the JESSPOOL class while IP PrintWay extended mode is running.

## Authorizing SMF recording for Infoprint Server

If you want IP PrintWay extended mode to write SMF type 6 records, you must authorize the user ID that starts Infoprint Server to write SMF records. To do this, give the user ID that starts Infoprint Server daemons READ access to the BPX.SMF profile in the FACILITY class.

Users who start Infoprint Server must be members of the AOOPER RACF group, or have a UID of 0. (AOOPER is the default group name for Infoprint Server operators. However, your installation can assign a different name to this group.) Therefore, IBM recommends that you give the AOOPER group access to the BPX.SMF profile as shown in this example. If someone with a user ID of 0 who is not a member of the AOOPER group can start Infoprint Server (for example, using the aopstart command), you must also give this user ID access to the BPX.SMF profile, for example:

To define the BPX.SMF profile, enter this RACF command:

```
RDEFINE FACILITY (BPX.SMF) UACC(NONE)
```

To give READ access to the AOOPER group, enter these RACF commands:

```
PERMIT BPX.SMF CLASS(FACILITY) ID(AOOPER) ACCESS(READ) SETROPTS  
RACLIST(FACILITY) REFRESH
```

## SMF recording considerations

If you do not want IP PrintWay extended mode to write SMF type 6 records, specify the smf-recording=no attribute in the Infoprint Server configuration file.

If you permit the AOOPER group to the BPX.SMF profile while IP PrintWay extended mode daemons (aopoutd and aopwsmd) are started, stop and restart them to pick up the changes. For information about how to stop and start Infoprint Server daemons, see *z/OS Infoprint Server Operation and Administration*, S544-5745.

## 5.34 Customizing z/OS UNIX sendmail

- ❑ **sendmail** setup using default directories and files as defined in `/usr/lpp/tcpip/samples/sendmail/cf/sample.cf`
  - **Copy** `/usr/lpp/tcpip/samples/sendmail/cf/sample.cf` to `/etc/mail/sendmail.cf`
  - **Create** queue directory `/usr/spool/mqueue`
  - **Create and edit** aliases file `/etc/mail/aliases` (Perms 644)
    - Following is an example of an aliases file
  - **Run** `sendmail -bi` or `newaliases` command
  - **Create** localhost file `/etc/mail/local-host-names`
    - Contents for example: `local.host.name`
- ❑ **Example of an alias file:**

```
# Alias for mailer daemon
MAILER-DAEMON:IBMUSER
# Following alias is required by the new mail
protocol, RFC 822
postmaster:IBMUSER
# Alias to handle mail to msgs and news
nobody: /dev/null
```

Figure 5-37 Customizing z/OS UNIX sendmail

### Customizing z/OS UNIX sendmail

The simple mail architecture in which sendmail and popper fit includes a mail user agent (MUA), a mail transfer agent (MTA), and a mail delivery agent (MDA). An MUA is client software that a user invokes directly to send and receive e-mail. Examples of MUAs include Eudora, Netscape Navigator, pine and elm. An MTA is software that actually routes messages from a sender's system to the receiver's system. sendmail is an MTA. It is worth noting, however, that sendmail relies on other programs to implement non-SMTP based transport (for example, UUCP-based transport as well as local delivery to a user's mail spool file). An MDA is server software that delivers received mail to a user's MUA.

When the e-mail protocol is selected in a printer definition, IP PrintWay uses z/OS UNIX sendmail to prepare and send e-mails to the recipients listed in the printer definition. Sendmail is a mail transfer agent provided with z/OS Communications Server that provides enhanced SMTP support. Sendmail version 8.12.1 runs on z/OS V1R5.

### Guidelines to customize sendmail

Sendmail highly depends on the Domain Name Server (DNS). Therefore, be sure to set up this component correctly.

Customize the Timeout option in the sendmail configuration file to suit your installation. The Timeout option specifies how long sendmail waits before sending a warning that e-mail could not be delivered to a remote system and how long sendmail waits before bouncing e-mail. Bounced e-mail is undeliverable e-mail.

Start sendmail as a daemon so that sendmail can receive messages. The sendmail **-bd** option starts sendmail as a daemon, and the **-q** option specifies how often sendmail is to look in its queue to process pending mail, for example:

These commands switch to an effective UID of 0 and start sendmail as a daemon that checks its mail queue every minute:

```
su
/usr/sbin/sendmail -bd -q1m
```

To use the **su** command, you must be permitted to the BPX.SUPERUSER profile in the FACILITY class in RACF.

## Starting sendmail

If you want sendmail to start automatically when the z/OS system is initialized, specify the sendmail command in the `/etc/rc` file.

In the sendmail aliases file, you can create alias names to represent a list of real e-mail addresses. When you create an alias, you should specify the user ID that owns the list in the owner-alias statement. The user ID that owns the list receives notification about bounced e-mails. If you do not specify the user ID that owns the alias name, sendmail sends notification of bounced e-mails to the user ID assigned to the IP PrintWay startup procedure (basic mode) or to the user ID who started the Infoprint Server daemons (extended mode).

Configure the z/OS system so that a firewall does not prevent users from sending e-mails. Also if you want to receive replies from e-mails or sendmail notifications about undeliverable e-mails, configure your z/OS system so that it can receive e-mails from outside the firewall.

If you do not install sendmail in the default directory, `/usr/sbin`, specify the full path name of the sendmail command in the AOPMAILER environment variable in the IP PrintWay startup procedure (basic mode) or in the aopstart EXEC (extended mode).

Much of the sendmail samples directory is dedicated to the automated creation of the configuration file. The `/usr/lpp/tcpip/samples/sendmail/cf` directory contains a sample.mc file and the subsequent sample.cf configuration file that was created by running the m4 macro preprocessor on the sample.mc file.

Archived

# NetSpool

NetSpool intercepts print data from VTAM applications, such as CICS and IMS; transforms the data streams to EBCDIC line data, PCL, PDF, or other formats that the printer accepts; and allocates output data sets on the JES spool. JES or PSF can print the output data sets or JES can transmit them to other locations for printing. IP PrintWay can transmit the output data sets to remote printers in your TCP/IP network.

VTAM data streams supported by NetSpool:

- ▶ SNA character string (SCS) data over a logical unit (LU) type 1 session
- ▶ 3270 data over an LU type 0 or type 3 session
- ▶ Binary data over an LU type 0, type 1, or type 3 session

You can configure NetSpool so that you do not need to change existing VTAM applications. That is, existing VTAM applications can send print requests to NetSpool in the same manner as they currently send print requests to SNA network printers.

Some of the benefits of NetSpool are:

- ▶ Data integrity  
By placing VTAM application output on the JES spool, NetSpool lets you take advantage of the security, checkpoint and restart, and reprint capabilities that JES provides.
- ▶ Printer sharing  
NetSpool lets multiple VTAM applications simultaneously direct output to the JES spool for printing on a single shared printer.
- ▶ Print broadcasting  
NetSpool lets a single VTAM application direct output to the JES spool for printing on multiple printers.
- ▶ AFP formatting  
Users can specify AFP parameters, such as page definition, form definition, and character sets, to take advantage of AFP formatting when printing on AFP printers.
- ▶ DBCS support

NetSpool supports Double Byte Character Set (DBCS) data in both SCS and 3270 data streams.

► Transparent-data support

NetSpool supports transparent data in SCS data streams and provides exits for adding and changing transparent data.

► Owner and job attribute support

VTAM application programmers can embed the *owner* job attribute and other Infoprint Server job attributes in the application's print data that is sent to NetSpool. Also, the administrator can specify a NetSpool default owner in the printer definition.

► Transforms

NetSpool can transform data streams from one format to another, such as from SCS and 3270 data streams to:

- Line data streams for printing on IBM AFP printers (does not require Infoprint Server Transforms)
- Line data streams and then to PDF data for viewing in an e-mail attachment (requires the AFP to PDF transform)
- PCL data streams for printing on network printers (does not require Infoprint Server Transforms)

When converting to PCL, NetSpool can preserve additional print-formatting controls in SCS data streams by generating corresponding PCL commands. This results in greater print fidelity. For example, when converting SCS data, Infoprint Server can:

- Preserve line-density and print-density controls
- Preserve paper-source, output-bin, job-separation, and duplex controls
- Preserve portrait and landscape page-orientation controls
- Automatically change the page orientation and reduce the font size, if necessary, page by page

## 6.1 NetSpool overview

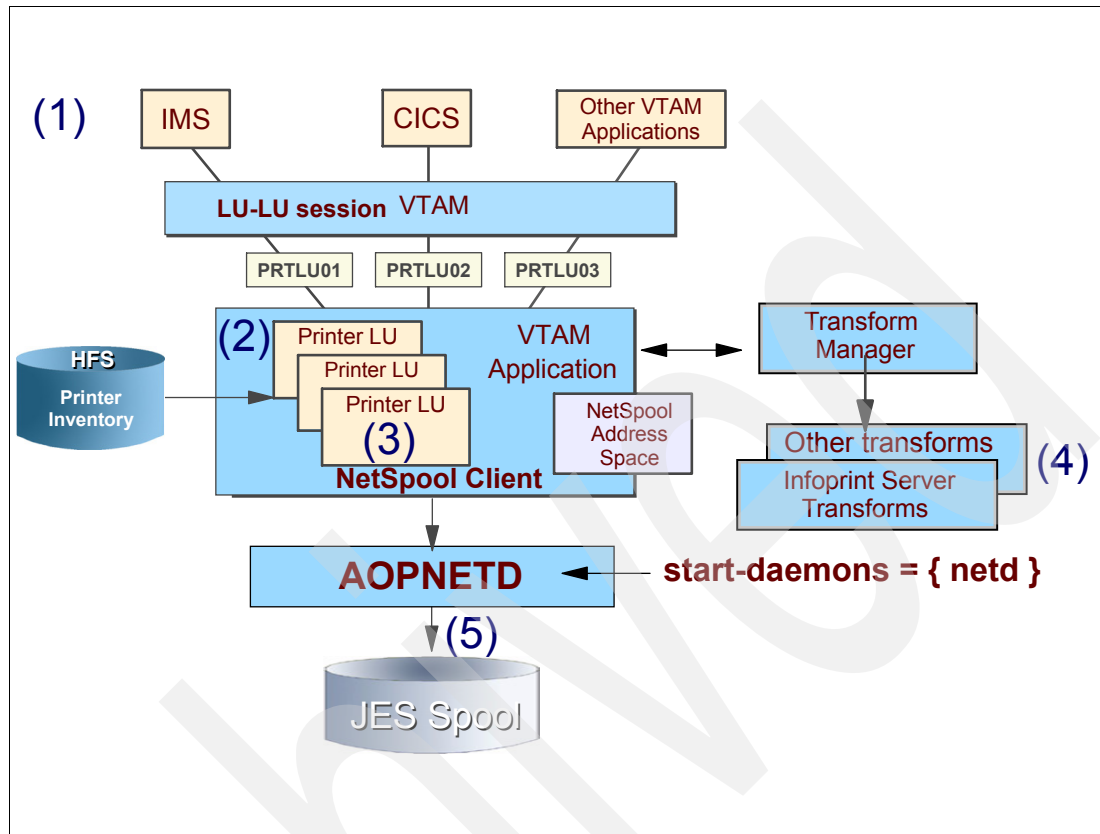


Figure 6-1 NetSpool system diagram

### NetSpool overview

The NetSpool component of Infoprint Server intercepts print data from VTAM applications, such as CICS and IMS; transforms the data streams to EBCDIC line data, PCL, PDF, or other formats that the printer accepts; and creates output data sets on the JES2 or JES3 spool. You can configure NetSpool so that you do not need to change existing VTAM applications. That is, existing VTAM applications can send print requests to NetSpool in the same manner as they currently send print requests to SNA network printers.

### NetSpool as a VTAM application

Figure 6-1 shows the steps that occur from the time VTAM applications send print requests to NetSpool printer logical units (LUs) until NetSpool allocates output data sets on the JES spool. An explanation of each step follows:

1. VTAM applications, such as CICS or IMS, establish communication sessions with NetSpool printer logical units (LUs) instead of with SNA-network printers. Each NetSpool printer LU must be defined to VTAM as an application logical-unit (LU).

NetSpool can process these types of VTAM data streams:

- SNA character string (SCS) data over an LU type 1 session. An SCS is composed of EBCDIC controls, optionally intermixed with end-user data, that is carried within a request/response unit.

- 3270 data over an LU type 3 or LU type 0 session. The 3270 inbound EBCDIC data stream can consist of attention identification (AID) bytes, orders, application data, sense information, and control information.
  - A binary data stream over an LU type 0, type 1, or type 3 session. NetSpool treats input data as binary data. NetSpool writes binary data to the output data set, in variable length records, without examining or converting the data stream.
2. NetSpool runs as a VTAM application on the same or different z/OS systems. NetSpool can process VTAM print requests sent to different NetSpool printer LUs.
  3. The administrator defines each NetSpool printer LU in a printer definition or printer pool definition in the Printer Inventory. NetSpool uses information in the printer definition to format data into lines and pages, transform data to a format that the printer accepts, and group the data into output data sets.
  4. NetSpool can convert EBCDIC data to ASCII. It can transform SCS and 3270 data streams to EBCDIC line data, PCL, PDF, or other formats that the printer accepts. To do some transforms, it uses transforms that Infoprint Server Transforms and other optional transform products provide.
  5. NetSpool dynamically allocates output data sets on the JES spool using JES allocation parameters specified in the printer definition, including:
    - JES work-selection parameters, such as class, forms name, and destination. These parameters cause JES to direct the output data sets to the correct program, such as IP PrintWay or PSF.
    - Advanced Function Presentation (AFP) parameters, such as the name of a form definition and page definition. PSF uses these parameters when printing data on IBM AFP printers.
    - Distribution information, such as name and address, which can be printed on output header pages.

## NetSpool daemon

The NetSpool daemon, **aopnetd**, controls part of the NetSpool processing. You start the daemon with the **aopstart** command. You can only run one NetSpool daemon, but it can control several NetSpool started tasks.

## NetSpool started task address space

You can start more than one NetSpool task at the same time. Each started task runs in a different address space. You might want to start more than one task for these reasons:

- ▶ To spread processing across multiple address spaces
- ▶ To reduce region-size requirements by spreading the printer LUs across multiple regions
- ▶ To enable different sets of printer LUs to be started and stopped at different times



## 6.2 NetSpool features

- ❑ **NetSpool features**
  - **Transforms - SCS and 3270 data streams**
    - To line data and PCL
    - To line data and then PostScript or PDF
    - Binary data support - No formatting
  - **Validation of print requests**
  - **Operator control**
    - NetSpool LUs from Infoprint Central
    - z/OS consoles
  - **Broadcasting output**
    - Printer pool definition
  - **Exits and filters**
  - **Owner and job attribute support**
    - Print data
    - Printer definition
    - Default - user ID of the NetSpool daemon aopnetd

Figure 6-2 NetSpool features

### NetSpool and transforms

NetSpool can transform data streams from one format to another. For example, NetSpool can do these transforms:

- ▶ Transform SCS and 3270 data streams into line data streams. Line data can be printed on IBM AFP printers. Infoprint Server Transforms is not required.
- ▶ Transform SCS and 3270 data streams into PCL data streams. PCL data can be printed on network printers. Infoprint Server Transforms is not required.
- ▶ Transform SCS and 3270 data streams to line data and then to PostScript data. PostScript data can be printed on network printers or sent to e-mail destinations. The AFP to PostScript transform, provided by Infoprint Server Transforms, is required.
- ▶ Transform SCS and 3270 data streams to line data and then to PDF data. PDF data can be sent to e-mail destinations. The AFP to PDF transform, provided by Infoprint Server Transforms, is required.

For information about how NetSpool transforms SCS and 3270 data streams, see the appendixes in *z/OS Infoprint Server User's Guide*, S544-5746.

### Validation of print requests

NetSpool can validate, with some exceptions, that the document can print as requested on the selected printer. For example, NetSpool can reject documents that are too large to print on the selected printer.

## **Operator control**

The operator can control NetSpool LUs from Infoprint Central, from the system console, or from MCS consoles. For example, the operator can display the status of NetSpool LUs, stop them, and start them.

The operator can use tools such as SDSF and Infoprint Central to find output data sets that NetSpool writes to the JES spool. Infoprint Central can display additional status such as whether the output data sets (called print jobs) completed successfully, were retained due to failed transmission to LAN printers, or were deleted before printing.

## **Broadcasting output**

You can print output to several printers and send it to e-mail destinations with one print request. To do this, the administrator creates a printer pool definition in the Printer Inventory. When VTAM application data is submitted to a printer pool definition, NetSpool creates multiple output data sets on the JES spool.

## **Exits and filters**

System programmers can write and install exits to customize NetSpool processing. NetSpool exits can add data to the beginning of output data sets, map graphic escape characters to other printable characters, and modify or delete transparent data in SCS data streams. System programmers can also write filter programs to provide their own data stream transforms.

## **Binary data support**

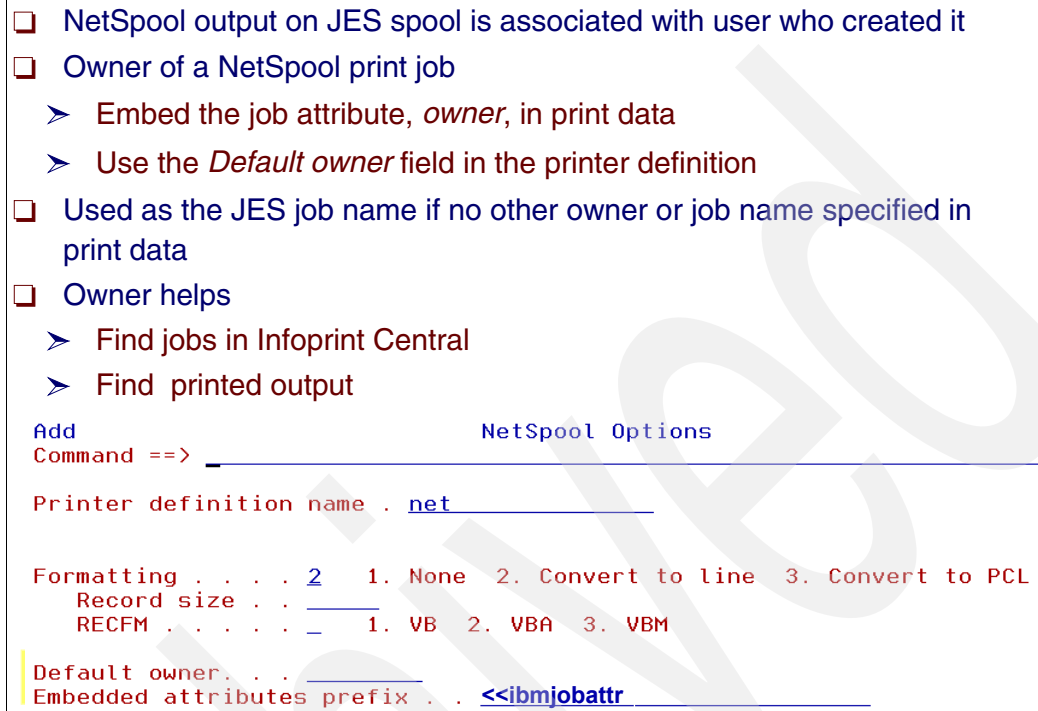
The administrator can request in the printer definition that NetSpool treat the data stream as binary data. NetSpool writes binary data to the spool as variable length records without formatting the data and without rejecting unsupported commands, orders, or data. This function is useful if you want to pass through all data without change and without including transparent (TRN) controls.

## **Owner and job attribute support**

VTAM application programmers can embed the owner job attribute and other Infoprint Server job attributes in the application's print data that is sent to NetSpool. Application programmers might want to embed the owner job attribute to help identify the owner of printed output or the user to charge in an accounting system for printing jobs. Application programmers might want to embed other job attributes, such as forms, page-definition, or form-definition. Also, the administrator can specify a NetSpool default owner in the printer definition.

The default Infoprint Server job owner for the associated logical unit is retrieved from the printer definition if the print data does not specify an owner. The job owner is used for output data sets created for this NetSpool LU if not specified elsewhere. The default job owner is the user ID of the NetSpool daemon AOPNETD.

## 6.3 NetSpool owner definitions



The screenshot shows the 'NetSpool Options' dialog box. It has a title bar 'NetSpool Options' and a 'Command ==>' field. Below this is a section for 'Printer definition name . net'. There are three tabs: 'Formatting', 'Record size', and 'RECFM'. The 'Formatting' tab is selected, showing options: '1. None', '2. Convert to line', and '3. Convert to PCL'. The 'Record size' tab is selected, showing options: '1. VB', '2. VBA', and '3. VBM'. The 'RECFM' tab is selected, showing options: '1. VB', '2. VBA', and '3. VBM'. Below these are fields for 'Default owner.' and 'Embedded attributes prefix . . <<ibmjobattr'.

- ☐ NetSpool output on JES spool is associated with user who created it
- ☐ Owner of a NetSpool print job
  - Embed the job attribute, *owner*, in print data
  - Use the *Default owner* field in the printer definition
- ☐ Used as the JES job name if no other owner or job name specified in print data
- ☐ Owner helps
  - Find jobs in Infoprint Central
  - Find printed output

Add Command ==> \_\_\_\_\_

Printer definition name . net

Formatting . . . . . 2 1. None 2. Convert to line 3. Convert to PCL

Record size . . . . . \_\_\_\_\_ 1. VB 2. VBA 3. VBM

RECFM . . . . . — 1. VB 2. VBA 3. VBM

Default owner. . . . . \_\_\_\_\_

Embedded attributes prefix . . . . . <<ibmjobattr

Figure 6-3 NetSpool owner definitions

### NetSpool owner definitions

This single-valued attribute specifies the default Infoprint Server job owner for the associated logical unit (LU) if the print data does not specify an owner. The job owner is used for output data sets created for this NetSpool LU. NetSpool now lets you specify the owner of a NetSpool print job. You can either:

- ▶ Embed the job attribute, *owner*, in print data. This attribute applies only to NetSpool. The job owner in this attribute is also used as the JES job name if no other owner or job name is specified in the print data. The job owner specified in this field is not used as the JES job owner. The JES job owner is always the ID of the user who started the Infoprint Server daemons.
- ▶ Use the Default owner field in the printer definition, as shown in Figure 6-3.

Either method can help you identify the owner of printed output or the user to charge in an accounting system for printing jobs, such as SMF accounting records.

**Default value:** NetSpool uses the ID of the user who started the Infoprint Server daemons as the default job owner.

### Using NetSpool owner definitions

The Infoprint Server assigned owner helps you find jobs using Infoprint Central.

## 6.4 Embedding job attributes in the NetSpool print data

- ❑ Job attributes can be imbedded in the print data
  - Job owner
  - Job attributes, such as forms, pagedefs, formdefs
- ❑ Code text strings containing job attributes
  - NetSpool uses attributes and removes them from data
- ❑ Syntax of text string
  - Prefix - <<ibmjobattr
  - Continuation field - 0
  - Length field - 010
  - Example:  
    <<ibmjobattr0010owner=JUHA.....print data.....

Figure 6-4 Specifying job attributes for NetSpool print data

### Specifying job attributes

You can embed the owner job attribute and other job attributes for NetSpool jobs in print data, as follows:

- ▶ The owner job attribute for a print job.
- ▶ Other job attributes to specify formatting attributes for an application. For example, the forms, page-definition, or form-definition job attributes.

To embed the owner job attribute or other job attributes in print data, you code a text string that contains the job attributes. NetSpool uses the embedded attributes, and then removes the text string from the data stream before it converts the data stream and puts it on the JES spool.

As an example of job attribute specification, the following text string embeds the owner job attribute “JUHA” before the print data “....print data.....”:

```
<<ibmjobattr0010owner=JUHA....print data.....
```

The syntax of the text string is:

#### Prefix

The part of the header that indicates that job attributes are embedded in the print data. The prefix must match the embedded attributes prefix field in the printer definition in the Printer Inventory.

Prefix in the example: <<ibmjobattr

**Continuation field** The part of the header that indicates whether another text string follows this text string. You might need to use two or more text strings if the text string would be longer than the line length that is specified in the print data.

Continuation field in the example: 0 - no other text strings follow.

**Length field** The part of the header that indicates the total length of all the job attributes.

Length field in the example: 012

## Job attribute examples

Specifying NetSpool print data set names might produce the following IP PrintWay messages:

- ▶ Target printer's netspool options does not specify default owner and the owner job attribute is not used:  
+AOP130I Data set: AOPSS.NETSP00L.JOB41872.D0000013.SC43RM02 was deleted.
- ▶ Target printer's netspool options specifies **ITSO** as the default owner. The owner job attribute is not used:  
+AOP130I Data set: ITSO.ITSO.JOB41872.D0000017.SC43RM02 was deleted.
- ▶ Target printer's netspool options specified ITSO as the default owner and the owner=JUHA job attribute is used:  
+AOP130I Data set: JUHA.JUHA.JOB41872.D000001E.SC43RM02 was deleted.
- ▶ The data set name syntax:  
user\_ID.jobname.Infoprint\_job\_ID.AOPNETD\_JES\_spool\_ds\_seq#.printing\_VTAM\_LU\_name

Following are examples of specifying one or more job attributes, such as the owner, forms, page-definition, or form-definition. Separate more than one job attribute with a space.

```
owner=murphy
owner=smith copies=3
department-text=Purchasing
```

## Rules for coding text strings for Netspool

Put the text string in the first request unit (RU) in the first chain at the start of a print request. For example, in a CICS application, include the text string in the data that is sent in the first SEND command of a chain. NetSpool determines the start of a print request using the bracket and chaining of the data, and using the end-of-file rules specified for the LU in the printer definition.

In 3270 data streams, put the text string after the Command Code and the Write Control Character, but before any End Message Control Character (X'19').

Consider using a delimiter, such as a carriage return, between the text string and any print data to be sure that the data prints correctly on the page. In 3270 data streams, you can use a Set Buffer Address Control as the delimiter.

In data streams that contain double-byte character set (DBCS) data, code the text string as single-byte data.

IBM recommends that you do not use embedded job attributes if you specify the Timer end-of-file rule. Results can be unpredictable.

## 6.5 Customizing NetSpool

- ❑ Editing the aopd.conf configuration file
  - `start-daemons = {netd}`
- ❑ Customizing the NetSpool message log
  - Informational and error messages in both the common message log and the NetSpool message-log data set
- ❑ Customizing the NetSpool startup procedure
- ❑ Working with JES spool limits
  - In z/OS V1R6 (and later), use:
    - `_BPX_UNLIMITED_OUTPUT` environment variable
- ❑ Changing LE run-time options

Figure 6-5 NetSpool customizing tasks

### Editing the aopd.conf configuration file

NetSpool uses the following attributes in the aopd.conf configuration file. Add or edit these attributes to customize NetSpool for your installation, as follows:

- ▶ **start-daemons = {netd}** - To start the NetSpool daemon with the **aopstart** command, add **netd** to the values in this statement. This statement is required to start the NetSpool daemon. If you change this attribute while Infoprint Server is running, stop and restart all Infoprint Server daemons to pick up the change. Also, restart the z/OS HTTP Server if you run Infoprint Central to pick up the change.
- ▶ **ebcdic-codepage** - Specify the default EBCDIC code page NetSpool uses as the source (document) code page when it converts data from EBCDIC to ASCII. If not specified, the default is code page IBM-1047.
- ▶ **job-prefix = prefix** - A prefix that NetSpool and Print Interface use for all Infoprint Server job IDs. Specify 2 letters, numbers, or national (@ \$ #) characters. Enclose a prefix that contains national characters in single or double quotation marks. The first character cannot be numeric.
- ▶ **log-retention** - Specify how many days worth of messages NetSpool writes to the common message log. If not specified, NetSpool only writes messages to the console log and to the optional NetSpool message-log data set.

## Customizing the NetSpool message log

The NetSpool started task can write informational and error messages in both the common message log and the NetSpool message-log data set:

- ▶ The common message log lets you see messages from most Infoprint Server components in one place. The log contains messages from all components of Infoprint Server except for IP PrintWay basic mode. It does not contain messages from Infoprint Server Transforms or other transform products. By default, no messages are kept in the common message log.
- ▶ The NetSpool message-log data set is an optional data set that is organized as a circular queue. When a message extends to the end of the data set, the next message starts at the beginning. In this way, the size of the message data set remains stable, with old messages automatically overlaid by newer messages. Not all NetSpool print job-related errors are written to the NetSpool message-log, so IBM recommends using the common message log for job-related messages.

To use the NetSpool message log, you must allocate space for the data set and initialize it. You can use the sample JCL in SYS1.SAMPLIB (APIMIJCL) to allocate space and run the ANFMFILE program to initialize the data set.

## JES spool limits

JES allows Infoprint Server daemons to continue processing when they exceed their spool limit. The `_BPX_UNLIMITED_OUTPUT` environment variable lets z/OS UNIX programs that have an effective UID 0 and run in their own address space, such as the Infoprint Server daemons, continue processing when they exceed their spool limit. In this case, the JES spool limit for each program is fixed at slightly less than 1 GB, with large spool limits for lines and AFP pages. Spool limits and JES actions specified in JES initialization statements or on the JOB JCL statement do not apply.

The aopstart EXEC defines `_BPX_UNLIMITED_OUTPUT=YES` in the environment in which it starts Infoprint Server daemons.

The specification of `_BPX_UNLIMITED_OUTPUT=YES` in the aopstart EXEC does not affect the z/OS SMB server, which uses the Infoprint Server Application Programming Interface (API) to write data to the JES spool.

## 6.6 Customizing the NetSpool startup procedure

```
//NETSPOOL PROC LUCLASS=(1),INV='AOP1',DURATION=NOLIMIT,
//
//          SIZE=31M,MSGFILE=USER.MSGFILE
//*****
//*      SAMPLE NAME:          APIJPJCL
//*      DESCRIPTIVE NAME:     Sample NetSpool Startup Procedure
//*      STATUS:               HNET705
//*      FUNCTION:This is a sample NetSpool startup procedure, which you
//*                  can use as a model for your installation.
//*      NOTE:Change the data set names and symbolic parameters for your
//*                  installation.
//*                  See the OS/390 Infoprint Server Customization publication
//*                  (G544-5694) for an explanation of the JCL statements
//*                  shown here.
//*****
//APIPPAAA EXEC PGM=APIPPAAA,REGION=&SIZE,TIME=&DURATION,          X
//          PARM='LUCLASS=&LUCLASS,INV=&INV'
//* Modify the next STEPLIB to replace the transparent data exit or
//* the beginning of file exit. This library must be APF authorized.
//*STEPLIB DD DSN=NETSPOOL.EXIT.SLOADLIB,DISP=SHR
//APIMMSG DD DSN=&MSGFILE,DISP=SHR
//SYSPRINT DD SYSOUT=*
```

Figure 6-6 NetSpool startup procedure

### NetSpool start procedure

You can start more than one NetSpool task at the same time. Each started task runs in a different address space. You might want to start more than one task for these reasons:

- ▶ To spread processing across multiple address spaces
- ▶ To reduce region-size requirements by spreading the printer LUs across multiple regions
- ▶ To enable different sets of printer LUs to be started and stopped at different times

**Restriction:** If you start more than one NetSpool task, you cannot use Infoprint Central to work with NetSpool LUs.

### Multiple NetSpool tasks

The NetSpool daemon `aopnetd` controls part of the NetSpool processing. You start the daemon with the `aopstart` command. You can only run one NetSpool daemon, but it can control several NetSpool started tasks.

If you start more than one NetSpool task, specify unique class values in the `LUCLASS` parameter in the startup procedure for each task. These class values correspond to LU class values specified in the printer definitions in the Printer Inventory. Unique class values cause each NetSpool started task to start different printer LUs.



## Editing the NetSpool startup procedure

NetSpool provides a startup procedure in SYS1.PROCLIB (APIJPJCL). You can copy and modify this procedure for your installation.

Edit the APIJPJCL procedure if any of these conditions apply to your installation:

- ▶ If you did not add the Language Environment run-time library, CEE.SCEERUN, to the system LNKLST, add the SCEERUN data set to the STEPLIB DD statement.
- ▶ If you did not add the C++ run-time library, CBC.SCLBDLL, to the system LNKLST, add the SCLBDLL data set to the STEPLIB DD statement.
- ▶ If you use a NetSpool exit and the library that contains the exit is not in the standard z/OSR search order, add the library to the STEPLIB DD statement.
- ▶ If you want to start NetSpool LU classes other than class 1, specify the classes in the LUCLASS parameter of the PROC statement.
- ▶ If the Printer Inventory name is not the default name, AOP1, specify the inventory name in the INV parameter of the PROC statement.
- ▶ If you do not use the default name for the NetSpool message-log data set, change the data set name in the MSGFILE parameter of the PROC statement.
- ▶ If you want to receive NetSpool messages in Japanese, specify the LANGUAGE parameter in the EXEC statement.
- ▶ When using MCS consoles, messages sent in response to NetSpool operator commands are routed to the console that issued the command. If you want to receive NetSpool unsolicited messages on an alternate console, specify the CONSNAME parameter in the EXEC statement.

## APIJPJCL procedure PARM field parameters

You can specify a maximum of 100 characters for the value of the PARM parameter.

<b>LUCLASS=</b>	The classes of printer LUs that this NetSpool task is to start. The class of each printer LU is specified in the printer definition in the Printer Inventory.  Valid class values are 1 to 64. Enclose the classes in parentheses, even if you specify only one class value. To specify more than one class, separate each class with a comma. NetSpool starts all printer LUs that are assigned to any one of the classes specified. For example, if you specify LUCLASS=(1,2), NetSpool starts all printer LUs assigned to class 1 and all printer LUs assigned to class 2.
<b>INV=</b>	The name that you assigned to the Printer Inventory in the inventory attribute in the aopd.conf configuration file.
<b>CONSNAME=</b>	The name of an MCS or extended MCS console where you want NetSpool to display its unsolicited messages.
<b>LANGUAGE=</b>	The language that this NetSpool task is to use for messages.
<b>TRACE=ON</b>	Specifies that NetSpool is to read the data set in the TRACEOPT DD statement to obtain the trace options.

## 6.7 Defining NetSpool user ID and LUs

- ❑ User ID for start procedure
  - `RDEFINE STARTED APIJPJCL.* STDATA(USER(AOPSTC) GROUP(AOPOPER))`
  - `SETROPTS RACLIST(STARTED) REFRESH`
- ❑ LU class
  - A number from 1 to 64
- ❑ NetSpool LUs
  - LUs start by LU class
  - NetSpool start procedure can specify LU class
    - `PARM='LUCLASS=`
- ❑ Multiple NetSpool tasks
  - Start different classes for each task

Figure 6-7 Customizing NetSpool procedures and classes

### Creating a user ID to assign to the NetSpool startup procedure

The user ID assigned to the NetSpool startup procedure must be defined to RACF with an OMVS segment, a home directory, and a default group that has an OMVS segment with a group identifier (GID). You can assign any user ID, for example you can use AOPSTC.

### Defining the NetSpool startup procedure to RACF

You must define a profile for the NetSpool startup procedure in the RACF STARTED class.

```
RDEFINE STARTED APIJPJCL.* STDATA(USER(AOPSTC) GROUP(AOPOPER))
SETROPTS RACLIST(STARTED) REFRESH
```

### NetSpool LUs

When NetSpool starts, it starts LUs according to LU class; therefore, specify the same LU class for all NetSpool LUs that you want to start at the same time.

If you start more than one NetSpool task, specify unique class values in the LUCLASS parameter in the startup procedure for each task. These class values correspond to LU class values specified in the printer definitions in the Printer Inventory. Unique class values cause each NetSpool started task to start different printer LUs.

AOIPDC3		General Printer Definition	
Command ==>			
Printer definition name . SPOOLB			
Description . Spool to SYSOUT Class B			(extend)
Location. . . Poughkeepsie			(extend)
	Component name	Custom values	
Section	(enter to browse)	(enter to browse)	
Allocation	=>	=> *	
Processing	=>	=> *	
NetSpool options	=>	=>	
NetSpool end-of-file	=>	=>	
IP PrintWay options	=>	=> *	
NetSpool LU name . SC65PR05 LU classes . . 1			(extend)
Spooling mode. . . 1 1. Line 2. Stream			

Figure 6-8 NetSpool LU definitions for a printer

### Grouping printer LUs

NetSpool lets you group NetSpool printer LUs into logical-unit classes. A class is identified by a number from 1 to 64. Reasons why you might want to group logical printers into classes include:

- ▶ To start classes of logical printers at different times. For example, you might want to process requests for one class of printers during the day and process requests for another class of printers during the night.
- ▶ To spread processing of different classes of logical printers over different address spaces. You might want to do this if you have a large number of logical printers.

## 6.8 Defining NetSpool printer LUs to VTAM

- ❑ To define a NetSpool printer LU to VTAM
  - Select NetSpool printer LU names
    - Considerations for IMS/CICS users of **telnet**
  - Create VTAM APPL statements for NetSpool LU name
    - NetSpool runs as a VTAM application maintaining separate
    - LU-LU sessions for each NetSpool printer LU
    - The secondary LU is the NetSpool printer LU
  - Define resources in VTAM applications
    - BIND parameters

- ❑ Specify NetSpool LU names in the printer definitions

```
NETSPOOL VBUILD TYPE=APPL
LUPRT001 APPL MODETAB=ISTINCLM,DLOGMOD=S3270,EAS=1,SESSLIM=YES
IMSPR002 APPL MODETAB=ISTINCLM,DLOGMOD=SCS,EAS=1,SESSLIM=YES
```

Figure 6-9 Defining NetSpool printer LUs to VTAM

### Selecting NetSpool printer LU names

Each installation determines how many NetSpool printer logical units (LUs) to define. A sample configuration might be to define one NetSpool printer LU for each SNA-network printer to be replaced in your installation. Alternatively, you could define larger or smaller number of NetSpool logical printers than you have SNA-network printers.

Each NetSpool printer LU is identified with a 1 to 8 character alphanumeric name. You specify this LU name in the printer definitions and in the application-program LU name you use to define the NetSpool printer LU to VTAM.

### Naming considerations for IMS/CICS users of telnet

In IMS or CICS subsystems, applications receive communication services from terminal-management functions of subsystems. Thus, the IMS programmer thinks of a device as an LTERM, and the CICS programmer thinks of a device as a TCTTE entry.

Many IMS and CICS applications have algorithms that derive LTERM or TCTTE names for printers by doing some hashing technique on the input LTERM or TCTTE names. If NetSpool is used with Telnet, this creates problems because Telnet assigns the input LU name (and thereby the LTERM or TCTTE name) arbitrarily from a pool of LU names. If this type of algorithm is in use in your installation, and you are using Telnet, IBM recommends that you use the IP-LU mapping feature of Telnet. This feature enables you to specify the input LU name rather than having it randomly selected from a pool. This would, in turn, permit the

application to select the LU name for the printer. See the LUMAP statement in the Telnet profile description in *z/OS Communications Server: IP Configuration Reference*, SC31-8776.

## Creating APPL statements

NetSpool runs as a VTAM application program, maintaining separate LU- LU sessions for each NetSpool printer LU. Each LU-LU session is between:

- ▶ A primary LU, which is the VTAM application that initiates the VTAM session and sends print requests
- ▶ A secondary LU, which is the NetSpool printer LU

You must create a VTAM APPL definition statement in the SYS1.VTAMLST data set for each NetSpool printer LU. Create the APPL statement in a new or existing application major node definition.

```
NETSP00L  VBUILD TYPE=APPL
LUPRT001  APPL  MODETAB=ISTINCLM,DLOGMOD=S3270,EAS=1,SESSLIM=YES
IMSPR002  APPL  ACBNAME=LUPRT002,MODETAB=ISTINCLM,DLOGMOD=SCS,EAS=1,SESSLIM=YES
```

## Specifying correct BIND parameters

The VTAM BIND request that the VTAM application, the primary LU, sends to establish a session with a NetSpool printer LU, the secondary LU, must result in correct BIND parameters. That is:

- ▶ The LU type must be consistent with the type of print data to be sent during the session.
- ▶ The FM Profile and TS Profile values must be valid for that LU type.

Data type	LU type	FM profile	TS profile
3270 data	LU 0	2	2
3270 data	LU 3	3	3
SCS	LU 1	3 or 4	3 or 4

## SNA architecture

SNA architecture defines six control layers. The functions of the SNA layers are:

- ▶ Data link control (DLC): Defines several protocols for communication.
- ▶ Path control: - Performs many network layer functions, including routing, segmentation, and reassembly.
- ▶ Transmission control: Provides a reliable end-to-end communication service, as well as encryption/decryption services.
- ▶ Data flow control: Manages request and respond processing, determines whose turn it is to communicate, groups messages, and interrupts data flow on request.
- ▶ Presentation service: Specifies the data transmission algorithms that translate data format, coordinate resource sharing, and synchronize transaction operations.
- ▶ Transaction services: Provides application services in the form of programs that implement distributed processing or management services.

Logical Unit (LU) is a port through which a user accesses the SNA network in order to communicate with other users and the functions provided by the system services control point (SSCP). When a client LU wants to start a session with a server LU, it generates a BIND request.

BIND options in SNA transmission control contain all the information needed by the two ends of the LU-LU session for effective data exchange. A relatively small number of Presentation Service (PS) profiles, Function Management (FM) profiles, and Transmission Services (TS) profiles have evolved that are commonly used subsets of the available Bind options. Each profile identifies a prescribed subset of the SNA architecture:

- ▶ TS profile and TS usage fields: Specify facilities that are primarily in the transmission control layer. These include pacing counts, maximum RU size, and information as to whether sequence numbers (or ids) and certain TC commands will be used.
- ▶ FM profile and FM usage fields: Specify facilities that are primarily in the data-flow control layer. These include the request/response mode, the send/receive mode (duplex or half-duplex), chaining, brackets, and the allowable data-flow control commands.
- ▶ PS profile and PS usage fields: Specify the characteristics of the presentation services in each half-session.

### **VTAM applications**

VTAM applications that establish sessions with NetSpool printer LUs can either use the BIND parameters in the logon mode table entry referred to in the APPL statement for the NetSpool printer LU, or the applications can override the BIND parameters, using their own resource definitions. In either case, the BIND parameters must be correct.

In most cases, CICS and IMS subsystems override the BIND parameters associated with the NetSpool printer LU with BIND parameters that are consistent with the type of data in the print requests. However, in one case, when an IMS application sends a BIND request, and the NetSpool printer LU logical printer is defined in IMS as a non-SNA 3270 device, IMS uses the BIND parameters specified for the NetSpool printer LU, without any changes. Therefore, in this instance, the BIND parameters for the NetSpool printer LU must be correct for a 3270 data stream.

## 6.9 Defining NetSpool printers

General Printer Definition		
AOPIPDC3 Command ==> _____		
Printer definition name . IAZFSS _____		
Description . Shredder _____		(extend)
Location. . . ITS0 Sysplex _____		(extend)
Section	Component name (enter to list)	Custom values (enter to customize)
Allocation	=> AL-C-ANYLOCAL _____	=> _____
Processing	=> _____	=> *
NetSpool options	=> NS-Opts-NoFmt _____	=> _____
NetSpool end-of-file	=> _____	=> _____
IP PrintWay options	=> OPT-COMMON _____	=> *
NetSpool LU name . SC43PR02		LU classes . . 1 _____ (extend)
Spooling mode. . . 1. Line 2. Stream		
AOIPDNO NetSpool Options		
Command ==> _____		
Printer definition name . IAZFSS _____		
No Formatting		
Formatting . . . . 1 1. None 2. Convert to line 3. Convert to PCL		
Record size . . 255		
RECFM . . . . 1 1. VB 2. VBA 3. VBM		
Default owner. . . ITS0		
Embedded attributes prefix . . <<ibmjbatr _____		

Figure 6-10 Defining NetSpool printers and options

### Define NetSpool printers

You can configure printer definitions for use by NetSpool before or after starting the NetSpool program. If NetSpool is already started, NetSpool automatically starts the printer LU when you save the printer definition, provided that the printer LU is assigned to one of the LU classes that NetSpool has started. If the printer LU is not active in VTAM, NetSpool automatically starts it when the printer LU becomes active.

### NetSpool LU name

The logical unit (LU) name that NetSpool uses to identify a printer must be a unique LU name within the Printer Inventory. This field is required for NetSpool to start a session with this printer. This name must match an LU name specified in the ACBNAME field of the VTAM APPL definition statements.

### NetSpool LU name in printer definition

In each printer definition, you must specify the NetSpool printer logical unit (LU) name that you want to associate with the printer. NetSpool uses this name to establish a session with VTAM applications. This name is the VTAM secondary LU (SLU) name and must match the name in the ACB parameter of the APPL statement.

**Note:** When you name logical printers, you can do one of these without changing the primary LU (PLU) resource definitions of your VTAM applications:

- ▶ Assign NetSpool logical-printer names that are the same as the printer names currently defined in the resource definitions of the VTAM applications. For example, if the printer LU name used by a VTAM application is IMSPR001, the NetSpool LU name could be IMSPR001.
- ▶ Assign NetSpool logical-printer names that are different from the printer names used by the VTAM application. You might want to do this to make the NetSpool LU names more meaningful to your operators. For example, if the printer LU name used by your VTAM application is IMSPR001, the NetSpool LU name could be LUPRT001.

## NetSpool Options

The panel shown in Figure 6-10 lets you view and customize NetSpool attributes for a printer definition. The fields on this panel include the printer definition name and printer attributes that NetSpool uses.

**Embedded attributes prefix** - This field identifies job attributes embedded in the print data. For information about how to embed job attributes in VTAM application print data, see *z/OS Infoprint Server User's Guide*, S544-5746. NetSpool uses the embedded job attributes when it allocates an output data set on the spool. In the example in Figure 6-10, <<ibmjobattr is specified.

**Record size** - The maximum record size for the variable-length records that NetSpool writes to the output data set when the formatting option is None. This field is not required; if you leave this field blank, the default value is 32752. The range of values is 1 to 32752.

**Note:** If the length of the data in the input Request Unit (RU) is less than the maximum record size, NetSpool writes one record.

If the length of the data in the input RU is greater than the maximum record size, NetSpool writes multiple records.

If you select formatting option Convert to line or Convert to PCL, do not specify a value in this field.



## 6.10 Defining printer pools

```
A0PIPOLI                                     Pool List                                Row 1 to 1 of 1
Command ==> _____                                Scroll ==> HALF

Actions:
A-Add B-Browse C-Copy D-Delete E-Edit
A Pool Name      LU Name  Description
=====
b email          SC43PR05 Nastygram
***** Bottom of data *****

-----

A0PIPOAD                                     Printer Pool
Command ==> _____

Pool name . . . email
LU name . . . SC43PR05
Description . Nastygram

LU classes. . 1                                (extend)
NetSpool end-of-file component . . .           (list)
Default owner. . . . .
Embedded attributes prefix . . . . .

Printer definition names . . POK1              IAZFSS              (list)

More: +
```

Figure 6-11 Printer pool panels

### Using printer pools

A printer pool definition lets you broadcast data, that is, print the same data to several printers at the same time. You can also e-mail the data at the same time you send it to a printer. Only NetSpool supports this function; therefore, only VTAM applications can broadcast data. To be able to do this, the administrator creates a printer pool definition in the Printer Inventory. When VTAM application data is submitted to a printer pool definition, NetSpool creates multiple output data sets on the JES spool.

### Printer Pool panel

On this panel, you specify the names of the printer definitions in the pool. The printer definition identifies the target printer or e-mail destination. NetSpool creates a separate output data set on the JES spool for each printer definition, using the attributes in the Allocation section of each printer definition.

In the individual printer definitions in the list, you do not need to specify an LU name, LU class, or end-of-file rules because you specify them in the printer pool definition. However, if you want to specify NetSpool formatting values, you must specify them in the first printer definition in the list. NetSpool uses these fields in the first printer definition and ignores them in subsequent printer definitions:

- ▶ All fields in the NetSpool Options section
- ▶ SCSCConversion and NetSpool PCL Conversion fields in the Processing section
- ▶ SEGMENT and HOLD fields in the Allocation section

## LU classes

The logical-unit (LU) class for this NetSpool LU. If you specify more than one class, the NetSpool LU is assigned to all of the specified classes. This field is not required; the default value is class 1. To extend the length of this field, place the cursor on the word “extend” and press Enter.

**Note:** When NetSpool starts, it starts LUs according to LU class; therefore, specify the same LU class for all NetSpool LUs that you want to start at the same time.

## NetSpool end-of-file component

The name of a NetSpool end-of-file component. This component contains attributes (such as End of bracket) that NetSpool uses to determine the end of a file. This field is not required; if you do not specify a component name, NetSpool uses default values.

**Note:** To list existing NetSpool end-of-file components, place the cursor on the word “list” and press Enter (except in Browse Mode); from the list, you can also create and edit components.

## Default owner

The default Infoprint Server job owner for the associated logical unit if the print data does not specify an owner. The job owner is used for output data sets created for this NetSpool LU. If this field is blank, the default job owner is the user ID of the NetSpool daemon AOPNETD.

**Note:** The Infoprint Server job owner helps you find jobs in z/OS Infoprint Central. The JES job owner is always the user ID of the NetSpool task. The job owner in this field is also used as the JES job name if no other owner or job name is specified in the print data.

## Embedded attributes prefix

The prefix that identifies job attributes embedded in the print data. An example is <<ibmjjobattr.

## Printer definition names

The names of one or more printer definitions. The printer definitions must exist in the Printer Inventory. NetSpool creates an output data set on the JES spool for each printer definition. At least one printer definition name is required. To list existing printer definitions, place the cursor on the word “list” and press Enter. From this list, you can select, create, and edit printer definitions.

**Note:** NetSpool uses the following fields in the first printer definition in the list; NetSpool ignores these fields in subsequent printer definitions:

- ▶ All fields in the NetSpool Options section
- ▶ SCS Conversion and NetSpool PCL Conversion fields in the Processing section
- ▶ SEGMENT and HOLD fields in the Allocation section

## 6.11 Writing NetSpool exits

- ❑ Writing NetSpool exit and filter programs
  - Beginning of File exit for SCS data streams (APIPPTD1)
    - NetSpool calls this exit each time it creates a new output data set
  - Beginning of File exit for 3270 data streams (APIUBF3)
    - NetSpool calls this exit when it finds the Transparent Data control (TRN) in the input data stream
  - Transparent Data Control exit for SCS data streams (APIPPTD2)
    - NetSpool calls this exit when it finds the Transparent Data control (TRN) in the input data stream
  - Graphic Escape exit for SCS and 3270 data streams (APIUGEX)
    - NetSpool calls this exit when it finds a Graphic Escape SCS code, X'08', in the input data stream

Figure 6-12 NetSpool user exits

### Writing NetSpool exit and filter programs

NetSpool provides these exits, which let you customize how NetSpool processes SCS and 3270 data streams:

- ▶ The Beginning of File exit for SCS data streams (APIPPTD1). NetSpool calls this exit each time it creates a new output data set.
- ▶ The Beginning of File exit for 3270 data streams (APIUBF3). NetSpool calls this exit each time it creates a new output data set.
- ▶ The Transparent Data Control exit for SCS data streams (APIPPTD2). NetSpool calls this exit when it finds the Transparent Data control (TRN) in the input data stream.
- ▶ The Graphic Escape exit for SCS and 3270 data streams (APIUGEX). NetSpool calls this exit when it finds a Graphic Escape SCS code, X'08', in the input data stream.

In addition to writing NetSpool exits, you can also write filter programs to modify the data stream or add separator pages. When you write a filter program and specify it in the printer definition, Print Interface and IP PrintWay can also use it. However, a filter program is called only once for any data stream. NetSpool calls filters after the SCS or 3270 data stream has been converted to either line data or PCL data.

### Customizing JES to specify no blank truncation

JES, by default, truncates EBCDIC blank characters (X'40' characters) from the end of each record on the JES spool to save spool space. If you print XML data through PSF without first

transforming the XML data to AFP data, you should customize JES so it does not truncate blank characters.

To request no blank truncation, specify these JES parameters:

- ▶ JES2: BLNKTRNC=NO on the JES2 OUTCLASS statement. The default is BLNKTRNC=YES.
- ▶ JES3: TRUNC=NO on the JES3 SYSOUT statement. The default is TRUNC=YES. However, you can change the default in the TRUNC parameter of the JES3 BUFFER statement.

For example, if PSF processes XML data sets that are in output class X, specify BLNKTRNC=NO (in JES2) and TRUNC=NO (in JES3) for class X.

For more information, see:

- ▶ *z/OS JES2 Initialization and Tuning Reference* for information about the JES2 OUTCLASS statement.
- ▶ *z/OS JES3 Initialization and Tuning Reference* for more information about the JES3 SYSOUT statement.

**Note:** When you use IP PrintWay instead of PSF to send data to the printer, you can request either JES blank truncation (to save spool space) or JES no blank truncation. This is because any truncated characters are restored before data is printed.

## 6.12 Starting and stopping NetSpool

### ❑ NetSpool start command

```
START procedure_name[.id] [,JOBNAME=jobname] [,LUCLASS=lu_class ]
```

### ❑ Use the MVS MODIFY (F) and STOP (P) commands

- The MVS STOP command provides the same function as the KILL command
  - Use QUIT option after all current sessions end
  - Use FORCE option if some output data sets are incomplete

```
F jobname[.id],NetSpool_command
F jobname[.id],KILL
P jobname[.id]
F jobname[.id],QUIT
F jobname[.id],QUIT FORCE
```

Figure 6-13 Starting and stopping NetSpool commands

### Starting the NetSpool task

Each NetSpool started task runs in its own system address space and processes data sets for different classes of NetSpool printer LUs. When a NetSpool task starts, it attempts to start any printer LUs defined in the Printer Inventory and assigned to one of the started LU classes specified in the NetSpool startup procedure. If the administrator defines a new printer LU in the Printer Inventory after you start the NetSpool task, NetSpool automatically starts that printer LU if it is in one of the started LU classes. Before you begin:

- ▶ Make sure that the Printer Inventory Manager and NetSpool have been customized.
- ▶ Start the Printer Inventory Manager daemon (aopd) and the NetSpool daemon (aopnetd).
- ▶ Make sure that VTAM is started, and then activate any printer LUs defined in the Printer Inventory in the LU classes NetSpool will process. Use the VTAM VARY ACT command to activate the printer LUs.

If NetSpool cannot start a printer LU because the printer LU is inactive in VTAM or because it is started by another NetSpool started task, NetSpool starts the printer LU automatically when the printer LU becomes available.

### MVS START command

The following command is the syntax to start the NetSpool address space. The NetSpool startup procedure is a member named APIJPJCL in the SYS1.PROCLIB library, enter this command. You can specify a jobname for each NetSpool task that you start. A jobname lets

you easily distinguish between different NetSpool started task. You can specify values for the symbolic parameters defined in the APIJPJCL procedure.

```
START procedure_name[.id][,JOBNAME=jobname][,LUCLASS=lu_class]
```

After you start NetSpool, to display the status of the printer LUs that NetSpool started or attempted to start. use either:

- Infoprint Central

**Note:** You cannot use Infoprint Central to display LU status if your installation starts more than one NetSpool task.

- The NetSpool DISPLAY SELECTED operands on the **F jobname** command

## Stopping NetSpool

You can enter one of several NetSpool operator commands to stop a NetSpool started task, depending on how quickly you want to stop NetSpool. You can stop NetSpool:

- After all VTAM sessions end normally.
- Immediately, after NetSpool ends all VTAM sessions.
- Immediately, without ending VTAM sessions. Use this method only if other methods fail.

The VTAM **HALT** command also can cause sessions with NetSpool printer LUs to end or can stop the NetSpool task.

## Stopping NetSpool after current sessions end normally

Use the **QUIT** command option to stop the NetSpool started task after all current sessions end normally. The **QUIT** command prevents new sessions from starting. Each current session ends only when the VTAM application that established the session ends it. Thus, a long delay might occur before NetSpool stops and returns to the operating system.

```
F jobname[.id],QUIT
```

## Stopping NetSpool immediately

Use the **QUIT FORCE** command to stop the NetSpool started task after ending all current sessions immediately. Because NetSpool ends all current sessions immediately, some output data sets might be incomplete. NetSpool creates data sets with all data received.

```
F jobname[.id],QUIT FORCE
```

## Stopping NetSpool abnormally

Use the **KILL** option or MVS **STOP** command to stop the NetSpool started task immediately, without ending any VTAM sessions or creating output data sets. Data already sent to NetSpool is lost.

```
F jobname[.id],KILL
```

The MVS **STOP** command provides the same function as the **KILL** command.

```
P jobname[.id]
```

**Attention:** Use the **KILL** and **STOP** commands only when a hang condition prevents completion of a **QUIT FORCE** command.

## 6.13 NetSpool operator commands

- ❑ {F | MODIFY} jobname[.id],NetSpool\_command
  - Starting NetSpool printer LUs
    - VARY NET,ACT,ID=lu-name
    - F jobname[.id],LUNAME=lu-name,ADD
  - Stopping NetSpool printer LUs
    - VARY NET,INACT,ID=lu-name,TYPE=IMMED
    - F jobname[.id],LUNAME=lu-name,DEL
    - F jobname[.id],LUNAME=lu-name,PURGE
  - Displaying the status of NetSpool printer LUs
    - F jobname[.id],DISPLAY {LUNAME=luname | SELECTED | STARTED | WAITING | RACE}

Figure 6-14 NetSpool operator commands

### Entering NetSpool commands

To control NetSpool printer LUs, you can use the **F jobname** command to direct NetSpool commands to NetSpool. You can also use the MVS **STOP** command to stop a NetSpool started task.

The MVS **MODIFY** and **STOP** commands use the jobname and identifier fields to direct the command to the appropriate NetSpool task. If only one NetSpool task is started on a system, or if you specified a different jobname for each started task the **JOBNAME** parameter of the **START** command, specify only the jobname field in the command. However, if you started multiple NetSpool tasks with the same jobname, use the identifier field to distinguish between them.

The format of the MVS **MODIFY** command is:

```
F jobname[.id],NetSpool_command
```

### Starting NetSpool printer LUs

When you start NetSpool, the NetSpool printer LUs that are assigned to the LU classes specified on the EXEC statement in the NetSpool startup procedure are started. After NetSpool is started, you can use either (1) Infoprint Central or (2) the VTAM VARY ACT and LUNAME ADD commands to start an LU. You might want to start an LU that is assigned to a NetSpool LU class that has not been started, or to restart an LU that you stopped.

```
VARY NET,ACT,ID=lu-name
```

```
F jobname[.id],LUNAME=lu-name,ADD
```

## Stopping NetSpool printer LUs

While NetSpool is running, you can use either:

1. Infoprint Central
2. The LUNAME PURGE command
3. The VTAM VARY INACT and LUNAME DEL commands to deactivate the VTAM session with a printer and to stop NetSpool from processing any more print requests for the printer LU

If you remove a printer LU name from a printer definition after NetSpool is started, NetSpool automatically stops the printer LU after the VTAM application ends the session. Therefore, you do not need to use stop the printer LU in this case.

```
VARY NET,INACT,ID=lu-name,TYPE=IMMED  
F jobname[.id],LUNAME=lu-name,DEL
```

NetSpool deletes the selected printer LU from its list of printers. Use the LUNAME DEL command without preceding it with a VARY INACT command only when the printer LU has not been successfully started. That is, the printer LU is in the WAITING state.

## Stopping an LU with the LUNAME PURGE command

The LUNAME PURGE command deactivates the printer LU in VTAM immediately and closes the printer LU, deleting it from NetSpool's list of selected printers. NetSpool creates an output data set with any data received from VTAM before you enter the LUNAME PURGE command. However, the VTAM session is ended as soon as the command is entered, and the printer LU does not receive any new data from VTAM.

The LUNAME PURGE command is useful when, because of a significant error with the printer LU, you want processing to stop immediately for that printer LU and you do not want to allow any more data to be sent to it.

```
F jobname.id,LUNAME=lu-name,DEL
```

## Displaying the status of NetSpool printer LUs

Use either (1) Infoprint Central or (2) the DISPLAY command to display the status of printer LUs.

The DISPLAY command returns these states:

<b>STARTED</b>	The printer LU is either in a session with a VTAM application or is ready to accept a request from a VTAM application to establish a session.
<b>WAITING</b>	<p>The printer LU has not been started because it is not available. NetSpool periodically (every 60 seconds) attempts to start the printer, automatically starting it when it becomes available. You do not receive a message on the console when NetSpool automatically starts a printer LU at a later time.</p> <p>Because NetSpool periodically attempts to start printer LUs in the WAITING state, you might want to either start or stop a printer LU that is in the WAITING state.</p> <p>To start a printer LU that is not active in VTAM, activate the printer LU with the VTAM <b>VARY ACT</b> command. When the printer becomes active, NetSpool automatically starts the printer LU.</p>



To stop a printer LU, enter the LUNAME DEL command. Because NetSpool has not successfully started the printer LU, you do not need to first vary the printer inactive in VTAM.

To start a printer LU that is started by another NetSpool task, first stop the printer in the other NetSpool started task, using the VTAM **VARY INACT** command and the LUNAME DEL command. Then, activate the printer LU in VTAM. This NetSpool task automatically starts it.

**PENDING CLOSE** The LUNAME DEL or LUNAME PURGE command has been issued for the printer LU. NetSpool will stop the printer LU after NetSpool creates an output data set with all data received *before* the LUNAME DEL or LUNAME PURGE command was entered.

After the printer LU stops, it no longer is displayed when you enter the DISPLAY command for this NetSpool started task. At that point, you can restart the printer LU by activating the printer LU with the VTAM **VARY ACT** command and then adding the printer LU to NetSpool using the LUNAME ADD command.

The format of the DISPLAY command is:

```
F jobname[.id],DISPLAY {LUNAME=lu-name|SELECTED|STARTED|WAITING|TRACE}
```

Where:

<b>LUNAME=lu-name</b>	Requests that NetSpool display the status for the printer LU.
<b>SELECTED</b>	Requests that NetSpool display the printer LUs that are selected for processing. The status of the printers is <b>STARTED</b> , <b>WAITING</b> , or <b>PENDING CLOSE</b> . The minimum abbreviation allowed is <b>SEL</b> .
<b>STARTED</b>	Requests that NetSpool display all printer LUs in the <b>STARTED</b> state. The minimum abbreviation allowed is <b>STA</b> .
<b>WAITING</b>	Requests that NetSpool display all printer LUs in the <b>WAITING</b> state. The minimum abbreviation allowed is <b>WAIT</b> .
<b>TRACE</b>	Requests that NetSpool display all printer LUs that are being traced and whether internal tracing is active.

An example follows:

```
F NETSPPOOL,DISPLAY SELECTED
API1008I NETSPPOOL Display of SELECTED LUs
API1002I NETSPPOOL SC43PR05 - LU is STARTED, SESSION = INACTIVE, 137
POOLNAME = email, LUTYPE = 1, PLU = SC43RM02, EOFRULE = EB
API1002I NETSPPOOL SC43PR01 - LU is STARTED, SESSION = INACTIVE, PRINTERNAME =
IAZFSSL
API1002I NETSPPOOL SC43PR02 - LU is STARTED, SESSION = INACTIVE, 139
PRINTERNAME = IAZFSS, LUTYPE = 1, PLU = SC43RM02, EOFRULE = EB
API1002I NETSPPOOL SC43PR04 - LU is STARTED, SESSION = INACTIVE, PRINTERNAME =
ILPO
API1002I NETSPPOOL SC43PR03 - LU is STARTED, SESSION = INACTIVE, PRINTERNAME =
FIKE
```

Archived

# Infoprint Central

Infoprint Central is a Web-based print management system primarily for help desk operators. However, authorized job submitters can also use it. Infoprint Central requires the z/OS HTTP Server and a Web browser. No applications other than a Web browser need to be installed on users' workstations. Supported browsers include Microsoft® Internet Explorer® 5.5 (and later), Netscape Navigator 7.0 (and later), and IBM Home Page Reader 4.0 (and later).

With Infoprint Central, you can:

- ▶ **Work with print jobs:** You can find and work with any print jobs that are on the JES spool. You see more information about print jobs that Infoprint Server processes. For example, you can see whether an Infoprint Server print job completed successfully and where it printed, even if the print job is no longer on the JES spool.

You can use several different search criteria to find print jobs. After you find a print job, you can delete, hold, release, move, or change the priority of the print job. And, you can see all messages from Infoprint Server for that one print job.

- ▶ **Work with printers:** You can find and work with printers that are defined in the Printer Inventory, including IBM AFP printers that PSF controls (called PSF printers), and TCP/IP-attached and SNA-attached printers to which IP PrintWay extended mode sends print jobs (called IP PrintWay printers). You can find and work with IP PrintWay printers only when you run IP PrintWay extended mode.

You can use a variety of search criteria to find printers. After you find a printer, you can see its status, the printer's job queue, and other properties of the printer. Also, you can do these printer actions:

- PSF printers: Start, stop, space, interrupt, pause (JES2 only), ping, turn online, turn offline, and reset. Also, you can change forms and other job-selection criteria, and you can link to the printer's Web page.
- IP PrintWay printers: Start, stop, redirect, restore, ping, turn online, turn offline, and reset. Also, you can see all the messages from Infoprint Server for that one printer, and you can link to the printer's Web page.

Some printer actions (ping, turn online, turn offline, and reset) are available only for TCP/IP-attached printers.

- **Work with NetSpool logical units:** You can work with any NetSpool logical units (LUs) that are defined to NetSpool in the Printer Inventory and also defined to VTAM.

After you find a NetSpool LU by name, you can start and stop it. When you start an LU, that LU is started in both Infoprint Server and VTAM. Likewise, when you stop an LU, that LU is stopped in both Infoprint Server and VTAM. You can also display information about the current VTAM session established with a NetSpool LU.

You cannot work with NetSpool LUs if more than one NetSpool task uses the same Printer Inventory. Instead, you must use NetSpool commands and VTAM commands.

- **Display printer definitions:** You can display the properties of any printer definition in the Printer Inventory. You can use a variety of search criteria to find printer definitions, including the printer definition name and the printer's location. This can help you find the name of a printer in your area.
- **Check system status:** You can see the status of all Infoprint Server daemons and tasks. This can help you determine the cause of a printing problem. You can also start and stop IP PrintWay job selection rules to change which print jobs IP PrintWay processes.

Additional functions that Infoprint Central provides are:

- **Security:** You can use the HTTP Server protection facility to restrict access to Infoprint Central. In addition, you can use RACF to restrict access to print jobs, IP PrintWay printers, PSF printers, NetSpool logical units, IP PrintWay job selection rules, and Infoprint Server daemons.
- **Auditing:** You can find out who used Infoprint Central to perform actions on print jobs, IP PrintWay printers, PSF printers, and NetSpool LUs. For example, a message in the log for an Infoprint Server print job identifies who deleted the print job.

When you log on to Infoprint Central, you can log on to only one z/OS system. You can see all the print jobs, printer queues, and Infoprint Server messages for that one z/OS system only. To see print jobs, printer queues, and messages for more than one system, you can open separate instances of your Web browser and log on to several z/OS systems at a time.

## 7.1 Infoprint Central and Infoprint Server

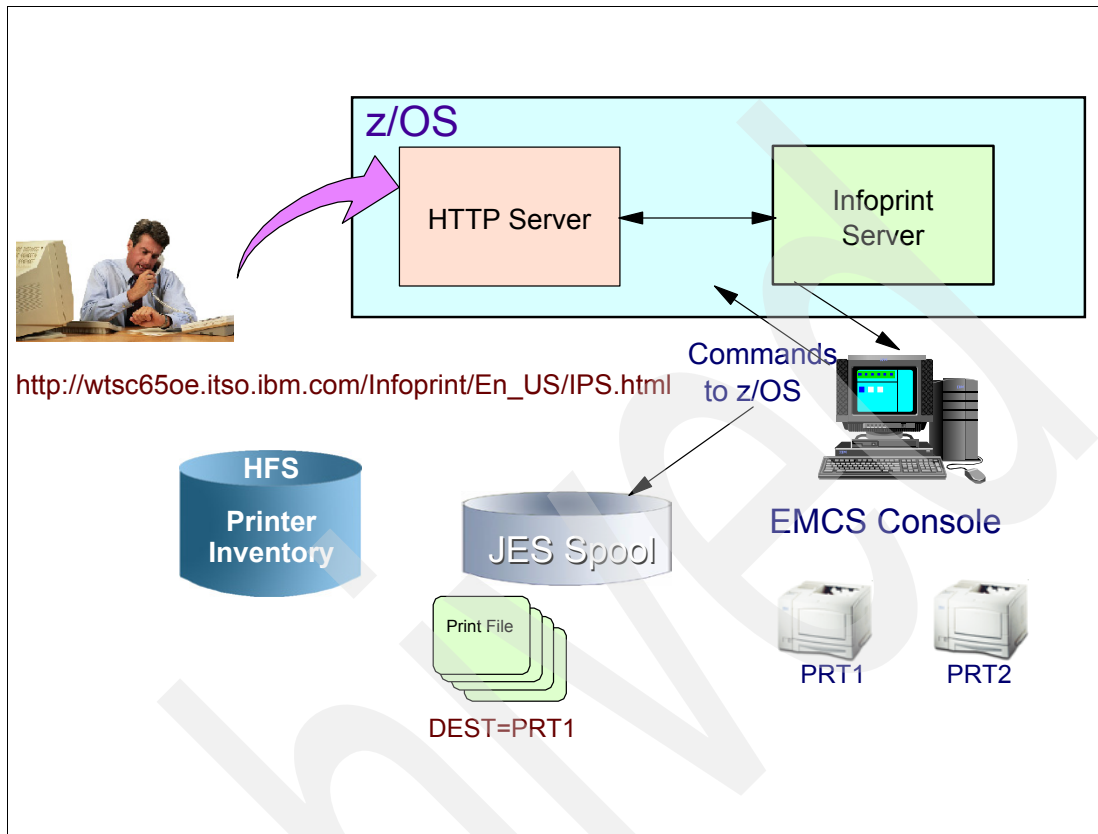


Figure 7-1 Infoprint Central access to Infoprint Server printers and Printer Inventory

### Infoprint Central and z/OS HTTP Server

Infoprint Central is a Web-based print management system primarily for help desk operators. However, authorized job submitters can also use it. Infoprint Central requires the z/OS HTTP Server and a Web browser. To log on to Infoprint Central, enter a URL in the browser:

- ▶ For the English version: [http://hostname:port/Infoprint/En\\_US/IPS.html](http://hostname:port/Infoprint/En_US/IPS.html)  
If the HTTP Server uses Secure Sockets Layer (SSL):  
[https://hostname:port/Infoprint/En\\_US/IPS.html](https://hostname:port/Infoprint/En_US/IPS.html)
- ▶ For the Japanese version: [http://hostname:port/Infoprint/Ja\\_JP/IPS.html](http://hostname:port/Infoprint/Ja_JP/IPS.html)  
If the HTTP Server uses Secure Sockets Layer (SSL):  
[https://hostname:port/Infoprint/Ja\\_JP/IPS.html](https://hostname:port/Infoprint/Ja_JP/IPS.html)

**Note:** To encrypt and decrypt information that passes between the user's browser and the HTTP Server, you can customize the HTTP Server to use the Secure Sockets Layer (SSL) protocol.

### EMCS console

Infoprint Central uses an extended MCS (EMCS) console to send commands to the z/OS system. The name of the console is defined in the aopd.conf configuration file, as shown in "aopd.conf configuration file parameters" on page 77.

## 7.2 Infoprint Central

- ❑ Infoprint Central primarily for:
  - Help desk operators
  - Other authorized users or job submitters
- ❑ Work with printers and IP PrintWay extended mode
- ❑ Single point of control for print management on z/OS
  - Requires z/OS HTTP Server
  - Infoprint Central supports these Web browsers:
    - Microsoft Internet Explorer 5.5 (or a higher level)
    - Netscape Navigator 7.0 (or a higher level)
    - IBM Home Page Reader 4.0 (or a higher level)
- ❑ Work objects
  - Print jobs
  - Printers
  - NetSpool Logical Units
  - System Status

Figure 7-2 Infoprint Central overview

### Infoprint Central

Infoprint Central is a Web-based print management system primarily for help desk operators. However, other authorized users or job submitters can also use it. It is primarily used by help desk operators. However, other authorized users and job submitters can also use Infoprint Central.

### Work with printers

You can find and work with printers that are defined in the Printer Inventory including IBM AFP printers controlled by PSF (called PSF printers), and TCP/IP-attached or SNA-attached printers to which IP PrintWay extended mode sends print jobs (called IP PrintWay printers). If you run IP PrintWay extended mode, use Infoprint Central because it lets you work with IP PrintWay extended mode printers. It also lets you work with output data sets that IP PrintWay extended mode is currently processing and displays the status of output data sets, including whether IP PrintWay has retained them.

### Print management on z/OS

Infoprint Central is a Web-based print management system. When you log on to Infoprint Central, you can log on to only one z/OS system. You can see all the print jobs, printer queues, and Infoprint Server messages for that one z/OS system only. To see print jobs, printer queues, and messages for more than one system, you can open separate instances of your Web browser and log on to several z/OS systems at a time. Infoprint Central requires the z/OS HTTP Server and a Web browser. No applications other than a Web browser need to be

installed on users' workstations. Supported browsers include Microsoft Internet Explorer 5.5 (and higher), Netscape Navigator 7.0 (and higher), and IBM Home Page Reader 4.0 (and higher).

### **Work with print jobs**

You can find and work with print jobs that are on the JES spool. Plus, you can see more information about print jobs that Infoprint Server processes. For example, you can see whether an Infoprint Server print job completed successfully and where it printed even if the print job is no longer on the JES spool. You can use several different search criteria to find print jobs. After you find a print job, you can delete, hold, release, move, or change the priority of the print job. And, you can see all messages from Infoprint Server for that one print job.

### **Work with NetSpool logical units**

You select the NetSpool logical units (LUs) you want to work with. You can work with any NetSpool LU that is defined in the Printer Inventory.

**Limitation:** If more than one NetSpool task is running on the z/OS system, you cannot use Infoprint Central to work with NetSpool LUs. Instead, you must use NetSpool commands and VTAM commands, which you can direct to the NetSpool task.

### **System status**

System status displays information about the Infoprint Server system. You can:

- ▶ Check whether Infoprint Server daemons and tasks are started.  
You cannot start and stop daemons or tasks from this panel. Ask your operator to start daemons with the **aopstart** command or AOPSTART procedure.
- ▶ Check which IP PrintWay job selection rules are in effect.  
You can start and stop job selection rules from this panel.

### **Help desk operators and authorized users**

Lets help desk operators and other authorized users or job submitters work with print jobs, printers, and NetSpool logical units (LUs); display printer definitions; and check system status. Infoprint Central is a Web-based print management system.

## 7.3 Customizing the z/OS HTTP Server configuration file

- ❑ **Sample configuration: /usr/lpp/internet/samples/config/C/httpd.conf**
  - **Add directives for Infoprint Central Web pages**
    - Add these Pass directives **before** the **ServerInit** and **Service** directives and **before** the **generic Pass directive, Pass /\***

```
Pass /Infoprint/Scripts/* /usr/lpp/Printsrv/InfoprintCentral/Scripts/*
Pass /Infoprint/Images/* /usr/lpp/Printsrv/InfoprintCentral/Images/*
Pass /Infoprint/help/En_US/* /usr/lpp/Printsrv/InfoprintCentral/help/En_US/*
Pass /Infoprint/En_US/*.html /usr/lpp/Printsrv/InfoprintCentral/html/En_US/*.html
ServerInit /usr/lpp/Printsrv/lib/aopcentral.so:initialize*
Service /Infoprint* /usr/lpp/Printsrv/lib/aopcentral.so:dllMain*
```
  - **Add directives for Infoprint Central Web pages**
    - Add Into the section that contains **AddType** directives

```
AddType .js application/x-javascript ebcodic 1.0 # JavaScript
AddType .htc text/x-component ebcodic 1.0 # .htc files
AddType .css text/css ebcodic 1.0 # cascading style sheets
```
  - **/usr/lpp/Printsrv/samples/httpd.conf.updates contains the directives**

Figure 7-3 Editing the z/OS HTTP Server configuration file

### Customizing the z/OS HTTP Server

You must customize and start the z/OS HTTP Server to display Infoprint Central Web pages.

Suggestion:

- ▶ Start a separate HTTP Server to be used exclusively by Infoprint Central. This can improve Infoprint Central performance. Also, it lets you customize the HTTP Server for Infoprint Central without affecting other applications that use the HTTP Server.
- ▶ Start an HTTP Server on each z/OS system where Infoprint Server is running. This is done because the HTTP Server can display Infoprint Central Web pages only for Infoprint Server running on the same z/OS system as the HTTP Server.
- ▶ Customize the HTTP Server to use the Secure Sockets Layer (SSL) protocol if you want to encrypt information that passes between the user's browser and the HTTP Server.

The z/OS HTTP Server configuration file, `httpd.conf`, contains directives that customize the HTTP Server. The default location of this configuration file is `/etc/httpd.conf`.

You must add directives so that the HTTP Server can display Infoprint Central Web pages. In addition, you can add directives to protect access to Infoprint Central Web pages and to tune the HTTP Server.

The sample `httpd.conf` is in `/usr/lpp/internet/samples/config/C`.



## Editing the z/OS HTTP Server configuration file

The Service, ServerInit, Pass, and AddType directives for Infoprint Central are required. Add the directives to the HTTP Server configuration file in the order shown.

- In the English version in the “Service Directives” section of the file, add these directives.

Add these Pass directives before the ServerInit and Service directives and before the generic Pass directive, Pass /\*. Do not add these Pass directives in the section indicated for Pass directives in the comments of the HTTP Server configuration file.

```
Pass /Infoprint/Scripts/* /usr/lpp/Printsrv/InfoprintCentral/Scripts/*
Pass /Infoprint/Images/* /usr/lpp/Printsrv/InfoprintCentral/Images/*
Pass /Infoprint/help/En_US/* /usr/lpp/Printsrv/InfoprintCentral/help/En_US/*
Pass /Infoprint/En_US/*.html /usr/lpp/Printsrv/InfoprintCentral/html/En_US/*.html
ServerInit /usr/lpp/Printsrv/lib/aopcentral.so:initialize*
Service /Infoprint* /usr/lpp/Printsrv/lib/aopcentral.so:dllMain*
```

- In the section that contains AddType directives, add these directives at the end of the section.

If any of these directives already exist, delete or comment out the existing directives so that the directives suitable for Infoprint Central are used. Notice that the .css directive is present in the default configuration file.

```
AddType .js application/x-javascript ebcidic 1.0 # JavaScript
AddType .htc text/x-component ebcidic 1.0 # .htc files
AddType .css text/css ebcidic 1.0 # cascading style sheets
```

**Note:** File /usr/lpp/Printsrv/samples/httpd.conf.updates contains the required directives. To avoid spelling errors, copy the directives from httpd.conf.updates to the HTTP Server configuration file.

### Important:

- Add the Pass directives *before* the ServerInit and Service directives.
- Do not add the Pass directives in the section indicated in the configuration file comments for Pass directives.
- If any of the AddType directives already exist, delete or comment out the existing directives so that the directives suitable for Infoprint Central are used. Notice that the .css directive is present in the default configuration file.

## 7.4 Protecting Infoprint Central Web pages

- ❑ Use a protection method that provides Infoprint Central with a unique z/OS user ID and password that has been authenticated by RACF or a similar security system

- Example: This directive lets only z/OS user IDs USER01 and USER02 use Infoprint Central Web pages

```
Protect /Infoprint* {  
  ServerSystem_Logon  
  UserId%%CLIENT%%  
  AuthtyBasic  
  %%SAF%%sswdFile  
  Mask    USER01,user01,USER02,user02  
}
```

- Notice that you should specify the user IDs with both uppercase and lowercase letters so that users can log on using either case

Figure 7-4 Protecting Infoprint Central Web pages

### Protecting Infoprint Central Web pages

The z/OS HTTP Server lets you use any of several methods to protect Web pages. You must use a protection method that provides Infoprint Central with a unique z/OS user ID and password that has been authenticated by RACF or a similar security system.

### Default UserID directive

The default UserID directive, `UserId %%CLIENT%%`, in the HTTP Server configuration file lets only users with a valid z/OS user ID use Web pages. This directive is suitable for Infoprint Central. As an alternative, you can specifically protect Infoprint Central Web pages by including a Protect directive for the Infoprint Central pages.

### Specifying the directive

Specify the Protect directive before the Pass and ServerInit directives for Infoprint Central. This directive lets only z/OS user IDs USER01 and USER02 use Infoprint Central Web pages. Notice that you should specify the user IDs with both uppercase and lowercase letters so that users can log on using either case:

```
Protect /Infoprint* {  
  ServerSystem_Logon  
  UserId%%CLIENT%%  
  AuthtyBasic  
  %%SAF%%sswdFile  
  Mask    USER01,user01,USER02,user02 }
```

## 7.5 z/OS HTTP Server environment variables file

- ❑ ICU\_DATA - refer to IBM XML Toolkit V1R4 libraries
  - /usr/lpp/ixm/IBM/xml4c-5\_5/lib
- ❑ LIBPATH - path used to locate dynamic link libraries (DLLs)
  - /usr/lpp/ixm/IBM/xml4c-5\_5/lib
  - /usr/lpp/ixm/IBM/xslt4c-1\_9/lib
  - /usr/lpp/Printsrv/lib
- ❑ NLSPATH - directory path containing message catalogs
  - /usr/lpp/Printsrv/%L/%N
  - /usr/lpp/Printsrv/En\_US/%N
- ❑ AOP\_READ\_COMMUNITY - SNMP community name
- ❑ AOP\_WRITE\_COMMUNITY - SNMP write community name
- ❑ AOPCENTRAL - directory for Infoprint Central files
  - Default: AOPCENTRAL=/usr/lpp/Printsrv/InfoprintCentral
- ❑ AOPCENTRAL\_CODEPAGE - Infoprint Central EBCDIC code page
- ❑ AOPCONF - Infoprint Server configuration file path name
  - Default: AOPCONF=/etc/Printsrv/aopd.conf
- ❑ AOPLIMIT - max number of objects to display
- ❑ CLASSPATH - full path names of Infoprint Central Java Archive files
- ❑ LANG - language used for messages
- ❑ LC\_ALL - locale for time and date formatting in messages
- ❑ LC\_CTYPE - locale for EBCDIC code page
- ❑ LC\_TIME - locale for time/date formatting in messages
- ❑ TZ - time zone used to format date and time in messages

Figure 7-5 z/OS HTTP Server environment variables file

### z/OS HTTP Server environment variables file

You must add or edit the environment variables required by the Infoprint Central in the z/OS HTTP Server environment variable file, `httpd.envvars`. If an environment variable already exists in the HTTP environment variables file, add the new values to the existing values, separating values with a colon.

#### Required variables

##### ICU\_DATA

Add this value to the existing values. This value refers to IBM XML Toolkit V1R4 libraries:

`/usr/lpp/ixm/IBM/xml4c-5_5/lib`

##### LIBPATH

The path used to locate dynamic link libraries (DLLs). This environment variable is required. Add these IBM XML Toolkit V1.8 library directories and the Infoprint Server library directory to any existing values:

`/usr/lpp/ixm/IBM/xml4c-5_5/lib`

`/usr/lpp/ixm/IBM/xslt4c-1_9/lib`

`/usr/lpp/Printsrv/lib`

Separate directories with colons.

Default: None

## NLSPATH

The path of directories that contain message catalogs. This environment variable is required. Add these directories for Infoprint Server messages to any existing values:

```
/usr/lpp/Printsrv/%L/%N  
/usr/lpp/Printsrv/En_US/%N
```

Separate values with colons.

Default: None

## Optional variables

### AOP\_READ\_COMMUNITY

The SNMP community name that allows read access to TCP/IP-attached printers in your installation. The SNMP read community name is required for Infoprint Central to display information from the printer (such as status, paper level, and model) and to display the turn online, turn offline, and reset printer actions. This environment variable is optional.

- For information about how to assign an SNMP read community name to a printer, see the documentation for the printer or contact the printer manufacturer.
- You should assign the same SNMP read community name to all printers and specify it in this environment variable. If a printer's SNMP read community name is different from the one specified in this environment variable, Infoprint Central does not display information from the printer and does not display the turn online, turn offline, and reset printer actions.
- To see information from the printer and printer actions, Infoprint Central users must have READ access to the RACF profiles in the PRINTSRV and OPERCMDS classes that protect the printer.
- Infoprint Central does not reveal this community name to Infoprint Central users.

Default: public

### AOP\_WRITE\_COMMUNITY

The SNMP write community name that allows write access to TCP/IP-attached printers in your installation. The SNMP write community name is required to turn printers online, offline, and reset them. This environment variable is optional.

- For information about how to assign an SNMP write community name to printers, see the documentation for the printer or contact the printer manufacturer.
- If you assign different SNMP write community names to your printers, specify the most common community name in this environment variable. If this community name is not correct for a particular printer, Infoprint Central asks the user to enter the SNMP write community name for that printer.
- To do printer actions, Infoprint Central users must have CONTROL access to the RACF profile in the PRINTSRV class that protects the printer.
- Infoprint Central does not reveal this community name to Infoprint Central users.

Default: public

### AOPCENTRAL

The directory that contains Infoprint Central files. This environment variable is optional. If you installed Infoprint Server files in the default directory, you do not need to set this environment variable.

Default: AOPCENTRAL=/usr/lpp/Printsrv/InfoprintCentral

### AOPCENTRAL\_CODEPAGE

The EBCDIC code page that Infoprint Central uses when it searches for and displays attributes in the Printer Inventory. Specify the same code page for the locale that is

specified in the LC\_ALL or LC\_CTYPE environment variable in the aopstart EXEC or AOPSTART JCL procedure. This environment variable is optional.

Specify this environment variable only in the z/OS HTTP Server environment variables file (httpd.envvars).

Default: IBM-1047

### **AOPCONF**

The full path name of the Infoprint Server configuration file. This environment variable is optional. If you created the configuration file in the default location, you do not need to set this environment variable. The values in the z/OS HTTP Server environment variables file and in the aopstart EXEC must match.

Default: AOPCONF=/etc/Printsrv/aopd.conf

### **AOPLIMIT**

The maximum number of objects that Infoprint Central displays after a search. This limit prevents the HTTP Server from timing out before Infoprint Central can display all objects that meet the users' search criteria. This environment variable is optional. You can specify a number from 1 to 9999. If you specify an incorrect value, the default value is used.

Consider the following recommendations:

- If Infoprint Central users ask to see more than 250 objects, try increasing the limit. If the higher limit causes the HTTP Server to time out, try increasing the HTTP Server's timeout value.
- If the HTTP Server times out occasionally with the default limit, ask users to narrow their searches.
- If the HTTP Server times out repeatedly with the default limit, lower the limit or try increasing the HTTP Server's timeout value.

Default: AOPLIMIT=250

### **CLASSPATH**

The full path names of Infoprint Central Java Archive (JAR) files. This environment variable is optional. However, you must set it if you did not install Infoprint Server in the default directory (/usr/lpp/Printsrv), or if you specify the CLASSPATH environment variable in the HTTP Server environment variables file.

If you did not install Infoprint Server in the default directory, add these directories to any existing values:

- /*directory*/classes/ipa.jar
- /*directory*/classes/modelplugin.jar
- /*directory*/classes/penguin.jar
- /*directory*/classes/snmp.jar

*directory* is the name of the directory where you installed Infoprint Server.

Separate directories with a colon, and specify the entire value on one line.

Default:

```
/usr/lpp/Printsrv/classes/ipa.jar:  
/usr/lpp/Printsrv/classes/modelplugin.jar:  
/usr/lpp/Printsrv/classes/penguin.jar:  
/usr/lpp/Printsrv/classes/snmp.jar
```

## **JAVA\_HOME**

The path used to locate Java files. This environment variable is optional. However, you must set it if you did not install the Java 1.4 files in the default Java directory.

Default: /usr/lpp/java/J1.4

## **LANG**

The language used for messages. Infoprint Server provides messages in English and Japanese. Specify En\_US for English messages, or Ja\_JP for Japanese messages.

Default: C (equivalent to En\_US)

## **LC\_ALL**

The locale used to format time and date information in messages and the language of messages. Specify the same LC\_ALL value in the environments for the Printer Inventory Manager and Infoprint Central. This locale overrides the locale in the LC\_TIME variable.

Default: C (also called POSIX)

## **LC\_CTYPE**

The locale that determines the EBCDIC code page used to validate Infoprint Server attribute values. Specify the same LC\_CTYPE value in the environments for the Printer Inventory Manager and Infoprint Central. Also, specify a comparable value in the Language field on the Infoprint Server ISPF Configuration panel.

LC\_ALL does not override LC\_CTYPE.

Default: C (also called POSIX)

## **LC\_TIME**

The locale used to format time and date information in messages. Specify the same LC\_TIME value in the environments for the Printer Inventory Manager and Infoprint Central.

Default: C (also called POSIX)

## **TZ**

The time zone used to format date and time information in messages displayed by Infoprint Central. Specify the same TZ value in the environments for the Printer Inventory Manager and Infoprint Central.

Default: The z/OS HTTP Server default is EST5EDT (Eastern U.S. time zone). The default for the Printer Inventory Manager is GMT0.

The AOPCONF, LC\_ALL, LC\_CTYPE, LC\_TIME, and TZ environment variables in the HTTP Server environment variables file must match the variables specified in the environment for the **aopstart** command in the STDENV data set for the AOPSTART JCL procedure.

## 7.6 z/OS HTTP Server JCL procedure

```
//IMWPROC  PROC  ICSPARM='-p 88 -r /web/printway/httpd.conf',
//          LEPARM='ENVAR("_CEE_ENVFILE=/web/printway/httpd.envvars")'
//*****
/* LEPARM ==> LE runtime opts
/* LEPARM='ENVAR("_CEE_ENVFILE=/etc/httpd.envvars")'
/* ICSPARM ==> HTTP Server parameters
/* # Standalone HTTPD
/* ICSPARM='-p 8080 -r /etc/httpd.conf'
/* # WLM Queue Manager
/* ICSPARM='-SN WEBSN1 -p 8080 -r /etc/httpd.conf'
/* # WLM ApplEnv Queue Server
/* ICSPARM='-SN WEBSN1 -AE WEBHTML'
/* Comments removed
//WEBSRV   EXEC PGM=IMWHTTPD,REGION=0K,TIME=NOLIMIT,
//          PARM=('&LEPARM/ICSPARM')
//*STEPLIB DD DSN=SYS1.LINKLIB,DISP=SHR
//SYSIN    DD DUMMY
//SYSPRINT DD SYSOUT=*
//SYSERR   DD SYSOUT=*
//STDOUT   DD SYSOUT=*
//STDERR   DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//CEEDUMP  DD SYSOUT=*
//*SYSDUMP DD DSNAME=<hlq>.WEBMDUMP.D&LYMMDD..T&LHHMSS,
//*          UNIT=SYSDA,SPACE=(CYL,(100,50)),DISP=(NEW,DELETE,CATLG),
//*          DCB=(LRECL=4160,BLKSIZE=24960,RECFM=FBS)
```

Figure 7-6 z/OS HTTP Server JCL procedure

### z/OS HTTP Server JCL procedure

Changes to the sample IMWPROC JCL are as follows:

- ▶ -p port-number - The z/OS HTTP Server listens on this port number. The default port number is 80. This flag overrides the Port directive specified in the configuration file. Port 88 will be used for Infoprint Central HTTP server.
- ▶ -r configuration-file - Specifies the file to use as the configuration file. You must use this flag if you want to start the server with a configuration file other than the default /etc/httpd.conf file.
  - r /web/printway/httpd.conf will be used for Infoprint Central HTTP server.
- ▶ ENVAR - The ENVAR option sets initial values for specified environment variables that the server is started with. Using ENVAR, you can pass switches or tagged information using standard z/OS UNIX functions. You may set additional environment variables using the \_CEE\_ENVFILE ENVAR option. The default \_CEE\_ENVFILE shipped with the server is /etc/httpd.envvars, which is compiled into the IMWHTTPD program. You can use the \_CEE\_ENVFILE ENVAR option to override this default. The environment variables are shipped with the server under /usr/lpp/internet/server\_root/samples/config.
  - \_CEE\_ENVFILE=/web/printway/httpd.envvars is used in Figure 7-6 for Infoprint Central HTTP server.

See *z/OS HTTP Server Planning, Installing, and Using*, SC34-4826 for more information.

## 7.7 Customization tasks for Infoprint Central

- ❑ Creating the search database
  - `sdbu -z` command creates - `/var/Printsrv/searchdb`
- ❑ Customizing PSF for z/OS to use the Printer Inventory
  - Create a PSF FSS/FSA definition in the Printer Inventory
  - PSF start procedure - PSF uses Printer Inventory
    - `// EXEC PGM=APSPPIEP,PARM=('INV=pi_name')`
- ❑ Customizing JES3 console (CONSTD statement)
  - Escape character must be defined if CONSTD statement  
EDIT parameter coded: `EDIT=(",#,,)`

Figure 7-7 Customizing Infoprint Central

### Creating the search database

Infoprint Server maintains a search database that lets Infoprint Central find objects in the Printer Inventory. You must run the z/OS UNIX `sdbu -z` command to create this search database. To use the `sdbu` command, you must be connected to the RACF AOPADMIN group or have an effective UID of 0. The Infoprint Server search database daemon (aopsdbd) must also be started. This daemon starts automatically when you run the `aopstart` command or the AOPSTART JCL procedure. To create the search database, perform the following steps:

1. If you are not connected to the required RACF groups, switch to an effective UID of 0. To use the `su` command, you must be permitted to the BPX.SUPERUSER profile in the FACILITY class in RACF.
2. Start Infoprint Server daemons if they are not already started.
3. Create the search database.

```
sdbu -z
```

The search database files in the `/var/Printsrv/searchdb` directory are created.

If the search database daemon (aopsdbd) fails, you must run the `sdbu` command again to recreate the search database.



## Customizing PSF for z/OS to use the Printer Inventory

If you want to use Infoprint Central to work with IBM AFP printers that PSF for z/OS controls (called PSF printers), you must customize PSF for z/OS to use the Printer Inventory.

To customize PSF for z/OS to use the Printer Inventory:

- ▶ Create a PSF FSS definition for each PSF functional subsystem in the Printer Inventory.
- ▶ Create a PSF FSA definition for each PSF printer in the Printer Inventory.
- ▶ Edit your PSF startup procedures to specify that PSF is to obtain printer information from the Printer Inventory.

To use the Printer Inventory with PSF change the printer startup procedure to specify INV=*piname* as the first parameter in the PARM field of the EXEC statement:

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

*piname* is the four-character name of the Printer Inventory.

The INV=*piname* parameter indicates that PSF uses the Printer Inventory and obtains parameters from the specified Printer Inventory for each printer in the startup procedure. No other parameters in the PARM field are used when PSF uses the Printer Inventory. Restart your PSF printers after you have started the Infoprint Central daemon (daemon aopssid). This is because Infoprint Server automatically enables SNMP reporting (SNMP reporting option in the PSF FSA definition) if the Infoprint Central daemon is started. You must restart PSF printers after SNMP reporting has been enabled.

- ▶ You can use the Infoprint Server migration program, aopmig, to create the required PSF FSS and FSA definitions in the Printer Inventory.
- ▶ You can use Infoprint Server ISPF panels or the Printer Inventory Definition Utility (PIDU) to create and edit FSS and FSA definitions.
- ▶ You do not need to create PSF printer definitions in the Printer Inventory to use Infoprint Central. Printer definitions are required only if you want to print to PSF printers using Print Interface or NetSpool.

## Customizing JES3 console standards (CONSTD statement)

The CONSTD statement in the JES3 initialization stream lets you define standards for your JES3 console configuration. The EDIT parameter of the CONSTD statement lets you specify special characters to be used in editing commands processed by JES3 console services. If you code the EDIT parameter on the CONSTD statement, you must define an escape character so that Infoprint Central can escape special characters in filenames. You can omit the EDIT parameter, in which case JES3 uses the default escape character, which is the quotation mark (").

To specify the quotation mark (") as the JES3 escape character and the pound sign (#) as the backspace character, code:

```
EDIT=(",#,,)
```

## 7.8 Customizing aopd.conf file and RACF for Infoprint Central

- ❑ Editing the Infoprint Server configuration file
  - `start-daemons = { ssid }`
  - `console-name = name`
    - Default name - AOPAOPC1
- ❑ Set up security for Infoprint Central
  - **PRINTSRV class:** Profiles in this class restrict who can work with IP PrintWay and PSF printers
    - Activate the PRINTSRV class
    - `SETOPTS CLASSACT(PRINTSRV) RACLIST(PRINTSRV)`
  - **OPERCMDs class:** Profiles in the OPERCMDs class restrict who can work with PSF printers

Figure 7-8 Infoprint Central customization and security

### Editing the Infoprint Server configuration file

To use Infoprint Central, you must edit the Infoprint Server configuration file, `aopd.conf`. Add or edit the following attributes in the `aopd.conf` configuration file:

- ▶ `start-daemons = { ssid }` - The daemons that start when you run the `aopstart` command or `AOPSTART JCL` procedure. To use Infoprint Central, you must start the `aopssid` daemon. Add the `ssid` value to any existing values in this attribute.

If you change this attribute while Infoprint Server is running, stop and restart all Infoprint Server daemons to pick up the change.

- ▶ `console-name = name` - The name of the extended MCS console that Infoprint Central uses to send commands to the z/OS system.
  - The console name must be 2 - 8 alphanumeric or national (#, \$, and @) characters. The first character cannot be numeric.
  - Do not use these reserved names: `HC`, `INSTREAM`, `INTERNAL`, `OPERLOG`, `SYSIOSRS`, `SYSLOG`, `UNKNOWN`, `SYSJ3D0x`, and `SYSJ3R0x`.
  - If you change this attribute while the z/OS HTTP Server is running, restart the HTTP Server to pick up the change.

**Default:** `AOPinventoryC`, where `inventory` is the name of the Printer Inventory specified in the `inventory` attribute in this Infoprint Server configuration file. If you use the default name for the Printer Inventory (`AOP1`), the default console name is `AOPAOP1C`. If you change this attribute while the `aopssid` daemon is running, stop that daemon and restart it.

## Setting up security for printers

Infoprint Central lets users work with printers that IP PrintWay controls (called IP PrintWay printers) and with printers that PSF for z/OS controls (called PSF printers). You can protect IP PrintWay and PSF printers with profiles in the PRINTSRV and OPERCMD classes.

### PRINTSRV class

Infoprint Server profiles in the PRINTSRV class restrict who can work with IP PrintWay and PSF printers.

To set up security for Infoprint Central, you can define profiles in the PRINTSRV class. If you define any profiles in the PRINTSRV class, you must activate the PRINTSRV class in RACF. In addition, to improve performance, you should copy profiles in the PRINTSRV class into virtual storage. If you copy profiles into virtual storage, you must refresh the PRINTSRV class after you change any profiles or permit new users to the profiles to make the changes effective. The SETROPTS command activates the PRINTSRV class and builds in-storage profiles.

```
SETROPTS CLASSACT(PRINTSRV) RACLIST(PRINTSRV)
```

### OPERCMDS class

Profiles in the OPERCMD class control the use of operator commands.

You can define profiles in the OPERCMD class to restrict an Infoprint Central user's actions on PSF printers. Profiles in the OPERCMD class also restrict access to JES commands.

You can create:

- ▶ A broad profile to protect actions on all PSF printers. You control the actions users can perform by giving them either READ or UPDATE access to the broad profile. Each type of access lets the user perform a different set of actions.
- ▶ Specific profiles to protect specific actions.

In JES3 environments, you can also define profiles that apply just to one printer. To do this, you specify the printer name in the last qualifier of the profile name.

If you have already defined profiles to protect printers in the OPERCMD class, these profiles should be suitable for Infoprint Central. However, be sure to give Infoprint Central users and the AOOPER group the required access to these existing profiles.

**Note:** If you define profiles in both classes, users must have access to both profiles to perform actions on PSF printers.

## Security profiles in Printer Inventory

You specify the profile name that applies to a printer in the Printer Inventory:

- ▶ **IP PrintWay printers:** Specify the profile name in the printer's printer definition.
- ▶ **PSF printers:** Specify the profile name in the printer's FSA definition.

### Recommendations:

- ▶ IP PrintWay printers: Define profiles in the PRINTSRV class to protect printers. Otherwise, any Infoprint Central user can work with any IP PrintWay printer.
- ▶ PSF printers: Define profiles in the PRINTSRV class to protect printers, or define profiles in the OPERCMD class to protect *printer actions*. Otherwise, any Infoprint Central user can work with any PSF printer.

## 7.9 Infoprint Central Security - Printer Inventory

- ❑ Define profiles in the PRINTSRV class
  - PERMIT AOP.ADMINISTRATOR CLASS(PRINTSRV) ACCESS(READ) ID(userid or groupid)
  - Who can work with printers
    - Define a separate profile to protect each printer
    - Define one profile to protect a group of printers
    - Define a profile for all printers
- ❑ Define profiles in Printer Inventory
  - IP PrintWay printers - Protocol panel
  - PSF printers - FSA definition panel



Read access to  
Printer Inventory

Figure 7-9 PRINTSRV class authorization for Infoprint Central users

### Authorizing users using Infoprint Central

Some Infoprint Central actions require that users be authorized to read the Printer Inventory. These actions include viewing PSF printers, printer definitions, printer pool definitions, and IP PrintWay job selection rules.

To authorize users to read the Printer Inventory, using either Infoprint Central or Infoprint Server ISPF panels, give the AOP.ADMINISTRATOR profile in the PRINTSRV class universal READ access or give all Infoprint Central users READ access.

```
RDEFINE PRINTSRV(AOP.ADMINISTRATOR) UACC(NONE)
PERMIT AOP.ADMINISTRATOR CLASS(PRINTSRV) ACCESS(READ) ID(userid or groupid)
SETROPTS RACLIST(PRINTSRV) REFRESH
```

### Access to printers

You can define profiles in the PRINTSRV class to restrict who can work with printers. Profiles in the PRINTSRV class can apply to both IP PrintWay and PSF printers. You can define a separate profile to protect each printer, or you can define one profile to protect a group of printers or all printers. For example, if you want to authorize different users to work with printers in different locations, define separate profiles for printers in each location. You could define one profile for all printers in one location, and another profile for all printers in another location. If you want to authorize the same group of users to work with all printers, you need to define only one profile.

## 7.10 Infoprint Central Security - Protecting IP PrintWay printers

```
Edit
Command ==> _____ LPR Protocol

Printer definition name . POK45ANE _____
Operator security profile
. . . POK.ROOM1 _____

Printer IP address . 9.12.6.3 _____ (extend)
Print queue name . . afccu2 _____ (extend)
```

Figure 7-10 Protecting access to IP PrintWay printers with security profiles

### Defining Infoprint Central users as z/OS UNIX users

You must define users of Infoprint Central to RACF as z/OS UNIX users. Infoprint Central users do not need to be authorized to use TSO. If you give users temporary RACF passwords, tell users how to rlogin or telnet to the z/OS system to change the password the first time. The z/OS HTTP Server does not let users change passwords from the logon screen.

### Security profile definitions to IP PrintWay printers

You can define a separate profile to protect each printer, or you can define one profile to protect a group of printers or all printers. In Figure 7-10, the POK.ROOM1 security profile definition is for a group of printers in ROOM1 and POK is just a name for the location. If you want to authorize different users to work with printers in different locations, define separate profiles for printers in each location. If you want to authorize the same group of users to work with all printers, you need to define only one profile. *Printers that use e-mail protocol do not have operator security profiles.*

**Note:** To specify the same operator security profile in a group of IP PrintWay printer definitions, create a Protocol component with the name of the security profile and specify that Protocol component in all the printer definitions. Creating a component makes it easier to change the name of the profile if necessary.

## 7.11 Infoprint Central and operator command security

### ❑ Infoprint Central issues commands to JES

#### ➤ If JESPOOL or OPERCMDS profiles exist:

#### — Authorize a group (AOOPER) for UPDATE access

```
PERMIT JES2.*.* CLASS(OPERCMDS) ACCESS(UPDATE) ID(AOOPER)
PERMIT MVS.MODIFY CLASS(OPERCMDS) ACCESS(UPDATE) ID(AOOPER)
PERMIT MVS.VARY.NET CLASS(OPERCMDS) ACCESS(UPDATE) ID(AOOPER)
PERMIT JES3.*.* CLASS(OPERCMDS) ACCESS(UPDATE) ID(AOOPER)
PERMIT JES3.*.* CLASS(OPERCMDS) ACCESS(UPDATE) ID(AOOPER)
PERMIT MVS.MODIFY CLASS(OPERCMDS) ACCESS(UPDATE) ID(AOOPER)
PERMIT MVS.VARY.NET CLASS(OPERCMDS) ACCESS(UPDATE) ID(AOOPER)
PERMIT N1.*.*.D.* CLASS(JESSPOOL) ACCESS(UPDATE) ID(AOOPER)
SETROPTS RACLIST(OPERCMDS) REFRESH ---- JES2
SETROPTS RACLIST(OPERCMDS JESSPOOL) REFRESH ----- JES3
```

Figure 7-11 Setting up security for Infoprint Central

### Authorizing the AOOPER group to RACF profiles

The user who starts Infoprint Server must be a member of the AOOPER RACF group, or have a UID of 0. (AOOPER is the default group name for Infoprint Server operators. However, your installation can assign a different name to this group.) Therefore, you can give the AOOPER group access to at least the MVS.START.STC.\*, MVS.CANCEL.jobname, and MVS.DISPLAY.\* profiles in the OPERCMDS class and profiles that control SYSOUT data set access in the JESSPOOL class. Examples are shown in Figure 7-11. If someone with a user ID of 0 who is not a member of the AOOPER group starts Infoprint Server, also give that user access to the profiles. If you start Infoprint Server in the /etc/rc file, give user ID ROOT access to the profiles.

### Authorizing JES commands

Steps for authorizing the AOOPER group to JES commands:

- Give the RACF group for Infoprint Server operators (AOOPER group) access to profiles that protect all operator commands that Infoprint Central uses.

**Examples:** If the group name for Infoprint Server operators is AOOPER, enter these RACF commands to give this group UPDATE access to broad profiles in the OPERCMDS and JESSPOOL classes:

JES2:

```
PERMIT JES2.*.* CLASS(OPERCMDS) ACCESS(UPDATE) ID(AOOPER)
PERMIT MVS.MODIFY CLASS(OPERCMDS) ACCESS(UPDATE) ID(AOOPER)
```

```
PERMIT MVS.VARY.NET CLASS(OPERCMD5) ACCESS(UPDATE) ID(AOPOPER)
```

**JES3:**

```
PERMIT JES3.*.* CLASS(OPERCMD5) ACCESS(UPDATE) ID(AOPOPER)
PERMIT JES3.*.*.* CLASS(OPERCMD5) ACCESS(UPDATE) ID(AOPOPER)
PERMIT MVS.MODIFY CLASS(OPERCMD5) ACCESS(UPDATE) ID(AOPOPER)
PERMIT MVS.VARY.NET CLASS(OPERCMD5) ACCESS(UPDATE) ID(AOPOPER)
PERMIT N1.*.*.*.D*.* CLASS(JESSPOOL) ACCESS(UPDATE) ID(AOPOPER)
```

- Refresh the OPERCMD5 and JESSPOOL classes.

```
JES2: SETROPTS RACLIST(OPERCMD5) REFRESH
```

```
JES3: SETROPTS RACLIST(OPERCMD5 JESSPOOL) REFRESH
```

If you permit the AOPOPER group to profiles while the Infoprint Central daemon (aopssid) is started, stop and restart it to pick up the changes.

## 7.12 Infoprint Central and NetSpool security

### ❑ Infoprint Central and Netspool LUs

#### ➤ AOP.NETSPOOL profile

- Restrict access for display, start, stop NetSpool LUs
- DISPLAY - READ access
- START, STOP - UPDATE access
- View log - READ access

If a user has UPDATE access to the AOP.NETSPOOL profile, the AOP.ADMINISTRATOR profile is not checked

```
RDEFINE PRINTSRV (AOP.NETSPOOL) UACC(NONE)
PERMIT AOP.NETSPOOL CLASS(PRINTSRV) ACC(CONTROL) +
    ID(userid or groupid)
SETROPTS RACLIST(PRINTSRV) REFRESH
```

Figure 7-12 Security protection for NetSpool with Infoprint Central requests

### Setting up security for NetSpool logical units (LUs)

Infoprint Central lets users work with NetSpool logical units (LUs). You can define the AOP.NETSPOOL profile to restrict who can display, start, and stop NetSpool LUs.

Previously, it was discussed that to authorize users to read the Printer Inventory, using either Infoprint Central or Infoprint Server ISPF panels, give the AOP.ADMINISTRATOR profile in the PRINTSRV class universal READ access or give all Infoprint Central users READ access.

**Note:** If a user has UPDATE access to the AOP.NETSPOOL profile, the AOP.ADMINISTRATOR profile is not checked.

### Usage consideration

Define the AOP.NETSPOOL profile. Otherwise, any Infoprint Central user can display, start, and stop NetSpool LUs.



## 7.13 Infoprint Central and job selection rules

☐ Must be authorized to protect rules or groups of rules

- Define a profile to protect a rule or rules
  - Profile applies to rules defined in the Printer Inventory

☐ Job selection rules - you can be display, start, or stop

- Display rules: READ access to AOP.ADMINISTRATOR
- Start or stop rules: Requires CONTROL access

AddJob Selection Rule

Command ==> \_\_\_\_\_

Rule name . . . Class-Y

Description . Start all jobs in class Y (extend)

Operator security profile

. . . POK.CLASSY

DEST. . . EMAIL10

CLASS . . . Y -----

FORMS . . . \_\_\_\_\_

Creator . ROGERS

WRITER. . . \_\_\_\_\_

DEST IP address . 3 1. Include 2. Exclude 3. Ignore

Figure 7-13 Protecting access to job selection rules

### Setting up security for IP PrintWay job selection rules

Infoprint Central lets users work with IP PrintWay job selection rules. IP PrintWay job selection rules control which print jobs IP PrintWay extended mode selects to process from the JES spool. You can define RACF profiles in the PRINTSRV class to restrict who can start and stop job selection rules.

You can define a separate profile to protect each job selection rule, or you can define one profile to protect a group of rules or all rules. You specify the profile that applies to each rule in the job selection rule definition in the Printer Inventory. Read access to AOP.ADMINISTRATOR profile grants job selection rule displays.

### Usage considerations

Define profiles to protect job selection rules. Otherwise, any Infoprint Central user who is authorized to read the Printer Inventory can display, start, and stop job selection rules.

## 7.14 Infoprint Central security checks

- ❑ Access to Infoprint Server daemons - display
  - AOP.DAEMON profile to protect who can display
    - No access to start or stop daemons
- ❑ Access to display printer and printer pool definitions is allowed - READ access to Printer Inventory
  - No changes allowed
- ❑ RACF auditing of access failures
  - Security administrator notified with a message
  - Specify a NOTIFY parameter on the RACF profile
- ❑ AOP.PRIORITY profile in the PRINTSRV class
  - Restricts who can change the priority of a print job

Figure 7-14 Security checking for Infoprint Central

### Setting up security for Infoprint Server daemons

Infoprint Central lets users display the status of Infoprint Server daemons to see whether they are started. You can define profile AOP.DAEMON to restrict who can display daemons. If you do not define profile AOP.DAEMON, any Infoprint Central user can display the status of daemons. Infoprint Central does not let users start and stop daemons.

### Setting up security for printer and printer pool definitions

Infoprint Central lets users display printer definitions and printer pool definitions in the Printer Inventory. Users can only display information in these definitions. Users cannot change any information.

Any Infoprint Central user who is authorized to read the Printer Inventory can display printer definitions and printer pool definitions.

### Requesting RACF notification and auditing

Infoprint Central checks a user's access to profiles before it displays any objects (such as printers or print jobs) and action buttons on its Web pages. It then displays only those objects and actions to which the user has access.

For the RACF profile that Infoprint Central checks, you must decide whether RACF should:

- ▶ Notify the security administrator (with a message) if an access check fails. If you want to receive RACF messages about access check failures in Infoprint Central, you must

specify the RACF NOTIFY parameter for the profile. If the NOTIFY parameter is not specified, Infoprint Central suppresses RACF messages for access check failures.

- Log access checks to resources that are protected by the profile. Infoprint Central logs access checks as directed by the RACF AUDIT parameter for the profile.

**Recommendation:** Do not request notification or logging for RACF profiles that Infoprint Central checks. As an alternative, request that RACF log only access check failures greater than READ. This is because access check failures can be a normal occurrence as Infoprint Central builds its Web pages and determines which objects and actions to display. RACF messages and audit entries do not necessarily mean that the user asked to display an object or selected a printer action protected by the profile. In addition, if you request notification (NOTIFY), you might receive numerous RACF messages for the same user and RACF profile because Infoprint Central sometimes checks the same profile numerous times as it builds Web pages.

## Priority of print jobs

Infoprint Central lets these job submitters do all actions on their own print jobs, regardless of whether they have access to the profiles that protect their print jobs. The AOP.PRIORITY profile is an exception. If you define the AOP.PRIORITY profile, job submitters must have UPDATE access to this profile to change the priority of their own print jobs.

Infoprint Central lets users change the priority of print jobs. To restrict who can change the priority of a print job, you can define the AOP.PRIORITY profile in the PRINTSRV class. If you do not define the AOP.PRIORITY profile, the job submitter and any other user with the required access to profiles in the OPERCMDS and JESSPOOL classes can change the priority of a print job.

```
RDEFINE PRINTSRV (AOP.PRIORITY) UACC(NONE)
PERMIT AOP.PRIORITY CLASS(PRINTSRV) ACCESS(UPDATE) ID(userid or groupid)
```

## 7.15 Logging on to Infoprint Central from the Web

❑ IP address or host name of Infoprint Server system

➤ TSO user ID and password

[http://hostname:port/Infoprint/En\\_US/IPS.html](http://hostname:port/Infoprint/En_US/IPS.html)

[https://hostname:port/Infoprint/En\\_US/IPS.html](https://hostname:port/Infoprint/En_US/IPS.html)

[http://wtsc65oe.itso.ibm.com/Infoprint/En\\_US/IPS.html](http://wtsc65oe.itso.ibm.com/Infoprint/En_US/IPS.html)



Figure 7-15 Sign-on to Infoprint Central

### Sign-on to Infoprint Central

When you log on to Infoprint Central, you can log on to only one z/OS system. You can see all the print jobs, printer queues, and Infoprint Server messages for that one z/OS system only. Therefore, if you configure IP PrintWay extended mode to print on the same printers from different z/OS systems, you must open separate instances of your Web browser so that you can see all print jobs on a printer's queue and all messages from IP PrintWay about a printer.

### Using your browser

Set your browser to always retrieve the latest pages from the Web so that the Infoprint Central refresh function works correctly:

- ▶ If you use Microsoft Internet Explorer, change the setting for temporary Internet files to check for newer versions of stored pages on every visit to a page. Do not use the Microsoft default setting, which is to check for newer versions of stored pages automatically.
- ▶ If you use Netscape Navigator, compare documents in the cache to documents on the network every time.

Do not block or disable all cookies. Infoprint Central stores cookies on your system to save your search values. If you use Microsoft Internet Explorer, do not disable the meta refresh security option. Infoprint Central uses the meta refresh function to display your search results automatically. If you use Netscape Navigator, no action is required because Netscape always enables the meta refresh function.

## Logging on to Infoprint Central from the Web

To log on to Infoprint Central, enter a URL in the browser:

- For the English version:

`http://hostname:port/Infoprint/En_US/IPS.html`

If the HTTP Server uses Secure Sockets Layer (SSL):

`https://hostname:port/Infoprint/En_US/IPS.html`

- For the Japanese version:

`http://hostname:port/Infoprint/Ja_JP/IPS.html`

If the HTTP Server uses Secure Sockets Layer (SSL):

`https://hostname:port/Infoprint/Ja_JP/IPS.html`

In each Web address:

*hostname* is the address of the z/OS system where the HTTP Server is running.

*port* is the port where the HTTP Server receives requests. If the HTTP Server receives requests at the default port, you can omit the port number. The default port number depends on whether you have customized the HTTP Server to use Secure Sockets Layer (SSL):

- The SSL default port is 443.
- Otherwise, the default port is 80.

## 7.16 Infoprint Central action access

❑	Minimum access in the PRINTSRV RACF class profiles	
❑	Using Infoprint Central, the user can	Access required
1.	Find and display printers	- READ
2.	Ping printer	- READ
3.	Redirect job to other printer IP PrintWay	- CONTROL
4.	Restore IP PrintWay printers	- CONTROL
5.	Start and Stop printers	- CONTROL
6.	View log - IP PrintWay printers	- READ
7.	View properties	- READ
8.	Space, Repeat, Pause, Interrupt - (PSF)	- CONTROL
9.	Change forms, job selection - (PSF)	- UPDATE

Figure 7-16 Security access authorization with Infoprint Central

### Security access authorization with Infoprint Central

Figure 7-16 lists some printer actions users can do in Infoprint Central and the minimum access required to the printer's profile in the PRINTSRV class.

You must define users of Infoprint Central to RACF as z/OS UNIX users. Some Infoprint Central actions require that users be authorized to read the Printer Inventory. These actions include viewing PSF printers, printer definitions, printer pool definitions, and IP PrintWay job selection rules.

### Infoprint Central functions

Then authorized users can use Infoprint Central, a Web-based application, to do these additional **printer** functions:

- ▶ Stop printing to selected printers. Restart printing to stopped printers.
- ▶ Ping printers.
- ▶ View printers' Web pages to see extended printer status, such as whether a printer is out of paper or toner.
- ▶ Redirect all print jobs on a printer's queue (except for the print job that is currently processing), as well as all future print jobs, to an alternate printer. The alternate printer must also be an IP PrintWay printer. Redirections remain in effect until the operator restores the original printers or until IP PrintWay is restarted.
- ▶ Restore redirected printers.

- ▶ View messages for selected printers.
- ▶ Delete print jobs that IP PrintWay extended mode is currently processing.
- ▶ Hold print jobs that IP PrintWay extended mode is currently processing.
- ▶ Change the priority of print jobs before or after IP PrintWay extended mode has selected them for processing.
- ▶ Move print jobs to alternate printers, including print jobs that are currently being processed. The alternate printer must also be an IP PrintWay printer.
- ▶ View messages for selected print jobs.

### Access requirements for actions

Action in Infoprint Central	Access to profile in PRINTSRV class	Minimum access
Change forms - PSF only	Profile specified in Printer Inventory	UPDATE
Change job selection - PSF only	Profile specified in Printer Inventory	UPDATE
Find and display printers (*)	Profile specified in Printer Inventory	READ
Interrupt - PSF printers only	Profile specified in Printer Inventory	CONTROL
Pause - PSF printers only	Profile specified in Printer Inventory	CONTROL
Ping and run traceroute	Profile specified in Printer Inventory	READ
Redirect - IP PrintWay printers (*)	Profile specified in Printer Inventory	CONTROL
Repeat - PSF printers only	Profile specified in Printer Inventory	CONTROL
Reset (*)	Profile specified in Printer Inventory	CONTROL
Restore - IP PrintWay printers only	Profile specified in Printer Inventory	CONTROL
Space - PSF printers only	Profile specified in Printer Inventory	CONTROL
Start	Profile specified in Printer Inventory	CONTROL
Stop printer and delete or hold the current print job	Profile specified in Printer Inventory	CONTROL
Stop printer after the current print job completes	Profile specified in Printer Inventory	CONTROL
Turn offline (*)	Profile specified in Printer Inventory	CONTROL
Turn online (*)	Profile specified in Printer Inventory	CONTROL
View log	Profile specified in Printer Inventory	READ
View properties	Profile specified in Printer Inventory	READ

#### (\*) Notes:

1. To redirect an IP PrintWay printer to an alternate printer, CONTROL access is required to the profiles for both printers.
2. To display information from the printer and to display some printer actions (turn online, turn offline, and reset), the SNMP read community name for the printer must be "public," or it must be specified in the AOP\_READ\_COMMUNITY environment variable.
3. To perform some printer actions (turn online, turn offline, and reset), the Infoprint Central user must enter the SNMP write community name for the printer unless the write community name is "public" or it is specified in the AOP\_WRITE\_COMMUNITY environment variable.
4. The RACF profile for the printer in the PRINTSRV class restricts access to the reset, turn offline, and turn online actions.

## 7.17 PIDU command to create printer security profiles

- ❑ Use **pidu** command to specify the name of the same RACF profile in all IP PrintWay printer definitions that do not already contain a profile name
  - Names piped to "awk" program
  - awk program writes modify commands to modify the printer definitions to file /tmp/defs
  - Inspect the /tmp/defs file to make sure the modify commands are acceptable
  - Enter this command to update the Printer Inventory:
    - **pidu < /tmp/defs**

```
pidu -qc "list printer where printer-type=ip-printway and  
operator-security-profile=null;" | awk '{print "modify printer  
" operator-security-profile = \"POK>ROOM1\";"}' > /tmp/defs
```

Figure 7-17 PIDU command and security profiles

### PIDU command and security profiles

The PIDU commands in Figure 7-17 show how to specify the name of the same RACF profile in all IP PrintWay printer definitions that do not already contain a profile name.

- ▶ Enter the OMVS shell. Enter these commands as one command on the z/OS UNIX command line:

```
pidu -qc "list printer where printer-type=ip-printway and  
operator-security-profile=null;" | awk '{print "modify printer " $1  
" operator-security-profile = \"POK>ROOM1\";\"}' > /tmp/defs
```

The PIDU list command lists the names of all IP PrintWay printer definitions with no value in the operator-security-profile attribute. These names are piped to the awk program, which writes modify commands to modify the printer definitions to file /tmp/defs.

- ▶ Inspect the /tmp/defs file to make sure the modify commands are acceptable.
- ▶ Enter this command to update the Printer Inventory:

```
pidu < /tmp/defs
```

### awk command

**awk** is a powerful command that can perform many different operations on files. The general purpose of **awk** is to read the contents of one or more files, obtain selected pieces of information from the files, and present the information in a specified format.



## 7.18 Work with Printers panel

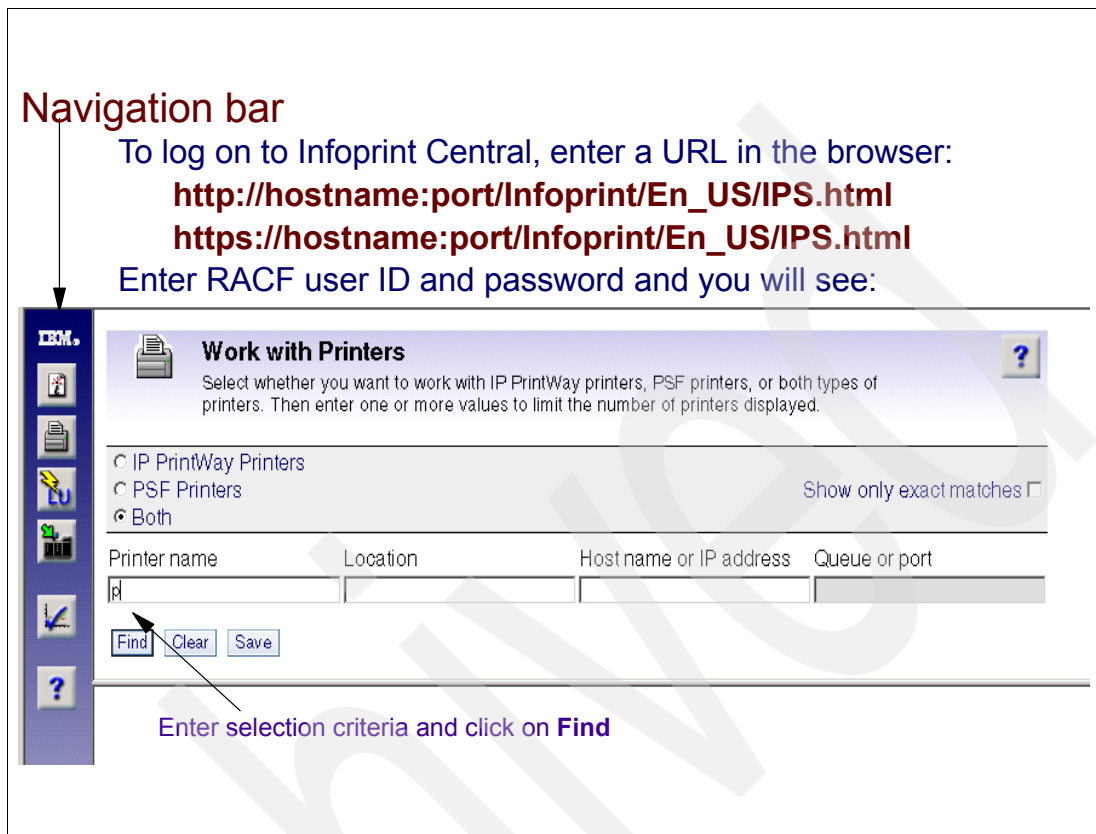


Figure 7-18 Work with printers panel - primary panel

### Working with printers

To log on to Infoprint Central, enter a URL in the Web browser:

`http://hostname:port/Infoprint/En_US/IPS.html`

If the HTTP Server uses Secure Sockets Layer (SSL):

`https://hostname:port/Infoprint/En_US/IPS.html`

*hostname* The address of the z/OS system where the HTTP Server is running.

*port* The port where the HTTP Server receives requests. If the HTTP Server receives requests at the default port, you can omit the port number.

The default port number depends on whether you have customized the HTTP Server to use Secure Sockets Layer (SSL):

- ▶ The SSL default port is 443.
- ▶ Otherwise, the default port is 80.

### Work with Printers panel

This is the first panel displayed after the log on. You can work with these types of printers:

**PSF printers** A Print Services Facility™ (PSF) printer is an IBM Advanced Function Presentation (AFP) printer that PSF for z/OS controls. You can work with any PSF printer that your administrator has defined to both

Infoprint Server and z/OS. Each PSF printer must be defined in an FSA definition in the Printer Inventory and in a JES initialization statement.

**IP PrintWay printers** An IP PrintWay printer is typically a (1) PCL or PostScript printer in your TCP/IP network or (2) VTAM-controlled printer in your SNA network. You can work with any IP PrintWay printer that your administrator has defined to Infoprint Server. Each IP PrintWay printer should be defined in a printer definition in the Printer Inventory.

### **Buttons on left side of Work with Printers panel**

The buttons on the left side of the panel from top to bottom are as follows:

- ▶ Work with print jobs
- ▶ Work with printers
- ▶ Work with NetSpool logical units (LUs)
- ▶ Find printer definitions
- ▶ Check system status
- ▶ Help

### **Selecting options on this panel**

Select whether you want to work with IP PrintWay printers, PSF printers, or both types of printers. Then enter one or more values to limit the number of printers displayed, as follows:

- ▶ You can select by printer name, location, hostname or IP address, or queue or port by typing the information in the field.
- ▶ If you want to see all printers that start with the letter p, just Enter p
- ▶ Select the Find button to display the next panel

## 7.19 Working with printers

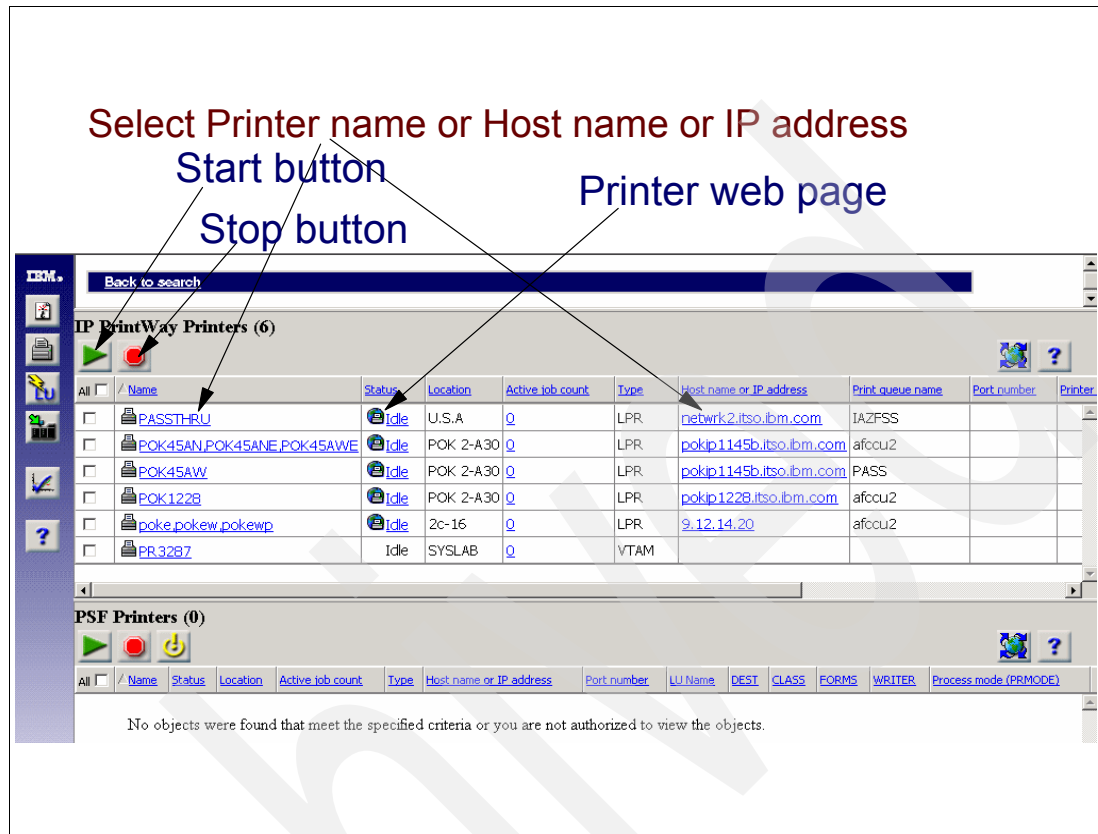


Figure 7-19 IP PrintWay printers display from working with Find printers button

### Working with printers

Work with Printers panel lets you select the printers you want to work with. This panel lists IP PrintWay printers that you are authorized to see. This panel lets you work with several printers at the same time and link to more detailed information about one printer. To work with printers, you can do any of the following:

- ▶ Select the boxes next to the printers that you want to work with. To work with all printers in the list, select All.
- ▶ Select an action button.
- ▶ To view any error messages, Select (Error).
- ▶ To view any information messages, Select (Information).

**To see detailed printer information about a printer and do additional actions:** Select the value in the Name or Printer host name or IP address field for the printer. On the IP PrintWay Printer Information panel, you can see more information about the printer and do these additional actions: Ping, Redirect, and View log.

**To see the print queue for a printer:** Select the value in the Active job count field for the printer.

**To see extended status information for a printer:** Select the Web page ICON in the Status field for the printer. This opens a new browser window with the URL set to the IP address of the printer. Not all printers have a Web page. Also, if the printer is controlled by a print server, Inprint Central cannot display the printer's Web page.

## Using the buttons

There are two buttons, the start and stop for the printer, as follows:

**Start** Starts the selected printers. The printers begin processing print jobs.

Results: You see an information or error message. If the message is not automatically displayed, select or to view the messages.

If the action was successful, the status of the printers changes to either Idle or Processing.

**Stop** Stops the selected printers so that IP PrintWay does not process any more print jobs for the printer. However, data that IP PrintWay has already transmitted to a printer and is already in the printer's buffer continues to print. Infoprint Server and JES continue to accept print requests for stopped printers and add print jobs to the printers' queues.

On the Stop Printer panel, you can select whether to stop the printer immediately or wait until the current print job completes printing. If you stop the printer immediately, the current print job is deleted.

### Attention:

To stop a printer immediately but without deleting the current print job:

- ▶ On the Stop Printer panel, select Complete the current print job and then select OK.
- ▶ On the IP PrintWay Printer Information panel, expand the Print Job Queue section.
- ▶ On the Infoprint Server Print Jobs panel, hold the print job that is processing.

If a printer is not working, you can redirect all print jobs to an alternate IP PrintWay printer after you stop it. To redirect a printer, Select (Redirect).

To restart a stopped printer, Select (Start).

If IP PrintWay is restarted, all stopped printers are automatically restarted.

## 7.20 IP PrintWay Information window

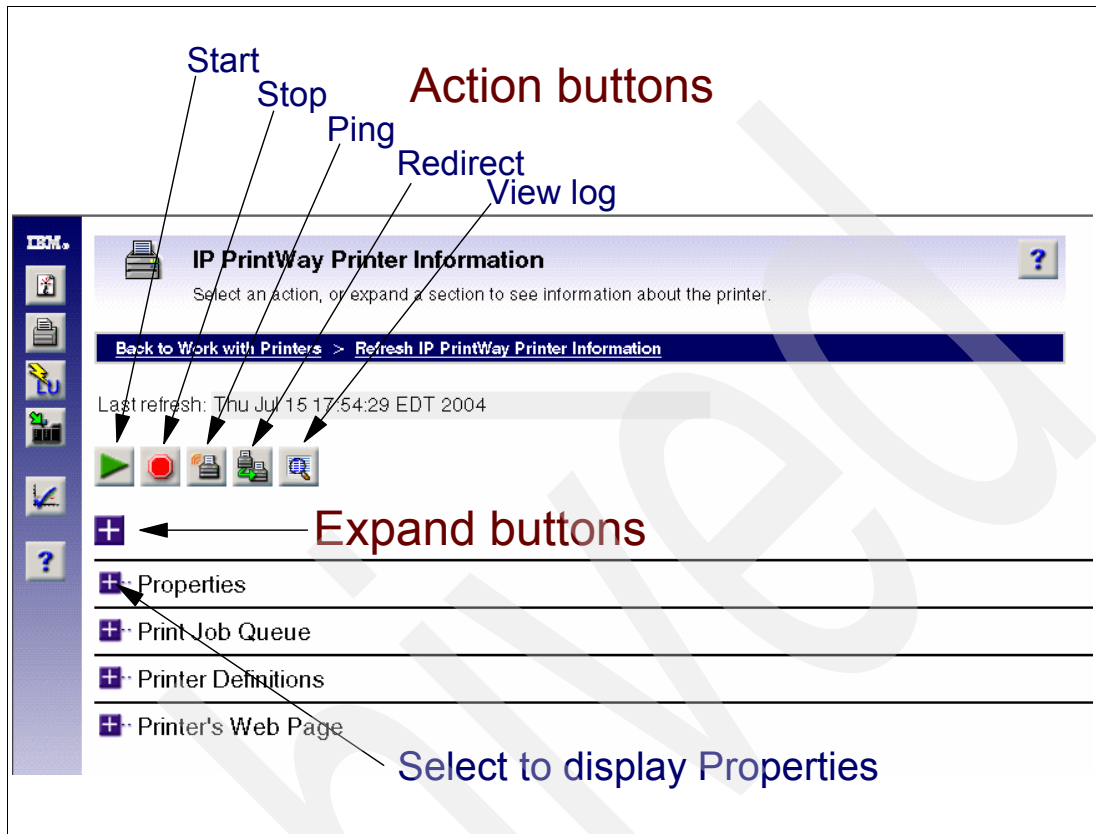


Figure 7-20 IP PrintWay printer information window

### IP Printway Printer Information panel

The panel shown in Figure 7-20 is a result of selecting a printer shown in Figure 7-19 on page 357. This IP PrintWay printer panel lets you work with the selected printer and see detailed information about it.

### Working with a printer action button

You can select one of the five action buttons, shown in Figure 7-20. When the action completes, you see an error message or an information message.

#### Redirection information

The printer where this printer is redirected, or other printers that are redirected to this printer. You see this section only when redirections are in effect for this printer. When IP PrintWay is restarted, all redirections that are in effect are restored.

To redirect the printer to a different printer, select the value in the Name field. On the IP PrintWay Printer Information panel, Select (Redirect).

To remove the redirection that is in effect, select the value in the Name field. On the IP PrintWay Printer Information panel, Select (Restore).

When IP PrintWay is restarted, all redirections that are in effect are restored.

## Detailed information about the printer

To see detailed information about the printer, to expand all sections, select (Expand all) at the top of the panel or Select an Expand button to expand one or more of the following sections:

<b>Properties</b>	The basic properties of the printer.
<b>Print Job Queue</b>	<p>Print jobs that were submitted to this printer. This queue contains active jobs, including print jobs that are being processed, are waiting to be processed, or are held. It also includes retained print jobs, which are print jobs that already completed but are retained for a period of time. Retained jobs are automatically deleted when the retention period expires. The print jobs will print in the order displayed.</p> <p>If someone redirected this printer to another printer, you do not see active print jobs submitted to this printer because these print jobs are on the queue of the alternate printer. However, you can see print jobs that Infoprint Server retained before you redirected the printer. If you Select (Release) to print retained print jobs, the print jobs print on the alternate printer. After you restore the printer, print jobs that were retained on the alternate printer remain on that printer's queue.</p>
<b>Printer Definitions</b>	The printer definitions in the Printer Inventory for this printer. You see this section only when at least one printer definition exists for the printer.
<b>Printer's Web Page</b>	A link to the printer's Web page, which contains extended status information. Infoprint Central opens a new browser window with the URL set to the IP address of the printer. Not all printers have a Web page. Also, if the printer is controlled by a print server, Infoprint Central cannot display the printer's Web page. You see this section only when IP PrintWay uses either the LPR or direct sockets transmission protocol to send print jobs to the printer.

## 7.21 Ping a printer

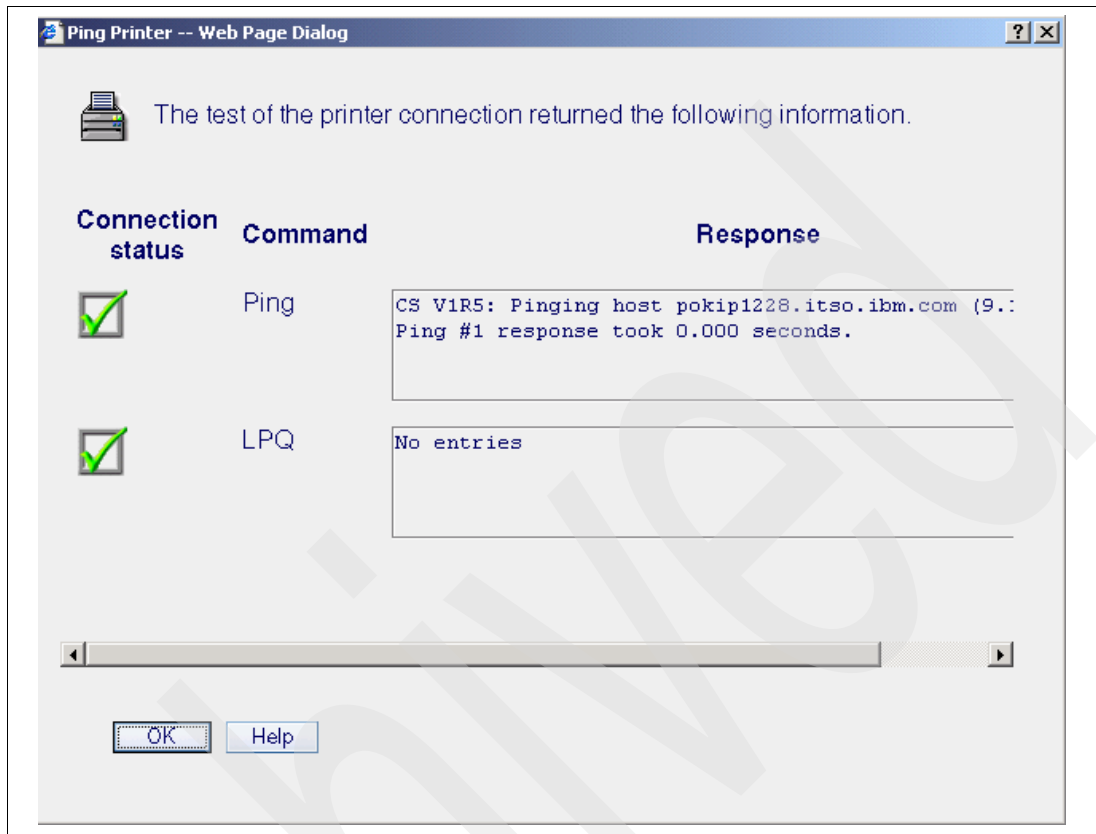


Figure 7-21 Ping a printer window

### Pinging a printer

When you select the Ping action button shown in Figure 7-20 on page 359, the window shown in Figure 7-21 is displayed. The Response field displays the responses received from these commands:

**Ping** Tests the TCP/IP network connection to a remote printer or print server.

**Success** means the TCP/IP network is working and the z/OS system can communicate with the printer. If you ping a printer instead of a print server, a successful response also means the printer is turned on. However, the printer might be offline.

**Failure** means the printer did not respond in 1 second. This can occur if the network is not working, the printer is not turned on, or the remote host is slow to respond.

**LPQ** Provides information about jobs on the LPD's print queue. The information provided depends on the printer's implementation of the LPD. You can see a response from this command only for IP PrintWay printers that use the LPR to LPD protocol to communicate with the printer.

**Success** means the LPD in the printer or print server is running. However, the printer might be offline.

**Failure** means the TCP/IP network connection to the printer or print server is not working, the printer does not contain an LPD, or the LPD is not running.

## Ping considerations

If the **ping** command is unsuccessful, consider the following:

- ▶ Select the **Ping** action button again to see if the problem persists.
- ▶ Make sure the printer is turned on.
- ▶ Make sure the value in the Printer host name or IP address field on the Printer Information panel is correct. If the value in this field is not correct, ask your administrator to change the value in the Printer Inventory.

If the **ping** command is successful but print jobs are not printing, make sure the printer is online.

If the **ping** command is successful but print jobs are not printing on the expected printer, the printer's IP address might be a duplicate of another IP address in the system. To see if this is the case:

- ▶ Turn off the printer. For a PSF printer, also turn off the IPDS LAN attachment (such as the i-data 7913 IPDS Printer LAN Attachment) if one exists.
- ▶ Wait at least 5 minutes for the system to clear the TCP/IP Address Resolution Protocol (ARP) tables.
- ▶ Select the Ping action button again.
- ▶ If the **ping** command is still successful, a duplicate IP address exists.

## LPQ considerations

If the **LPQ** command is successful and the response from the **LPQ** command indicates that the print queue does not exist, make sure the value in the Queue field on the Printer Information panel is correct. The print queue name is case-sensitive so be sure the correct uppercase and lowercase letters are used. If the value in this field is not correct, ask your administrator to change the value in the Printer Inventory.



## 7.22 Redirect a printer

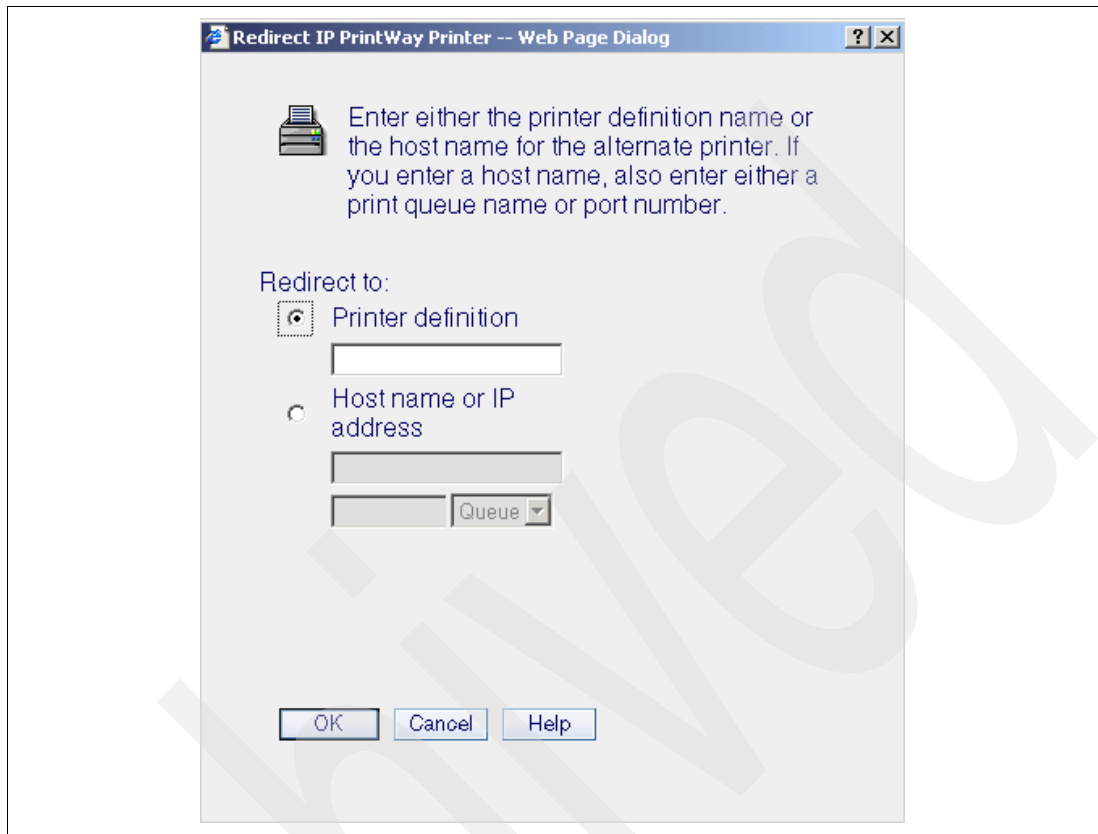


Figure 7-22 Redirect IP PrintWay Printer panel

### Redirect a printer

To see the panel shown in Figure 7-22, Select the Redirect action button shown in Figure 7-20 on page 359. The Redirect IP PrintWay Printer panel lets you move all print jobs currently on the queue of the selected IP PrintWay printer and all print jobs that are subsequently submitted to this printer to an alternate IP PrintWay printer. When you select the alternate printer, consider the following:

- ▶ The alternate printer must be an IP PrintWay printer.
- ▶ If a printer definition does not exist for the alternate printer in the Printer Inventory, the printer must support either the LPR or direct-sockets printing protocol.
- ▶ The alternate printer must not itself be redirected to an alternate printer.
- ▶ The alternate printer should be able to print the same types of data streams (for example, PostScript or PCL) as the original printer so that data prints correctly.

### Printer selected for redirect

To specify the printer where you want to redirect all print jobs, select how you want to specify the alternate printer, as follows:

**Printer definition** Enter the name of the printer definition for the alternate printer. IP PrintWay redirects print jobs to the printer address specified in this printer definition. However, IP PrintWay continues to use attributes in the printer definition for the original printer to format data.

**Host name or IP address** Enter either the host name or dotted-decimal IP address of the alternate printer.

### **Redirect considerations**

To list printer definitions, select **Find printer definitions** from the navigation bar.

Select **Host name or IP address** if no printer definition exists for the printer.

- ▶ If you select **Host name or IP address**, enter either the print queue name or the port number of the alternate printer. Then select either Queue or Port to identify the type of value you entered.
- ▶ Click **OK**.

## 7.23 Using the stop button

- ❑ Stop IP PrintWay from sending more data to printer
  - The printer is stopped until you restart it or IP PrintWay is restarted
  - Stops the selected printers so that IP PrintWay does not process any more print jobs for the printer
  - Data already transmitted to a printer and is already in the printer's buffer continues to print
  - Infoprint Server and JES continue to accept print requests for stopped printers and add print jobs to the printers' queues
- ❑ On the Stop Printer panel, you can select whether to stop the printer immediately or wait until the current print job completes printing. If you stop the printer immediately, the current print job is deleted

Figure 7-23 Using the stop button with Infoprint Central

### Stop button processing

You can start and stop IP PrintWay and PSF printers. When you stop a printer, you can select whether the print job that is currently processing is to be completed or deleted. You might want to stop a printer that is not working. You might need to restart PSF printers after the z/OS system is IPLed. Printers might be started and stopped automatically in the following situations:

- ▶ When the operator restarts IP PrintWay, stopped printers are automatically restarted.
- ▶ When the operator cancels the PSF functional subsystem (FSS), printers that the FSS controls are automatically stopped.
- ▶ When the z/OS system is IPLed, any IP PrintWay printers that you had stopped are restarted. If IP PrintWay printers are not working after the IPL, make sure that the IP PrintWay daemons are started on the System Status panel.
- ▶ When the z/OS system is reIPLed, all PSF printers are stopped unless your administrator defined them to JES with the automatic start option.

To stop printers:

- ▶ Select (Work with printers) from the navigation bar.
- ▶ On the Work with Printers panel, enter search values to find the printer.
- ▶ On the IP PrintWay Printers or PSF Printers panel, select the box next to the printer and select Stop to stop the printer. You can stop more than one printer at the same time.

On the Stop Printer panel, select whether to complete the print job that is currently processing or delete it.

## 7.24 Stop a printer options

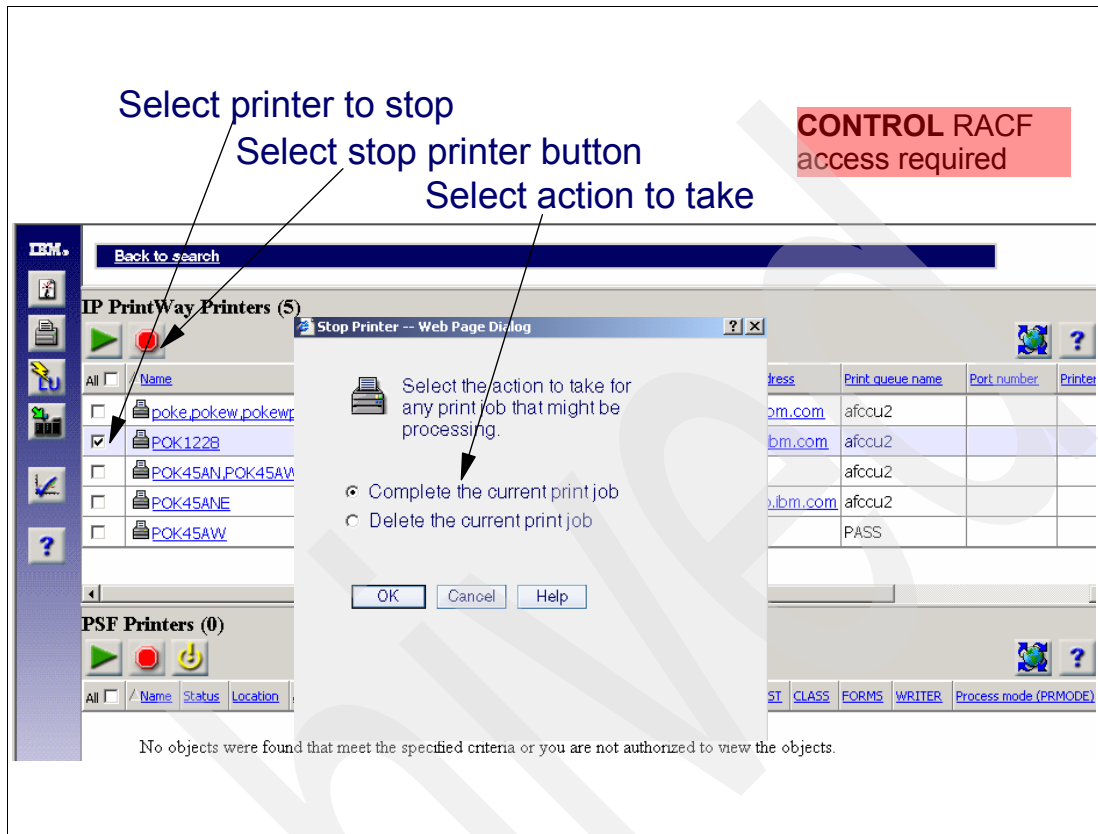


Figure 7-24 Stopping a printer options

### Options to stop printers

When you select the Stop action button shown in Figure 7-22, the Stop Printer - Web Page Dialog panel appears on the screen and lets you stop a printer. Select the action to perform for any print jobs that are being processed on that printer.

To select the action to perform for the current print job, select one of these options:

- ▶ **Complete the current print job:** The current print job finishes.
- ▶ **Delete the current print job:** The current print job is deleted from the JES spool.
- ▶ Then, click **OK**.

## 7.25 View log for a printer

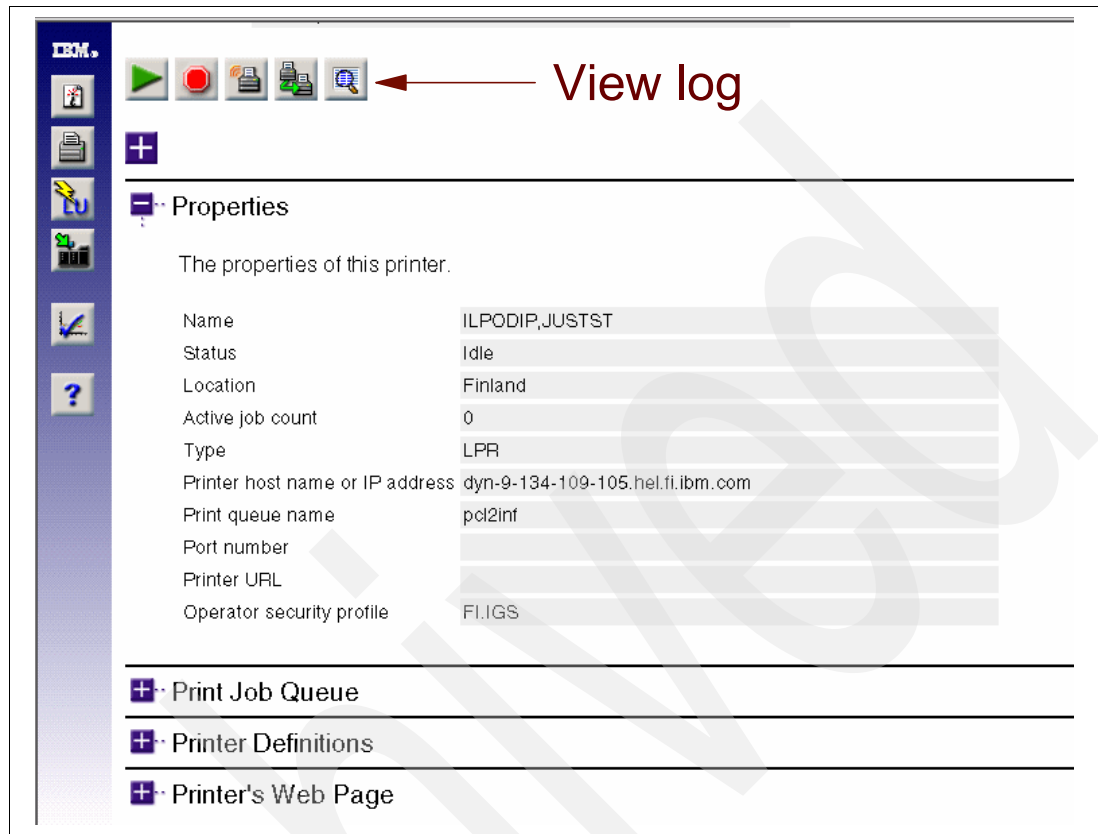


Figure 7-25 View the log for a printer

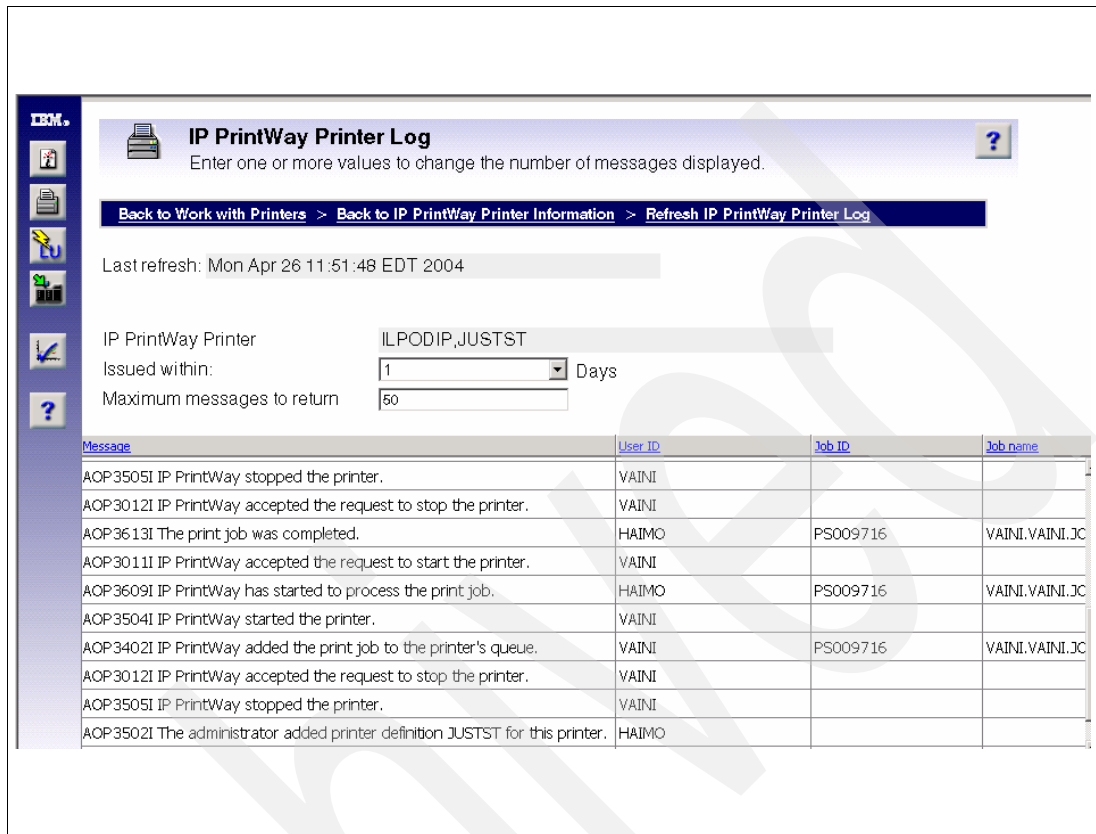
### View log

Infoprint Central lets authorized users view messages in the common message log for selected print jobs and IP PrintWay extended mode printers. In addition, Infoprint Server administrators can use the **aoplogu** command to select messages in a particular time range and copy them to a file or view them on the terminal. Enter this command from the z/OS UNIX command line.

The view log displays all messages from the Infoprint Server common message log for the printer. Messages from Infoprint Server Transforms and other transforms are not displayed. Ask your administrator to find those messages.

Result: The IP PrintWay Printer Log panel displays the messages.

## 7.26 Viewing printer log



**IP PrintWay Printer Log**  
Enter one or more values to change the number of messages displayed.

[Back to Work with Printers](#) > [Back to IP PrintWay Printer Information](#) > [Refresh IP PrintWay Printer Log](#)

Last refresh: Mon Apr 26 11:51:48 EDT 2004

IP PrintWay Printer: ILPODIP,JUSTST

Issued within: 1 Days

Maximum messages to return: 50

Message	User ID	Job ID	Job name
AOP3505I IP PrintWay stopped the printer.	VAINI		
AOP3012I IP PrintWay accepted the request to stop the printer.	VAINI		
AOP3613I The print job was completed.	HAIMO	PS009716	VAINI.VAINI.JC
AOP3011I IP PrintWay accepted the request to start the printer.	VAINI		
AOP3609I IP PrintWay has started to process the print job.	HAIMO	PS009716	VAINI.VAINI.JC
AOP3504I IP PrintWay started the printer.	VAINI		
AOP3402I IP PrintWay added the print job to the printer's queue.	VAINI	PS009716	VAINI.VAINI.JC
AOP3012I IP PrintWay accepted the request to stop the printer.	VAINI		
AOP3505I IP PrintWay stopped the printer.	VAINI		
AOP3502I The administrator added printer definition JUSTST for this printer.	HAIMO		

Figure 7-26 Example of a printer log

### Viewing a printer's log

IP PrintWay Printer Log panel lists messages from Infoprint Server for the selected printer. Fields before and after the message text contain additional information, such as the time the message was sent.

For more information about each message, use the z/OS LookAt online facility from the Internet at:

<http://www.ibm.com/eserver/zseries/zos/bkserv/lookat/>

or from a TSO/E command line.

Messages from Print Services Facility (PSF), Infoprint Server Transforms, and other transforms for print jobs are not displayed. Ask your administrator for those messages.

### Controlling messages displayed

In the panel shown in Figure 7-26, you can change the number of messages displayed, as follows:

- ▶ In the **Issued within** field, select the number of days of messages you want to see.
- ▶ In the **Maximum messages to return** field, specify the number (1 - 999) of messages you want to see.
- ▶ Click **Refresh**.

## 7.27 Infoprint Server messages in the OPERLOG

```
AOP048E 9.876.543.210: Unknown host. (program:aopwsmd) (job:JOB43496) 466
filename:SYS1.PROCLIB(JES3)) (dsn:VAINI.VAINI7.JOB43496.D0000010.
IB#JES3#) (job_selection_rule:SAPI)
AOP3417I IP PrintWay rejected the print job. The print job remains on 467
the JES spool. (program:aopwsmd) (job:PS000829) (filename:SYS1.
PROCLIB(JES3)) (dsn:VAINI.VAINI7.JOB43496.D0000010.IB#JES3#)
(job_selection_rule:SAPI)
AOP118I Job 831 document SC43RM02 started spooling to IAZFSS. 469
(program:aopnetd) (job:PS000831) (filename:SC43RM02) (dsn:ITSO.ITSO.
JOB43490.D000000D.SC43RM02)
AOP119I Job 831 document SC43RM02 completed spooling with status 470
"pending". (program:aopnetd) (job:PS000831) (filename:SC43RM02)
(dsn:ITSO.ITSO.JOB43490.D000000D.SC43RM02)
AOP034E shift-out-shift-in: The value 0 is not valid. (program:aopoutd 479
) (job:JOB43496) (filename:SYS1.PROCLIB(JES3)) (dsn:VAINI.VAINI8.JOB43
496.D0000011.IB#JES3#) (output_device:lpr://9.134.109.105/ebopcl).
AOP120I Job 836 document SC43RM02 started spooling to TOEVAIN with 495
transform afp2pdf.dll -c eu . (program:aopnetd) (job:PS000836)
(filename:SC43RM02) (dsn:AOPSS.NETSPOOL.JOB43490.D000000E.SC43RM02)
AOP119I Job 836 document SC43RM02 completed spooling with status 496
"pending". (program:aopnetd) (job:PS000836) (filename:SC43RM02)
(dsn:AOPSS.NETSPOOL.JOB43490.D000000E.SC43RM02)
ANFM601I Data set: 510
WTSCPLX9.VAINI.VAINI1.JOB43496.D0000013.B#BPXAS# The data set was
successfully transmitted to host and queue: 9.12.4.148 PASS
ANFM604I Data set: WTSCPLX9.VAINI.VAINI1.JOB43496.D0000013.B#BPXAS# The data
set is being released to JES
```

Figure 7-27 Viewing the printer log messages in the OPERLOG

### Infoprint Server message configuration file (aopmsg.conf)

The Infoprint Server message configuration file, `aopmsg.conf`, lets you customize message processing. The default location of this configuration file is `/etc/Printsrv/aopmsg.conf`.

This configuration file is optional. If the file does not exist or if an attribute in the file is omitted, Infoprint Server uses default values.

In the configuration file, you can enable the Infoprint Server hardcopy log function. The hardcopy log function sends additional Infoprint Server messages to the z/OS system hardcopy log so that you can use a z/OS message automation facility to process these messages.

If you do not enable the hardcopy log function, only messages that Infoprint Server sends to the console are sent to the hardcopy log. (Infoprint Server sends messages to the console that require the attention of an operator or administrator.) However, if you want non-console messages that are written only to the Infoprint Server common message log to be sent to the hardcopy log, you must enable the hardcopy log function.

The hardcopy log is either the system log (SYSLOG) or the operations log (OPERLOG), depending on how you configured the hardcopy medium in the `HARDCOPY` statement in the `CONSOLxx` Parmlib member or changed the hardcopy medium using the `HARDCPY` format of the `VARY` operator command.

The hardcopy log function can log messages from these Infoprint Server programs:

- ▶ aopd - Printer Inventory Manager daemon
- ▶ aophinvd - Historical Inventory daemon
- ▶ aopipddxp - IPP Server daemon
- ▶ aoplogd - Common message log daemon
- ▶ aoplpd - Line printer daemon (LPD)
- ▶ aopnetd - NetSpool daemon
- ▶ aopoutd - IP PrintWay extended mode daemon that sends output to remote printers in the TCP/IP network and to e-mail destinations
- ▶ aopsapd - SAP callback daemon
- ▶ aopsnmpd - SNMP subagent daemon
- ▶ aopssid - Infoprint Central daemon that communicates with JES
- ▶ aopsubd - Print Interface subsystem daemon
- ▶ aopwsmd - IP PrintWay extended mode daemon that selects output data sets from the JES spool
- ▶ aopxtd - Infoprint Server Transform Manager daemon

The hardcopy log function cannot log messages from other programs, such as messages from IP PrintWay basic mode.

The name of the program that wrote a message is in the *program* field of the message. If you view the message in the Infoprint Central message log, the program name is in the *Process* column.

Messages sent to the hardcopy log have this format:

```
messageID text (program:name) [(job:id)] [(filename:name)] [(dsn:name)]  
[(output_device:name)] [(job_selection_rule:name)]
```



## 7.28 Attributes for the message configuration file

```
# aopmsg.conf - Infoprint Server Message Configuration file
# This file is optional. If it is specified, the default location is:
# /etc/Printsrv/aopmsg.conf
# In this file, you can request that Infoprint Server send additional
# messages to the z/OS system hardcopy log. If this file does not exist,
# only messages that Infoprint Server writes to the console are sent to
# the hardcopy log.
# You can specify these attributes in the file:
# hardcopy-messages = all | list | none
# hardcopy-message-list = {message1 message2 ...}
#-----
# Uncomment the line below to send all eligible messages to the hardcopy log.
# Eligible messages include messages from Infoprint Server daemons, but not
# messages from IP PrintWay basic mode.
#hardcopy-messages = all
#-----
# Uncomment the two lines below to send messages AOP3614I and AOP3803E to
# the hardcopy log.
#hardcopy-messages = list
#hardcopy-message-list = {AOP3614I AOP3803E}
#-----
# Uncomment the line below to send no additional messages to the hardcopy log.
# This is the default action if this file does not exist or the
# hardcopy-messages attribute is not specified.
#hardcopy-messages = none
#-----
```

Figure 7-28 Attributes for the message configuration file

### Attributes for the message configuration file (aopmsg.conf)

The sample message configuration file is /usr/lpp/Printsrv/samples/aopmsg.conf.

You can specify these attributes in the aopmsg.conf configuration file:

hardcopy-messages = all | list | none

- all - Sends all eligible messages to the hardcopy log.
- list - Sends only the additional messages that are listed in the hardcopy-message-list attribute to the hardcopy log.
- none - Sends no additional messages to the hardcopy log. Only console messages are sent to the hardcopy log. This is the default.

If you change this attribute while any Infoprint Server daemons are running, restart all daemons.

Default: hardcopy-messages=none

hardcopy-message-list = {messageID messageID ...}

The message IDs of the messages that Infoprint Server sends to the hardcopy log when the hardcopy-messages=list attribute is specified. Specify the *entire* message ID including the severity code (E, I, S, T, or W). You can use either uppercase or lowercase characters. Enclose the list of messages in braces, and separate message IDs with spaces.

Default: None.

## 7.29 Infoprint Central work with Print Jobs

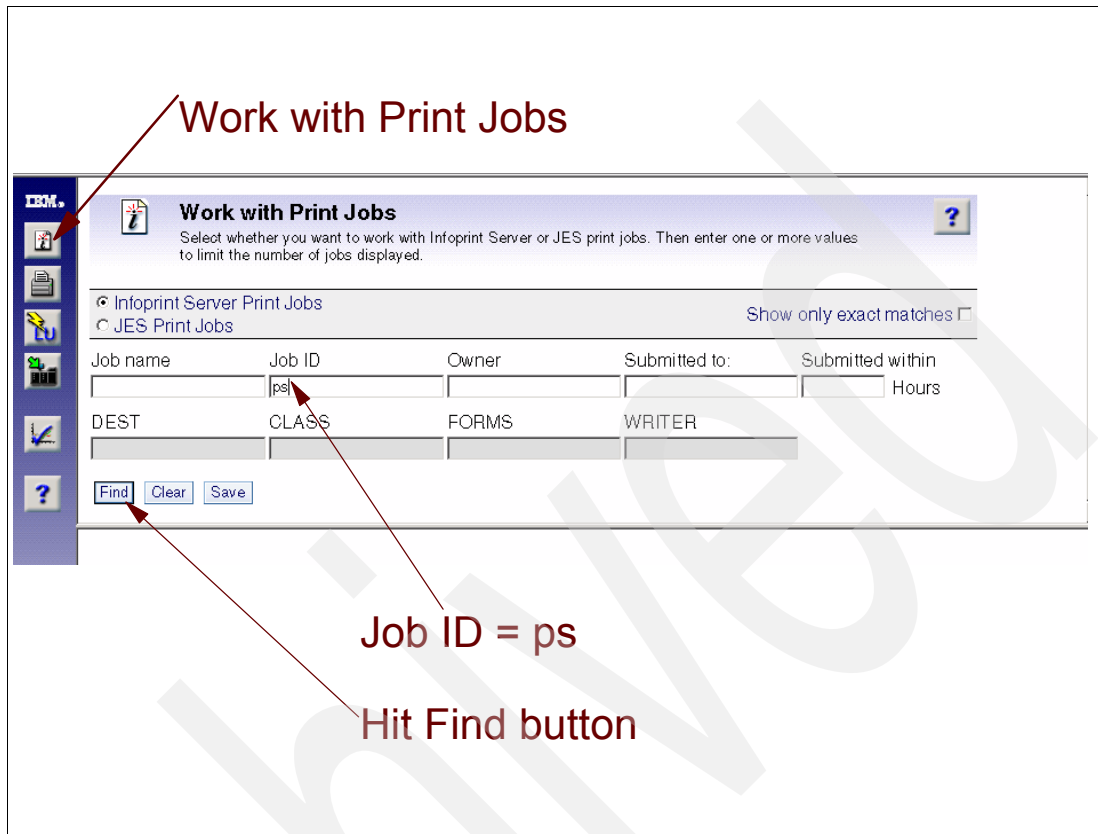


Figure 7-29 Infoprint Central Work with Print Jobs

### Work with Print Jobs

The Work with Print Jobs panel lets you select the print jobs you want to work with. You can work with any print jobs that are currently on the JES spool. You can also see information about any Infoprint Server print jobs that have already finished processing.

### Finding print jobs

Select the type of print jobs to display. Select either **Infoprint Server print jobs** or **JES print jobs**. The type you choose determines which print jobs are displayed, how much information you see about the print jobs, and which actions you can do on print jobs. The type you choose also changes which search fields are available on this panel.

### Infoprint Server print jobs

First select **Infoprint Server print jobs**. This option lets you:

- ▶ See more information about Infoprint Server print jobs, such as more detailed status.
- ▶ Do more actions on Infoprint Server print jobs, such as move a print job to another printer.
- ▶ See information, including messages, for Infoprint Server print jobs that have finished processing and are no longer on the JES spool.
- ▶ Search for print jobs using search values known only to Infoprint Server, such as the Windows logon name of the job submitter.

In Figure 7-29, the selection is for Job ID, and since all print jobs have a Job ID starting with PS, this selects all of those jobs.

## JES print jobs

If the print job is not found, select **JES print jobs**. This option lets you:

- ▶ Work with print jobs that Infoprint Server does not process, such as print jobs submitted directly to a PSF-controlled printer using JCL.
- ▶ Work with print jobs that Infoprint Server has not received due to an error in job submission. For example, job submitters might fail to specify the correct CLASS value on their DD or OUTPUT JCL statements, so IP PrintWay does not select the print jobs from the JES spool.

## Find print job option 2

A separate option could be to Enter a value in the **Submitted within** field to limit the number of jobs found. Do not enter a value if you want to find all possible print jobs. Use this field as follows:

- ▶ Enter a value in at least one of the search fields. If you enter values in more than one field, only print jobs that match all values are displayed.
  - Do not enter an asterisk (\*) or a question mark (?) as a wildcard symbol.
  - If you enter only the first characters of a value in a search field instead of the full value, clear Show only exact matches.
  - If you do not know the correct uppercase and lowercase letters for case-sensitive fields, Clear shows only exact matches.

## Save button

Select **Save** to save these search values. The saved search values are displayed the next time you use this panel when you Select the **Find** button.

## 7.30 Infoprint Server print jobs

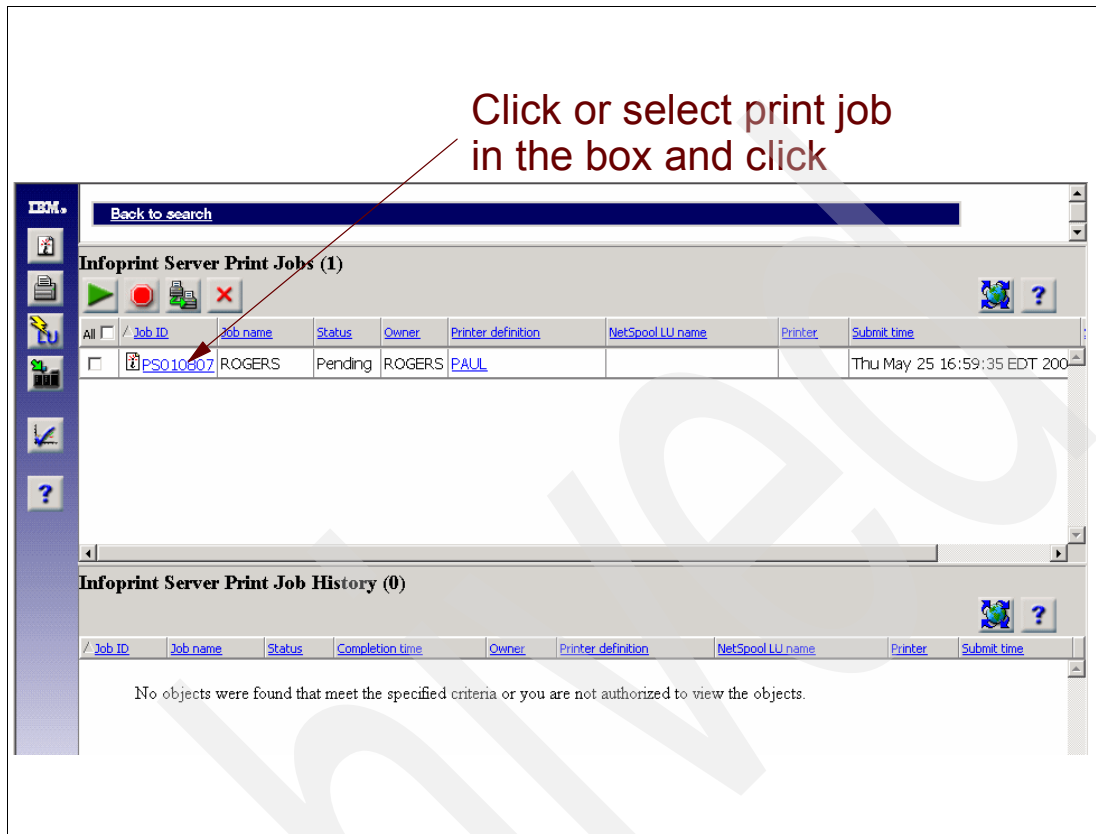


Figure 7-30 Infoprint Server job selected

### Infoprint Server print job

The panel shown in Figure 7-30 lists Infoprint Server print jobs that you are authorized to see. This panel lets you work with several print jobs at the same time. You can also link to more detailed information about one print job and to related objects.

### Working with print jobs

Select the boxes next to the print jobs that you want to work with. To work with all print jobs in the list, select **All**. You can select an action button to do one of the following:

- ▶ To view any error messages, click the Error button.
- ▶ To view any information messages, click the Information button.

### Detailed information about a print job

Select the value in the Job ID field for the print job. On the Infoprint Server Print Job Information panel that is then displayed and shown in Figure 7-31 on page 376, you can see more information about the print job and do these additional actions:

- ▶ Change priority
- ▶ Move
- ▶ View properties
- ▶ View log

### **Additional actions**

To see information about the printer where the print job is to print or has printed, and to work with that printer, select (IP PrintWay printer) in the Printer field for the print job. This field is blank if the printer is not an IP PrintWay printer or if IP PrintWay has not yet received the print job.

To see the printer definition used to format a print job, select the value in the Printer definition field for the print job.

Archived

## 7.31 Print job information

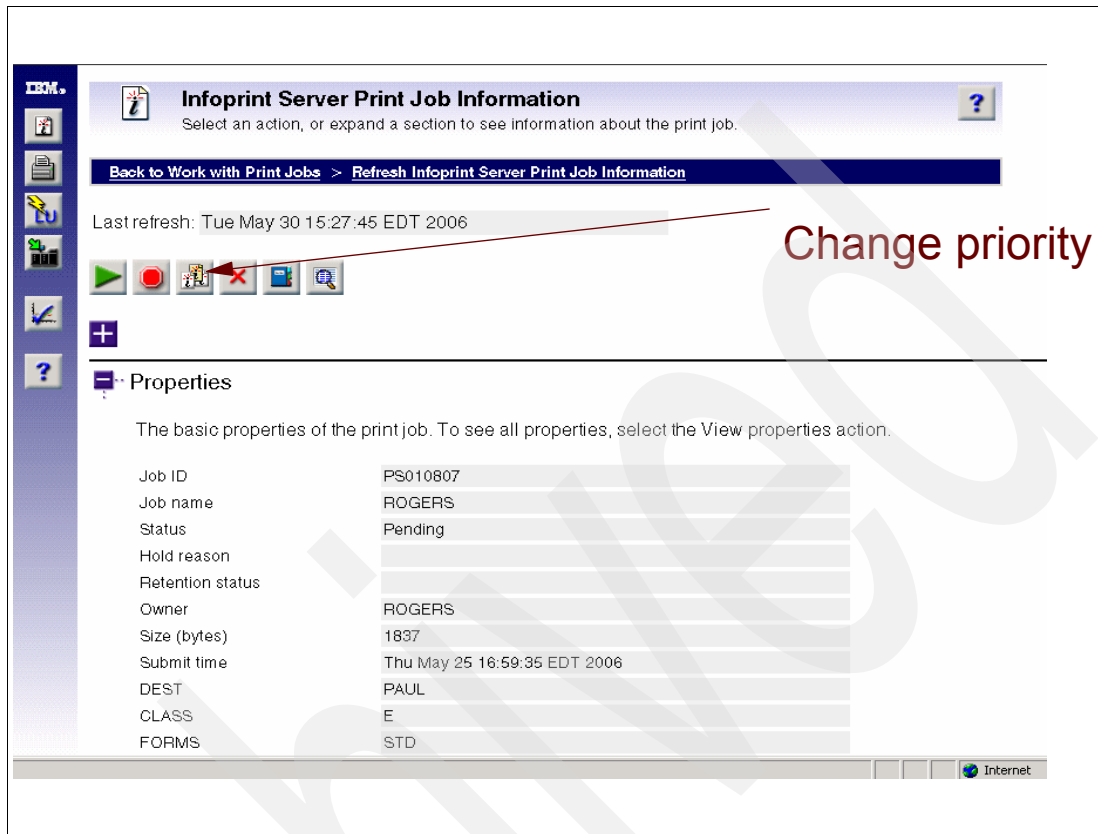


Figure 7-31 Infoprint Server Print Job Information panel

### Print job information

Figure 7-31 displays information about an Infoprint Server print job. This panel lets you work with the print job and to see detailed information about the print job. To work with a print job: Select an action button. When the action completes, you see an error message or an information message. To see detailed information: Select (Expand) to expand one of these sections:

- ▶ **Properties:** The basic properties of the print job. To see all properties and change some of them, select (View properties).
- ▶ **Documents:** The documents in the print job.
- ▶ **Printer:** The IP PrintWay printer where the print job prints. You see this section only if the print job is on the queue of an IP PrintWay printer. The printer is not displayed if the printer is not currently printing and no printer definition exists for the printer.
- ▶ **Printer Definition:** The printer definition used to process the print job.
- ▶ **NetSpool LU:** The NetSpool logical unit (LU) that received print data from a VTAM application and created the print job. You see this section only if the print request was submitted from a VTAM application (such as CICS and IMS).

### Change print output priority

To change the print job priority, Select the Change priority button.

## 7.32 Change print output priority

- ❑ This panel appears on top of the Print Job Information
  - The current priority of the print output is 15
  - Overtyping that to specify the new priority

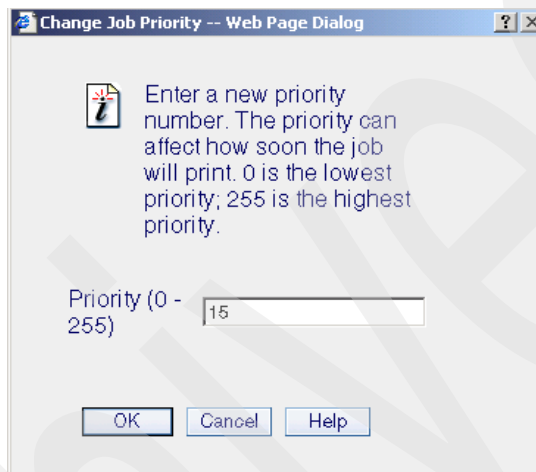


Figure 7-32 Panel to change print output priority

### Change output priority

The panel shown in Figure 7-32 lets you change the priority of a print job. The priority can affect how soon the print job will print.

To specify a new priority for the print job, do the following:

- ▶ Enter the new priority in the Priority field.
- ▶ Click **OK**.

### Print output priority

The priority of print output has a range between 0 and 255. This initial priority of the print output is the priority of the submitter, which is normally between 0 and 15.

## 7.33 Work with NetSpool Logical Units

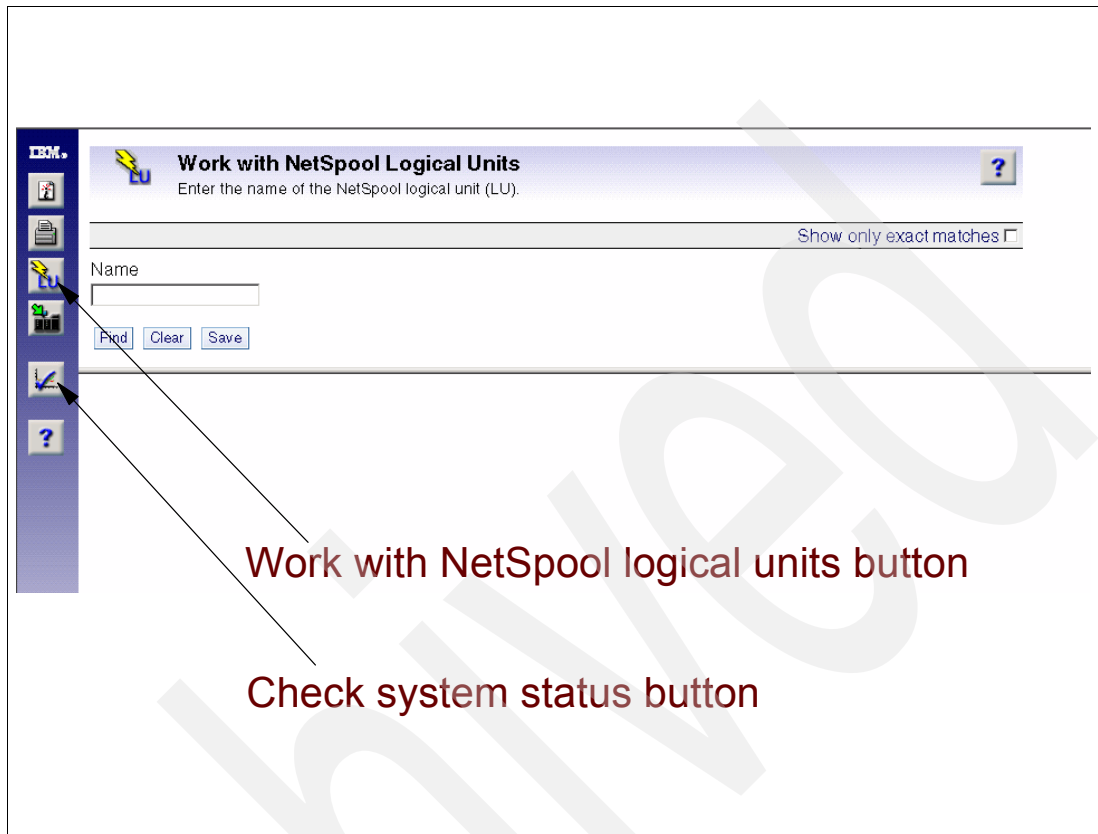


Figure 7-33 Work with NetSpool Logical Units panel

### Work with NetSpool Logical Units

The Work with NetSpool Logical Units panel lets you select the NetSpool logical units (LUs) you want to work with. You can work with any NetSpool LU that is defined in the Infoprint Server Printer Inventory.

### Find NetSpool LUs

Enter a value in the Name field to find LUs, as follows:

- ▶ Do not enter an asterisk (\*) or a question mark (?) as a wildcard symbol.
- ▶ Clear the Show only exact matches box if you enter only the first characters of the LU name instead of the full value.

Select **Save** to save the search values. The saved search values are displayed the next time you use this panel. Then, select **Find**.

**Attention:** Infoprint Central supports only one NetSpool task. If more than one NetSpool task is running on the z/OS system, do not use Infoprint Central to work with your NetSpool LUs. Instead, use NetSpool commands and VTAM commands so that you can direct NetSpool commands to the correct NetSpool task.

For information about NetSpool and VTAM commands, see *z/OS Infoprint Server Operation and Administration*, S544-5745. To see how many NetSpool tasks are running, select the Check system status button from the navigation bar.



## 7.34 System Status - System Daemons and Tasks

**System Status**  
Infoprint Server system level information

Last refresh: Wed May 17 18:25:57 EDT 2006

**System Daemons and Tasks**

The status of Infoprint Server daemons and NetSpool started tasks.

Printer Inventory AOP1

Component	Daemon or task name	Defined	Status
Printer Inventory Manager	Printer inventory daemon (aopd)	N/A	Started
	Message logging daemon (aoplogd)	N/A	Started
Print Interface	Line printer daemon (aoplpd)	Yes	Started
	IPP Server daemon (aoppipd)	Yes	Started
	Print Interface subsystem daemon (aopsubd)	Yes	Started
IP PrintWay	Output daemon (aopoutd)	Yes	Started
	Job selection daemon (aopwsmd)	N/A	Started
Transform Manager	Transform manager daemon (aopxtd)	Yes	Started
SNMP Subagent	SNMP subagent daemon (aopsnmpd)	No	
Infoprint Central	Command daemon (aopssid)	Yes	Started
	Inventory search database daemon (aopsdbd)	N/A	Started
	Historical job daemon (aophinvd)	N/A	Started

Figure 7-34 System Status - System Daemons and Tasks

### System Status - System Daemons and Tasks

The System Status panel displays information about the Infoprint Server system. This panel lets you see the following two buttons to expand:

- ▶ Check whether Infoprint Server daemons and tasks are started, which is shown in Figure 7-34.

You cannot start and stop daemons or tasks from this panel. Ask your operator to start daemons with the `aopstart` command or AOPSTART procedure.

- ▶ Check which IP PrintWay job selection rules are in effect. You can start and stop job selection rules from this panel, which is shown in Figure 7-35 on page 380.

To see detailed information select (+) **Expand** to expand one or more of these sections:

- ▶ **System Daemons and Tasks:** The status of all Infoprint Server daemons and NetSpool tasks. This section is displayed only if your administrator has given you authority to see the status of Infoprint Server daemons.
  - If the Defined field contains Yes or N/A and the Status field contains Stopped, the daemon or task might have ended abnormally or the operator might have stopped it. Ask your operator to start Infoprint Server daemons or the NetSpool task.
  - If the Defined field contains No and you think the daemon should be started, ask your administrator to define the daemon in the Infoprint Server configuration file and then restart Infoprint Server daemons. The file name of the Infoprint Server configuration file is `aopd.conf`.

## 7.35 System Status - IP PrintWay Job Selection Rules panel

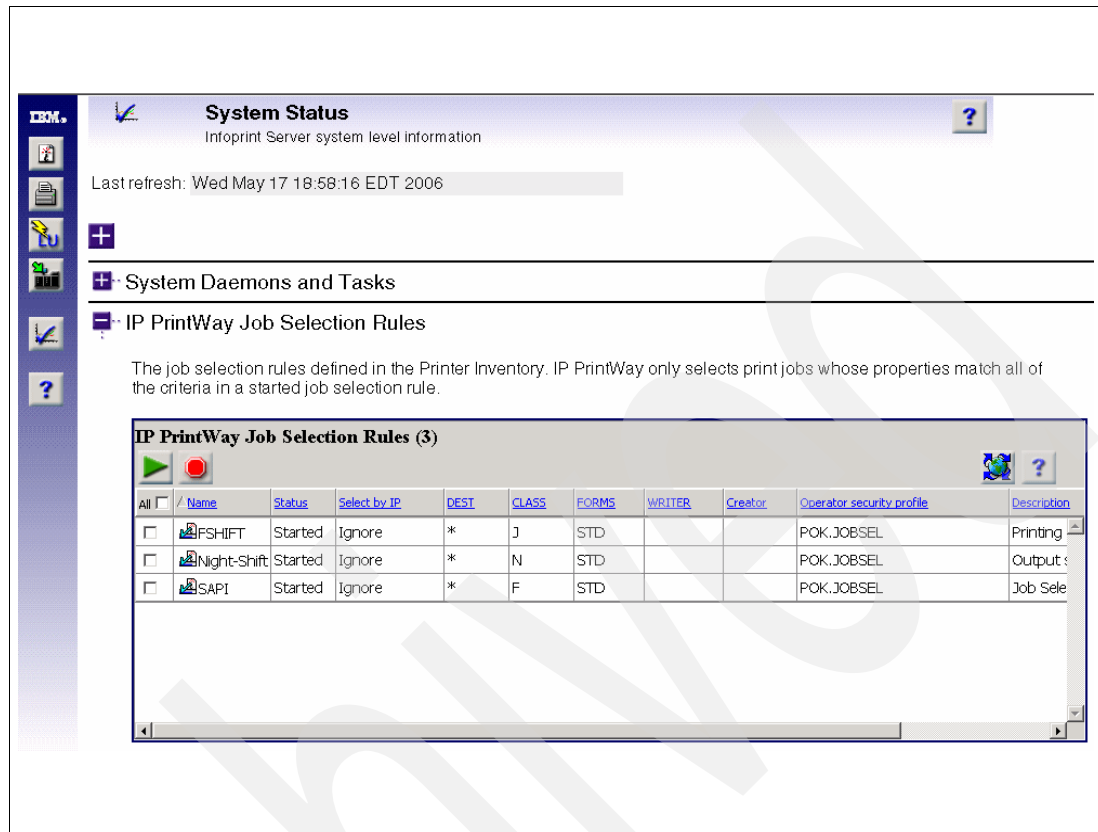


Figure 7-35 System Status - IP PrintWay Job Selection Rules

### System Status - IP PrintWay Job Selection Rules

The System Status panel displays information about the Infoprint Server system. This panel shown in Figure 7-35 is the check for which IP PrintWay job selection rules are in effect. You can start and stop job selection rules from this panel.

### IP PrintWay job selection rules

The job selection rules defined in the Printer Inventory. IP PrintWay only selects print jobs whose properties match all of the criteria in a started job selection rule. IP PrintWay uses the job selection rules to determine which print jobs to select from the JES spool for printing. If a job selection rule is started, IP PrintWay selects print jobs if they meet all the criteria in the rule. If more than one rule is started, and a print job meets the criteria in more than one rule, IP PrintWay prints the print job only one time. If no criteria are specified in a job selection rule, IP PrintWay selects all print jobs. If no job selection rules exist at all, IP PrintWay does not select any print jobs.

### Using this panel

This panel lists IP PrintWay job selection rules that you are authorized to see. This panel lets you check which job selection rules are started. You can also start and stop job selection rules on this panel.

To work with job selection rules, select the boxes next to the rules that you want to work with. To work with all rules in the list, select **All**.

## 7.36 Find Printer Definitions panel

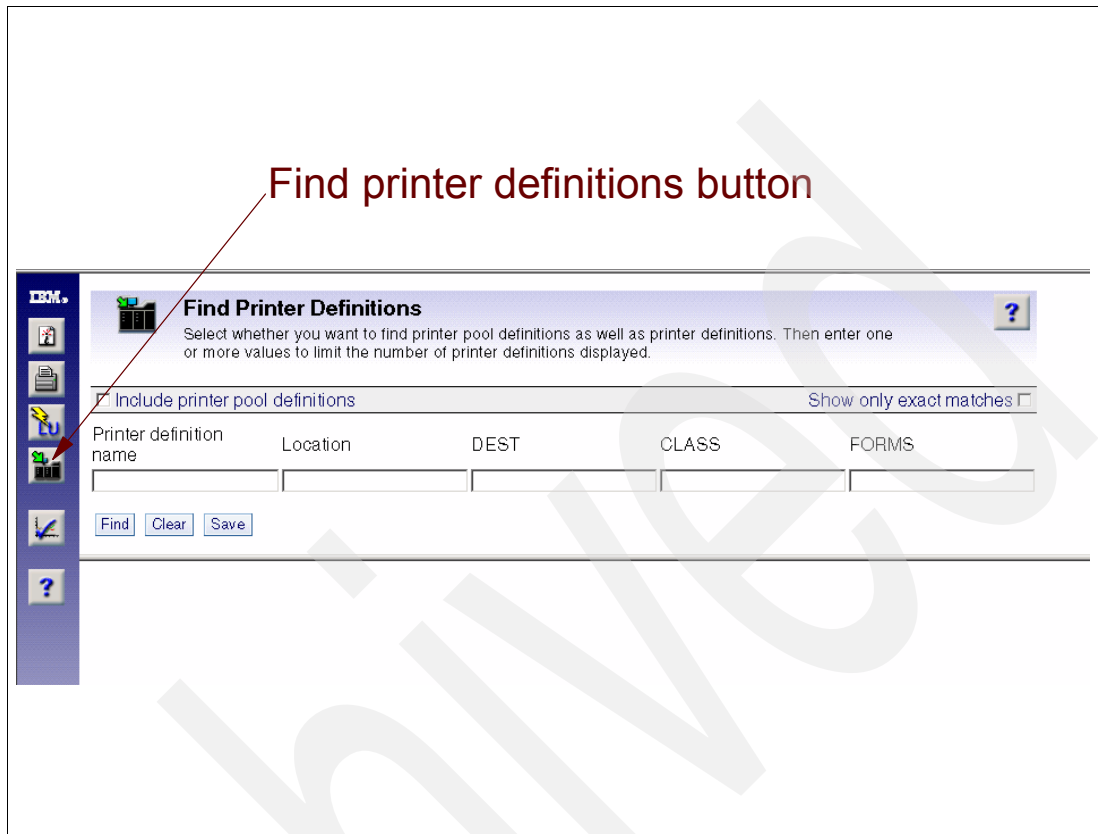


Figure 7-36 Find Printer Definition panel

### Find Printer Definition panel

The Find Printer Definition panel lets you find printer definitions and printer pool definitions contained in the Infoprint Server Printer Inventory.

#### Finding printer definitions

Select **Include printer pool definitions** to find printer pool definitions in addition to printer definitions.

Enter a value in at least one of the search fields. If you enter values in multiple fields, only printer definitions and printer pool definitions that match all of the values are displayed, as follows:

- ▶ Do not enter an asterisk (\*) or a question mark (?) as a wildcard symbol.
- ▶ The Clear button clears any fields where you have entered information.
- ▶ If you click **Save** to save these search values, the saved search values are displayed the next time you use this panel.
- ▶ Click **Find** when you have completed entering your search fields.

#### Usage considerations

To work with print jobs submitted to a particular printer definition, select **Work with print jobs** from the navigation bar. On the Work with Print Jobs panel, enter the name of the printer definition in the **Submitted to search** field. When you select **Work with print jobs**, you can

see all print jobs submitted to the printer definition, even print jobs that have completed processing and are no longer on the JES spool.

To work with a printer for which you know the printer definition name, select **Work with printers** from the navigation bar. On the Work with Printers panel, enter the name of the printer definition for the printer in the Printer name search field.

To find all of the printer definitions for a particular printer, select Work with printers from the navigation bar. On the Work with Printers panel, enter the name of one of the printer definitions for the printer in the Printer name search field. Then select the value in the Printer definition field to see all printer definitions for the printer

## 7.37 Printer Definitions

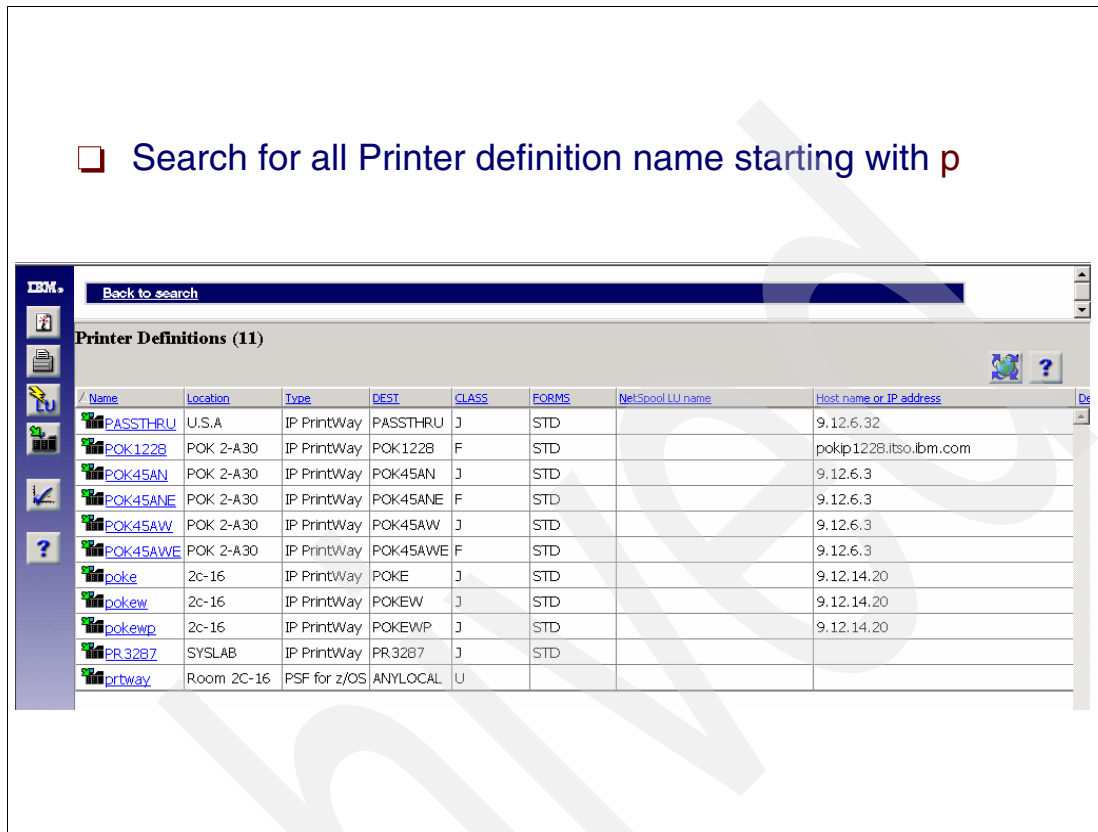


Figure 7-37 Printer Definitions panel

### Printer definitions

The Printer Definitions panel lists printer definitions. This panel lets you see information about several printer definitions at the same time and link to more detailed information about one printer definition.

### Using printer definitions

To see detailed information about a printer definition, Select the value in the Name field for the printer definition.

To see the NetSpool logical unit associated with a printer definition, Select the value in the NetSpool LU name field for the printer definition.

### Usage considerations

Initially, the list contains a maximum of 25 printer definitions. The total number of printer definitions that meet the search criteria is displayed in parentheses after the title. To see all printer definitions up to a maximum number that your administrator determines, scroll down to the end of the list and select **More**.

To refresh the list of printer definitions, select **Refresh**.

The panel shows the basic properties of the selected printer.

Archived

## User interfaces to Infoprint Server

In today's network environments, printers are often attached to a single workstation or are only available to users of a LAN. Infoprint Server lets you define all of your printers in a centralized repository. Any user in the network can send print jobs from z/OS and LAN clients to any printer that is defined to Infoprint Server.

Because all components of Infoprint Server use the same printer definition, you only have to configure each printer in one place. Users and application programs in a z/OS network, including LAN and z/OS UNIX System Services environments, can take full advantage of Infoprint Server's many benefits, including:

- ▶ Access all defined printers
- ▶ Handle print jobs effectively
- ▶ Detect and transform data streams
- ▶ Support common printer languages
- ▶ Monitor printer status
- ▶ Query job status
- ▶ Create AFP output from Windows applications
- ▶ Browse AFP documents on the Web
- ▶ Send print output to e-mail addresses
- ▶ Work with print jobs and printers

## 8.1 Printing from z/OS UNIX System Services

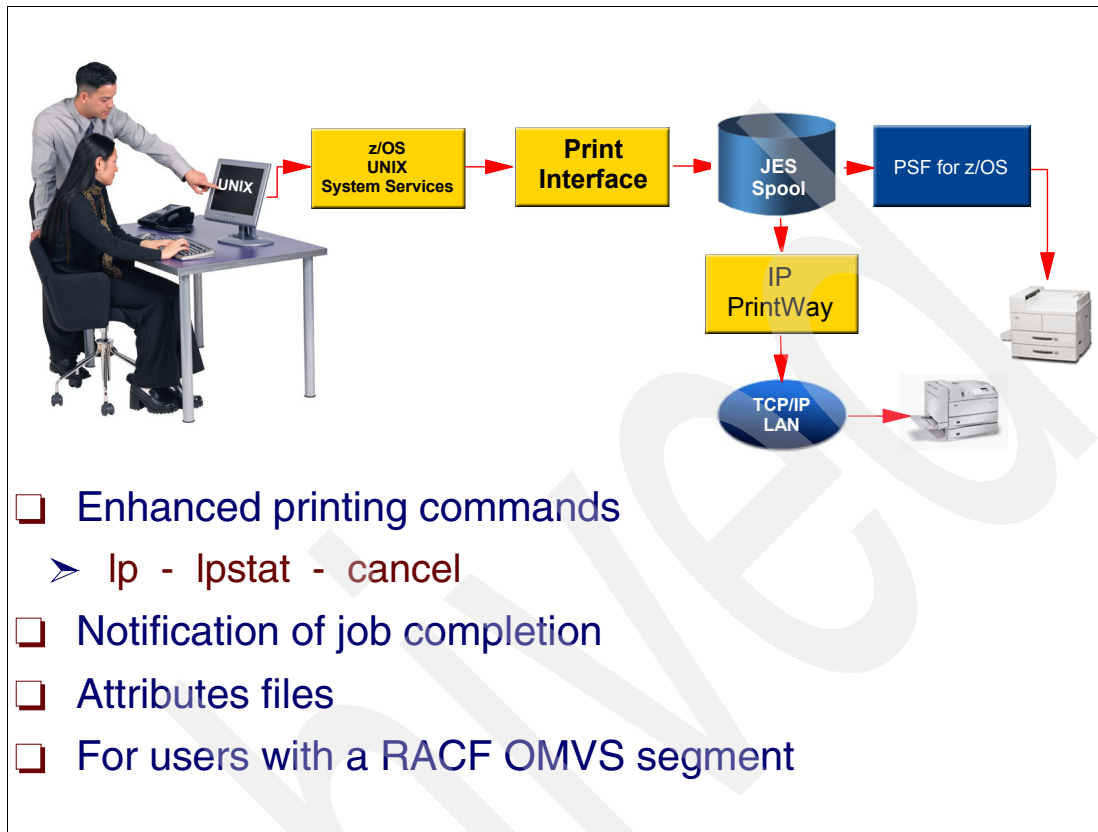


Figure 8-1 Printing from z/OS UNIX System Services

### Printing from z/OS UNIX System Services

Using the printing commands, you can print UNIX files and MVS data sets on any printer that your administrator has defined in a printer definition in the Infoprint Server Printer Inventory. You can print on local printers that are attached directly to z/OS, or on remote printers in a TCP/IP or SNA network. Instead of printing, you can use the same printing commands to send the data to any e-mail destination that your administrator has defined in a printer definition.

### Printing commands

The printing commands provide enhanced function over the commands of the same name that are described in z/OS UNIX System Services Command Reference. For example, when you print on IBM Advanced Function Presentation printers, you can specify options such as duplexing or a special overlay. You can also display the status of your print request, and you can cancel a print request. The printing commands adhere to the UNIX standards in XPG4.2, so you do not need to change the printing commands in your UNIX applications when you port them to z/OS.

The printing commands let you print, query, and cancel the printing of files, and let you send files to an e-mail destination instead of to a printer:

- ▶ `lp` - Print a file
- ▶ `cancel` - Cancel a print job
- ▶ `lpstat` - Show printer names and locations and status of print jobs



## 8.2 z/OS UNIX user prints a file

User issues **lp** to print a file

```
VAIN @ SC43:/> lp -d IAZFSS /tmp/one.line
AOP007I Job 838 successfully spooled to IAZFSS.
VAIN @ SC43:/>
```

User issues **lpstat** to obtain status

```
VAIN @ SC43:/> lpstat
Printer: IAZFSS
Job      Owner      Status      Format      Size      File
-----
838 VAIN      processing  line        12 /tmp/one.line
VAIN @ SC43:/>
```

User wants to **cancel** job

```
VAIN @ SC43:/> cancel 838
```

Figure 8-2 z/OS UNIX user printing a data set and managing the data set

### lp command

The **lp** command prints one or more files, or sends the files to an e-mail destination. The format of the command is as follows:

```
lp [-cmsw] [-d destination] [-n copies] [-o option] [-t title] [filename ...]
```

The address of the printer is specified in the printer definition in the Infoprint Server Printer Inventory, which your administrator manages. The e-mail addresses are specified in the printer definition or in job attributes. The files can be:

- ▶ MVS data sets, such as partitioned data sets or sequential data sets
- ▶ UNIX files, such as files in a Hierarchical File System (HFS), a zSeries File System (zFS), a Network File System (NFS), or a temporary file system (TFS)
- ▶ Lists of printable files

The **lp** command returns an Infoprint Server job ID, which you can use to query or cancel the job. The exact job ID in the system is PS000838, where PS is the prefix defined in the configuration file `aopd.conf` and the following message is issued to the submitter.

```
AOP007I Job 838 successfully spooled to IAZFSS.
```

**Note:** If you do not specify any files on the command line, or if you specify a dash (-) for the file name, **lp** prints from standard input.

The job ID number returned by the **lp** command cannot be used in JES commands.

## lpstat command

After Infoprint Server accepts the print job and the **lp** command returns an Infoprint Server job ID, you can use to query the print job using the **lpstat** command and cancel the print job using the **cancel** command.

The **lpstat** command shows the printer names, locations, and status of print jobs. The command writes printer definition names, location information specified in the printer definitions, and the status of jobs to standard output.

The format of the **lpstat** command is as follows:

```
lpstat [-dt] [-a [prntername ...]] [-o [prntername ...]]  
      [-p [prntername ...]] [-u [userid ...]] [jobid ...]
```

## cancel command

The **cancel** command cancels one or more print jobs that you submitted, with these restrictions:

- ▶ You can only cancel your own jobs.
- ▶ You cannot cancel a job after it has started processing.
- ▶ In a JES3 environment, you might not be able to cancel a job that is held on the JES3 spool.

The **cancel** command has the following format:

```
cancel jobid ...
```

*jobid* is the Infoprint Server job ID of the print job you want to cancel. If you do not know the Infoprint Server job ID, you can determine it by using the **lpstat** command to query all the jobs you submitted.

**Note:** In a sysplex the z/OS UNIX printing commands are available only on systems where the Infoprint Server is active.

## Online help for Infoprint Server commands

To get online help about Infoprint Server commands, use the **man** command. You can view man pages only in English. If the correct man pages are not displayed, specify this path on the **-M** option of the **man** command, or add it to your **MANPATH** environment variable ahead of other values:

```
/usr/lpp/Printsrv/man/En_US
```

## 8.3 lpstat command options

```
ROGERS @ SC67:/> lpstat -a
```

Printer	Jobs	Location	Description
NS	0	2C-16	VTAM NetSpool SCS Printer
PASSTHRU	0	U.S.A	Back to host
POK1228	0	POK 2-A30	IBM Infoprint Color 1228
POK45AN	0	POK 2-A30	1145A Portrait (Extended Mode)
POK45ANE	0	POK 2-A30	1145A Posrtrait (Extended Mode)
POK45AW	0	POK 2-A30	1145A Landscape/Rotated
POK45AWE	0	POK 2-A30	1145A Landscape/Rotated/Duplex
PR3287	0	SYSLAB	Test for OSA-ICC
ROSS	0	POK 2-A30	1145A Landscape/Rotated

-d	Query default printer	lpstat -d
-o	Query specified printer and jobs	lpstat -o poke
-p	Query specified printer	lpstat -p poke
-t	Query all printers and jobs	lpstat -t
-u	Query all printers and jobs by user ID	lpstat -u ROGERS
-a	Query names and locations of all printers	

Figure 8-3 z/OS UNIX user using the lpstat command

### lpstat command options

For jobs that Infoprint Server has processed (including jobs submitted in any of these ways: from a VTAM application through NetSpool; from a remote system, with the **lp** command through Print Interface, from batch JCL printed by IP PrintWay extended mode, or using the Print Interface subsystem), the **lpstat** command returns the information shown in Figure 8-3.

Following are the **lpstat** command options:

-d	Query default printer	lpstat -d
-o	Query specified printer and jobs	lpstat -o poke
-p	Query specified printer	lpstat -p poke
-t	Query all printers and jobs	lpstat -t
-u	Query all printers and jobs by user ID	lpstat -u ROGERS
-a	Query names and locations of all printers	

## 8.4 z/OS UNIX transform commands

- ❑ The transform commands transform data from one data format to another without printing it
  - `afp2pcl` --Transform AFP or line data to PCL data
  - `afp2pdf` --Transform AFP or line data to PDF data
  - `afp2ps` --Transform AFP or line data to PostScript data
  - `pcl2afp` --Transform PCL data to AFP data
  - `pdf2afp` --Transform PDF to AFP data
  - `ps2afp` --Transform PostScript data to AFP data
  - `sap2afp` --Transform SAP OTF or ABAP data to AFP data
  - `xml2afp` --Transform XML to AFP data
  - `xml2pdf` --Transform XML to PDF data
  - `x2afp` --Transform Xerox files to AFP data

Figure 8-4 Transform commands

### Transform commands

While Infoprint Server lets you submit data in many different formats, Advanced Function Presentation (AFP) printers print the AFP data stream. You can submit non-AFP data streams to AFP printers using these optional products, which convert jobs to AFP format:

- ▶ Infoprint Server Transforms transform data streams such as PCL, PDF, PostScript, and SAP to AFP format and vice versa.
- ▶ IBM Infoprint XML Extender for z/OS (5655-J66) transforms Extensible Markup Language (XML) files to AFP and PDF format.
- ▶ IBM Infoprint XT Extender for z/OS (5655-J65) transforms Xerox files to AFP format. The Xerox files can be line-conditioned data streams (LCDS) or metacode data streams.

Documents in AFP format are also called Mixed Object Document Content Architecture Presentation (MO:DCA-P) documents.

Usually, you do not have to worry about transforming your data to another format. If Infoprint Server Transforms is installed, Infoprint Server automatically calls the appropriate transform when you submit a print request to a printer definition (for a printer or for an e-mail destination) that your administrator has configured for transformation. You might, however, want to transform a file without printing it in these situations:

- ▶ You want to verify that the job can be transformed without errors.

- You intend to print a file many times. In this case, it is more efficient to transform the file once and print the output than to transform the file every time you print it.

## Transforms that convert AFP output

Three products of Infoprint Server Transforms convert print data from AFP format into PCL, PDF, and PostScript. These products let you print data in AFP format on PCL and PostScript printers, and transform an AFP file to PDF format for viewing on a workstation. Documents in AFP format are also called Mixed Object Document Content Presentation Architecture (MO:DCA-P) documents and you may want to present your document on the Web.

Infoprint Server automatically transforms print data in other formats to the Advanced Function Presentation (AFP) data stream when you submit them to a printer definition that the print administrator has configured to do so.

## z/OS UNIX commands to transform data

You can also use the **pc12afp**, **pdf2afp**, **ps2afp**, and **sap2afp** commands to transform files in these formats without printing them:

- Printer Control Language (PCL)
- Portable Document Format (PDF)
- PostScript
- SAP Advanced Business Application Programming (ABAP)
- SAP Output Text Format (OTF)

## Transform command examples

For example, to transform the PostScript file **myfile.ps** to an AFP file called **myfile.afp**, with each page 5.5 inches long and 4 inches wide, enter:

```
ps2afp -o myfile.afp -l 5.5i -w 4i myfile.ps
```

To submit the PCL file **sample.pcl** to the printer named **IAZFSS** and transform it automatically, enter:

```
lp -d IAZFSS sample.pcl
```

The IAZFSS printer definition is as follows:

### IAZFSS printer definition Processing section includes:

```
AOIPDRR                               Processing
Command ==>
Printer definition name . IAZFSS
Document code page . .
Printer code page. . . IBM-1047
Supported Data Formats and Associated Filters:
Data format:   Filter:
/ Line data
(extend)
::::::::::::::::::::::::::::::::::::
/ PCL          pc12afp.dll %filter-options
(extend)
```

## 8.5 Infoprint Server and transforms

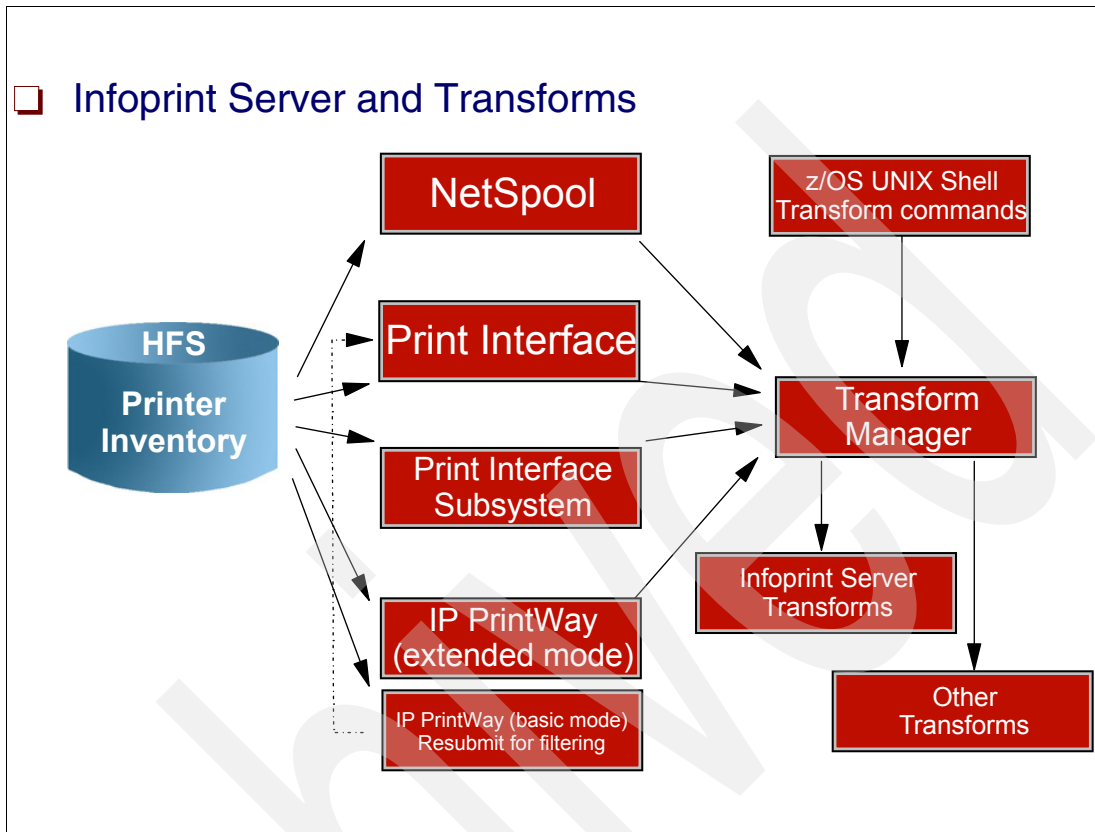


Figure 8-5 Infoprint Server handling transforms

### Transforming and printing output

When processing a print request, Print Interface, NetSpool, and IP PrintWay extended mode call Transform Interface to transform data from one format to another if the administrator specifies a transform filter in the printer definition. The administrator specifies a different transform filter for each data format.

The ways that Infoprint Server uses transforms and prints output data sets include:

- ▶ **Print Interface subsystem:** The Print Interface subsystem transforms data before writing it to an output data set on the JES spool. IP PrintWay or PSF then prints the data, or IP PrintWay sends it to an e-mail destination. To use the Print Interface subsystem, you specify the SUBSYS parameter on the DD JCL statement for the output data set.
- ▶ **IP PrintWay extended mode:** IP PrintWay extended mode transforms data in an output data set before it prints the data or sends it to an e-mail destination.
- ▶ **IP PrintWay basic mode:** IP PrintWay basic mode sends data in an output data set to Print Interface. Print Interface transforms the data and writes the transformed data to a new output data set on the JES spool. IP PrintWay then prints the data or sends it to an e-mail destination.
- ▶ **NetSpool:** NetSpool transforms SCS and 3270 data streams to EBCDIC line data or ASCII PCL data streams. Also, it uses other Infoprint transform products to convert line data to other formats for printing or e-mailing.

## 8.6 Infoprint Server job attributes

- ❑ Infoprint Server job attributes describe special requirements called attributes:
  - Whether to print on one or both sides of the paper
  - Fonts, page definitions, form definitions, overlays
  - Text to print on sheet paper or the subject of the e-mail
- ❑ Use the **-o** option of the **lp** command to specify attributes

```
lp -d poke -o "input-tray=top duplex=yes
overlay-front=O1ODD overlay-back=O1EVEN
resource-library=MYOVR.LIBRARY" special.job
```

Attributes can be stored in an attributes file

```
lp -d IAZFSS -o attributes=myatts special.job
```

Figure 8-6 Infoprint Server job attributes

### Infoprint Server job attributes

You can list any job attribute in an attributes file for the **lp** command. You can also list the attribute attributes. Thus, an attributes file can call other attributes files. If an attributes file calls itself, the command sends an error message. Attributes files must not contain any attributes without values.

When creating an attributes file, consider spelling out the complete attribute names and attribute values rather than using abbreviations. You can use spaces between the attribute name and the equals sign to align the equals sign and values. This makes your files easier to read and maintain. You can use comment lines in attributes files. The comment starts with a number sign, #, and ends at the end of line.

### Job attribute example

You could create an attributes file called `myatts` to request 5 copies of a job, simple duplex printing, and a specific output bin.

```
# These are my job attributes
copies      = 5
duplex      = yes
output-bin  = collator # Collate the job
```

You can include a number sign, #, as part of an attribute value if you precede it immediately with a back slash (\#).

## 8.7 Job attributes and JCL equivalents

- address-text	ADDRESS	- output-bin	OUTBIN
- building-text	BUILDING	- output-bin-number	OUTBIN
- carriage-control-type	RECFM	- overlay-back	OVERLAYB
- chars	CHARS	- overlay-front	OVERLAYF
- copies	COPIES	- owner	USER
- department-text	DEPT	- page-definition	PAGEDEF
- document-codepage	none	- print-error-reporting	DATACK
- document-format	none	- print-queue-name	PRTQUEUE
- document-type	none	- printer-ip-address	DEST=IP:
- duplex	DUPLEX	- resource-library	USERLIB
- filter-options	none	- restrict-printable-area	SYSAREA
- form-definition	FORMDEF	- room-text	ROOM
- forms	FORMS	- save-afp-statistics	AFPSTATS
- hold	HOLDIOUTDISP(JES2)	- shift-out-shift-in	PRMODE
- input-tray	INTRAY	- sysout-dataset-name	DSNAME
- input-tray-number	INTRAY	- sysout-job-id	none
- jes-priority	PRTY	- sysout-job-name	job name
- mail-bcc-addresses	MAILBCC	- table-reference-characters	DCB OPTCD=J
- mail-cc-addresses	MAILCC	- table-reference-characters	TRC
- mail-file-name	MAILFILE	- title-text	TITLE
- mail-from-name	MAILFROM	- x-image-shift-back	OFFSETXB
- mail-reply-address	REPLYTO	- x-image-shift-front	OFFSETXF
- mail-to-addresses	MAILTO	- y-image-shift-back	OFFSETYB
- name-text	NAME	- y-image-shift-front	OFFSETYF

Figure 8-7 Job attributes and JCL equivalents

### Job attributes and JCL equivalents

If you have previous experience with z/OS, you are accustomed to using the OUTPUT and DD statements of the Job Control Language (JCL) to specify processing options for print jobs.

Infoprint Server uses attributes to describe jobs and the documents in jobs. For example, you can specify the number of copies of a document to print by setting a value for the copies attribute.

#### Job attributes

Many job attributes correspond to parameters of the OUTPUT JCL statement. A few correspond to parameters of the DD and JOB JCL statements.

The JCL OUTPUT and DD statements parameters are available in a TSO/E session through OUTDES and ALLOC commands.

Figure 8-7 lists job attributes and corresponding JCL parameters.



## 8.8 Using job attributes

- ❑ z/OS UNIX lp command - -o option
- ❑ z/OS UNIX afp2pcl, afp2pdf, afp2ps commands - -j option
- ❑ Print Interface subsystem - SUBSYS parm in JCL DD
- ❑ AOPPRINT JCL procedure - OPTIONS parm on EXEC
- ❑ NetSpool - Embed job attributes
- ❑ Infoprint Port Monitor - Specify in the Options window
- ❑ AIX enq command - -o option
- ❑ OS/400 LPR command - DESTOPT option
- ❑ OS/400 remote queue - CRTOUTQ - Dest options field
- ❑ lprafp command - -o option

Figure 8-8 Using job attributes from different platforms

### Using job attributes

A job is a set of one or more documents that you submit to Infoprint Server for printing in a single printing session. A document is either a file or a group of similar files. Infoprint Server uses attributes to describe jobs and the documents in jobs. For example, you can specify the number of copies of a document to print by setting a value for the copies attribute. You can specify job attributes in different ways, depending on the method you use to submit a print job. Figure 8-8 shows how to specify job attributes for different job submission methods and refers you to the section that contains more information.

### Using attributes to print

Attributes are only one of the factors that determine how your job is printed. These values also affect your job:

- ▶ Print command options. For example, you can use the -n option of the lp command to specify the number of copies of a job.
- ▶ Values in the data stream. For example, the document can specify an overlay.
- ▶ Values in the page definition used to print the job. For example, the page definition can specify fonts.
- ▶ Values in the form definition used to print the job. For example, the form definition can specify duplex printing.
- ▶ Printer specifications in Infoprint Server lets the administrator specify printer characteristics for each printer.

## 8.9 Job attributes and NetSpool

- ❑ Embedding job attributes in print data
  - Owner and other job attributes
- ❑ Printer definition can contain an owner default attribute
  - If not in print data or printer definition
    - Owner is user ID who starts NetSpool
- ❑ Owner attribute allows the following:
  - Identify owner of print jobs or output or workstation name
  - Can use owner name on header page
  - For accounting purposes

Figure 8-9 Using job attributes with NetSpool

### NetSpool and job attributes

You can embed the owner job attribute and other job attributes for NetSpool jobs in print data. If you do not embed a job attribute in print data, your administrator can specify a default value for the attribute in the printer definition. If the owner is not embedded or specified in the printer definition, the default value is the ID of the user who started the Infoprint Server daemons.

### Imbedding owner attribute

You might want to embed the owner job attribute for a print job to help you identify the owner of print jobs in Infoprint Central or the owner of printed output. For example, the ID or name of the user who is printing a job, or the name of the workstation where the user submitted the print request. The identifying information can be printed on the print job's separator page and on a page header. The owner attribute can be used to charge in an accounting system for printing jobs.

To embed the owner job attribute or other job attributes in print data, you code a text string that contains the job attributes. Be sure to follow the instructions for coding text strings in *z/OS Infoprint Server User's Guide*, S544-5746. See also "Embedding job attributes in the NetSpool print data" on page 302 and "Defining NetSpool printers" on page 313.

NetSpool uses the embedded attributes, and then removes the text string from the data stream before it converts the data stream and puts it on the JES spool.

## 8.10 Printing from batch applications using IP PrintWay

❑ z/OS UNIX user submits batch job

- Output directed to IP PrintWay using DD and/or OUTPUT JCL
- Processed by Infoprint Server for the output

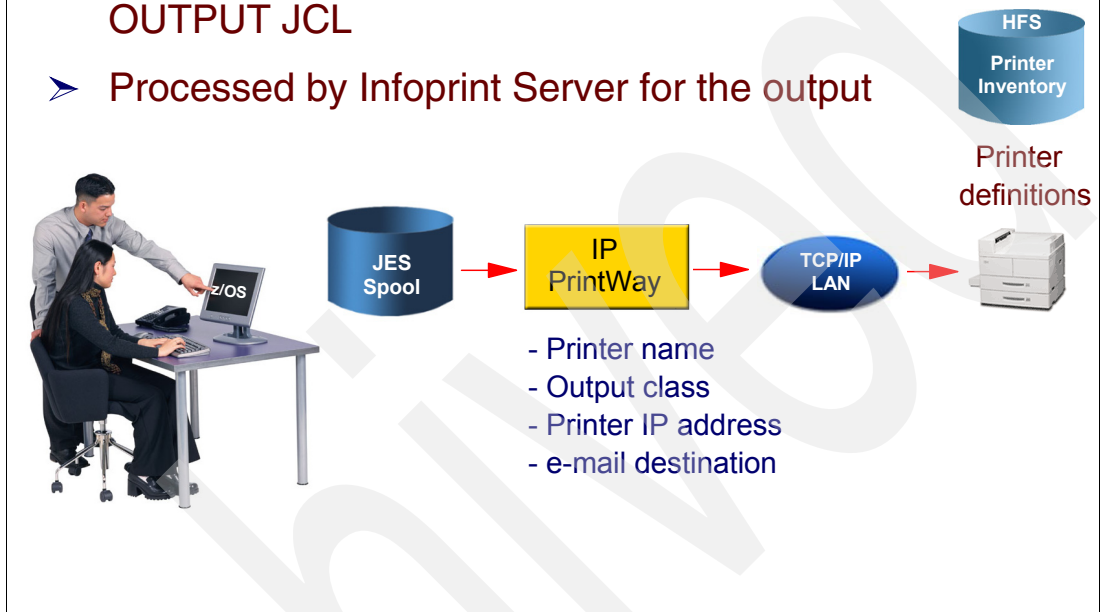


Figure 8-10 Printing from batch applications

### Printing from batch applications using IP PrintWay

To direct an output data set to IP PrintWay, on the DD or OUTPUT JCL statement for the data set you must specify either:

- ▶ The work-selection criteria that your administrator defined to JES for the IP PrintWay basic mode functional subsystem application (FSA)
- ▶ The job-selection criteria that your administrator defined in the Printer Inventory for IP PrintWay extended mode

### Printer Inventory has definitions

If you use the Print Interface subsystem, your administrator can specify the selection values in the printer definition in the Printer Inventory. In this case you do not need to specify them in JCL. For example, if your administrator specifies class P as the output class in the printer definition, you do not need to specify class P on your OUTPUT statement.

To direct your output to a particular printer or e-mail destination, you can specify either:

- ▶ The name of the printer definition your administrator created for the printer or e-mail destination
- ▶ The output class, destination name, or form name assigned to the printer definition
- ▶ The printer's IP address and either print queue name or port number

## 8.11 OUTPUT JCL for IP PrintWay

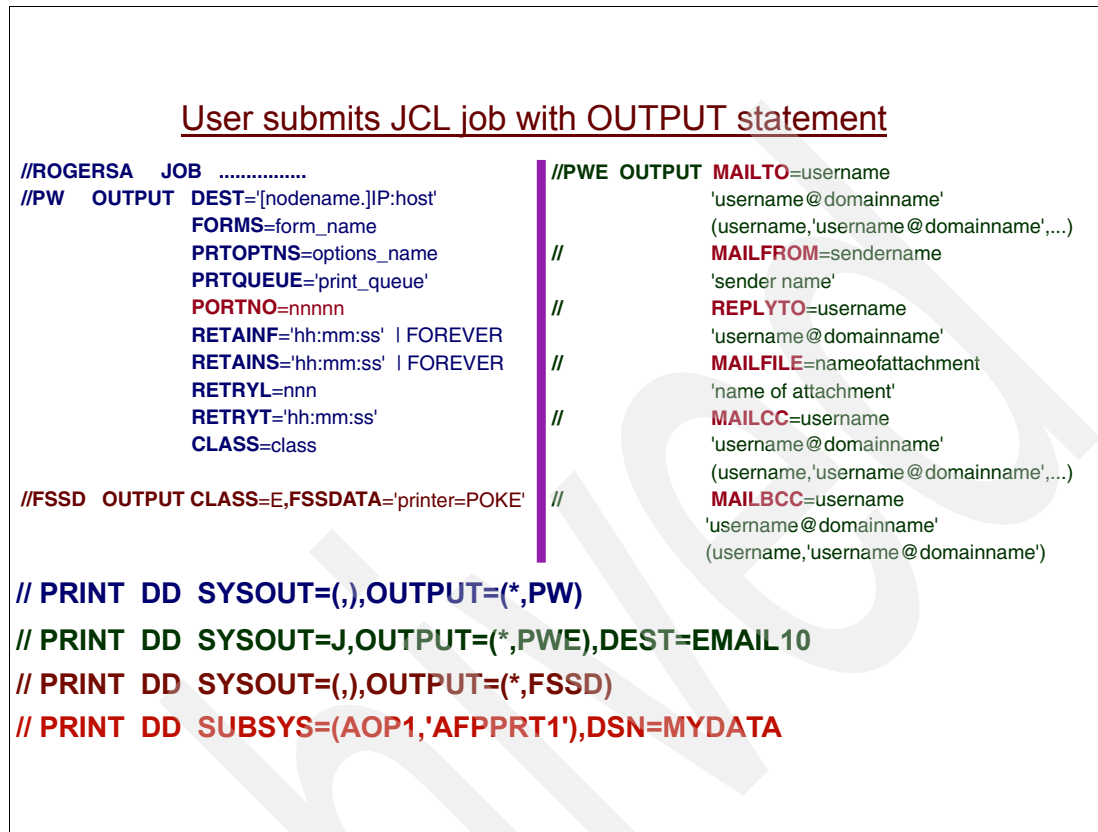


Figure 8-11 Printing from batch to IP PrintWay with IP address or e-mail address

### OUTPUT JCL statement

IP PrintWay can use the IP address or the e-mail address in the named printer definition to find the printer or e-mail destination. However, if you specify an IP address in the DEST=IP: parameter of the OUTPUT JCL statement, or an e-mail address in the MAILTO parameter of the OUTPUT JCL statement, IP PrintWay uses that address to find the printer or e-mail destination instead of the address in the printer definition.

**Note:** You might want to specify the DEST=IP: parameter or MAILTO parameter when your administrator has not created a printer definition for your printer or e-mail destination.

### Specifying the printer definition name

You can specify the name of the printer definition that your administrator created for the printer or e-mail destination in the Printer Inventory. Using this definition as follows:

- ▶ You specify the printer definition name in the FSSDATA='printer' parameter on the OUTPUT JCL statement, as shown in Figure 8-11. You can use the FSSDATA='printer' parameter with both IP PrintWay basic mode and IP PrintWay extended mode.
- ▶ However, if you use the Print Interface subsystem, you instead specify the printer definition name in the SUBSYS parameter on the DD JCL statement. The printer definition name is case-sensitive and must exactly match the name of the printer definition created by your administrator in the Infoprint Server Printer Inventory.

## Specifying the printer's IP address

To print a data set on a printer for which your administrator has not created a printer definition, specify the `DEST=IP:` parameter on the `OUTPUT JCL` statement. Specify the IP address or host name of the printer or the system to which the printer is attached. When you specify `DEST=IP:`, you must also specify either the `PRTQUEUE` parameter or the `PORTNO` parameter on the `OUTPUT JCL` statement.

If you also specify the name of a printer definition in either the `FSSDATA` or `SUBSYS` parameter, IP PrintWay uses printing options specified in that printer definition, but uses the printer's IP address specified in the `DEST=IP:` parameter. If you do not specify the name of a printer definition, IP PrintWay uses printing options specified in the IP PrintWay default printer definition.

If the resubmit for filtering function is used in IP PrintWay basic mode, do not code the `DEST=IP:`, `PRTQUEUE`, and `PORTNO` parameters on the `OUTPUT JCL` statement because IP PrintWay ignores these parameters and instead uses the IP address, print queue name, and port number in the printer definition.

## Sending output to an e-mail destination

When you submit a print request, you can send the file to an e-mail destination instead of to a printer. You can specify the e-mail addresses of the primary and secondary recipients in one of these ways:

- ▶ Your administrator can specify the e-mail address list of the recipients in the printer definition in the Printer Inventory.
- ▶ You can specify the e-mail addresses using optional JCL parameters.
- ▶ You can specify an address list in an alias file that your administrator defines to z/OS UNIX sendmail.

The e-mail has these characteristics:

- ▶ Each output data set is a separate e-mail attachment. The name of the attachment is the value specified in one of these:
  - The `MAILFILE` parameter on the `OUTPUT` statement for the output data set
  - The `DSNAME` parameter on the `DD` statement for the output data set
  - The job name
- ▶ You can specify these optional `OUTPUT JCL` parameters to customize your e-mails:

<b>MAILBCC</b>	The e-mail addresses of the “blind copy (bcc)” recipients of an e-mail. A bcc means that other recipients of the e-mail do not see the bcc recipient listed.
<b>MAILCC</b>	The e-mail addresses of the “copy (cc)” recipients of an e-mail. A cc means that other recipients of the e-mail can see the cc recipient listed.
<b>MAILFILE</b>	The file name of the attachment to an e-mail.
<b>MAILFROM</b>	The descriptive name or other identifier of the sender of an e-mail.
<b>MAILTO</b>	The e-mail addresses of the recipients of an e-mail.
<b>REPLYTO</b>	The e-mail address that recipients of an e-mail can reply to.
<b>TITLE</b>	The subject of the e-mail.

## 8.12 Infoprint Server sendmail support

- ❑ Before you begin using e-mail:
  - Customize sendmail
- ❑ Submit a batch job to print AFP data to e-mail address
  - With basic mode - requires transform in printer definition

```
//ROGERSZ JOB (POK,999),MSGCLASS=A,NOTIFY=ROGERS
//PRINT EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD SYSOUT=J,DEST=EMAIL10
//SYSUT1 DD DISP=SHR,DSN=ROGERS.SAPI.LIST3820
//SYSIN DD DUMMY
//
```



Figure 8-12 Submitting a batch job to send output to an e-mail address

### z/OS UNIX sendmail support

When the e-mail protocol is selected in a printer definition, IP PrintWay uses z/OS UNIX *sendmail* to prepare and send e-mail to the recipients listed in the printer definition. Sendmail is a mail transfer agent provided with z/OS Communications Server that provides enhanced SMTP support.

### Printer Inventory

The IP PrintWay printer to process the output to the e-mail destination is defined by the administrator in the Printer Inventory.

### Batch job to send e-mail output

Figure 8-12 shows a batch job that when submitted does the following:

- ▶ During execution using the IEBGENER program, it takes the AFP data set, ROGERS.SAPI.LIST3820 and creates an output data set on the JES spool.
- ▶ When the job ends, the output data set created on the spool is selected by IP PrintWay because the destination, DEST=EMAIL10, and the SYSOUT class J are specified.
- ▶ The processing done by IP PrintWay is illustrated in Figure 8-15 on page 403.

## 8.13 Processing options for e-mail printers

```
AOPIPDRR                                     Processing
Command ==> _____

Printer definition name . EMAIL10             AFP output

Document code page . . _____
Printer code page. . . IBM-1047

Print Interface Supported Data Formats and Associated Filters:
Data format:  Filter:
/ Line data   afp2pdf.dll %filter-options      (extend)
/ M0:DCA-P    afp2pdf.dll %filter-options      (extend)
- PostScript  _____                     (extend)
/ Text        aopfiltr.so                     (extend)
- PCL         _____                     (extend)
- PDF         _____                     (extend)
- SAP         _____                     (extend)
- Other       _____                     (extend)
/ Resubmit for filtering

SCS Conversion:
Margins: Top . . . . . Bottom . . . . . Left . . . . . Right . . . . .
Line length . . . . . Page length . . . . .
Tabs: Vertical . . . . . (extend)
      Horizontal . . . . . (extend)
```

Figure 8-13 Processing panel to define a printer in the Printer Inventory

### Defining Processing panel options for an e-mail printer

This ISPF panel, shown in Figure 8-13, shows how to specify the valid data formats for a printer that can print AFP and text data. Notice that Line data is selected also as AFP data type. This EMAIL10 printer in the Filter field converts AFP data to a PDF using the transforms.

### Resubmit for filtering option

This single-valued attribute indicates whether a filter in the filters attribute is to be used for data sets submitted as batch jobs to IP PrintWay basic mode. When resubmit-for-filtering=yes, IP PrintWay resubmits batch data sets to Print Interface. Print Interface calls the filter (if any) associated with the input data format and then writes the data to a new output data set on the JES spool for subsequent processing by IP PrintWay.

**Note:** This process is illustrated in Figure 8-15 on page 403.

## 8.14 Infoprint Server ISPF e-mail protocol panel

- ☐ Specify addresses of the e-mail recipients
  - Can specify real e-mail addresses or alias names that are defined to sendmail
  - An alias name represents one or more e-mail addresses
  - Define this field in a printer definition with e-mail protocol
    - To extend the length of this field, place your cursor on the word "extend" and press Enter.

```
Add Command ==> _____ E-mail Protocol
Printer definition name . EMAIL10 _____
E-mail addresses . . paulroge@us.ibm.com,rconway@us.ibm.com,haimo@us. (extend)
```

Figure 8-14 IP PrintWay protocol panel for an e-mail printer definition

### Infoprint Server ISPF e-mail protocol panel

In an IP PrintWay printer definition, you can select the transmission protocol that IP PrintWay uses to transmit output data sets from the JES spool to the target system. The target system can be a printer, a print server, or an e-mail destination.

Select the IP PrintWay e-mail protocol if you want IP PrintWay to transmit data sets to one or more e-mail addresses over the Internet using the z/OS UNIX sendmail function that z/OS Communications Server provides.

IP PrintWay transmits e-mails to the e-mail addresses, shown in Figure 8-14, that you specify in the printer definition. Some job submitters can override the e-mail addresses during job submission. You can set up just one printer definition for the e-mail protocol because the job submitter can specify the e-mail addresses. You must specify a default e-mail address in this printer definition.

### Batch job JCL

The batch job, shown in Figure 8-12 on page 400, specifies that the output data set should be sent to printer EMAIL10.

The protocol panel, shown in Figure 8-14, defines the e-mail addresses for printer definition EMAIL10.



## 8.15 Flow for batch job output

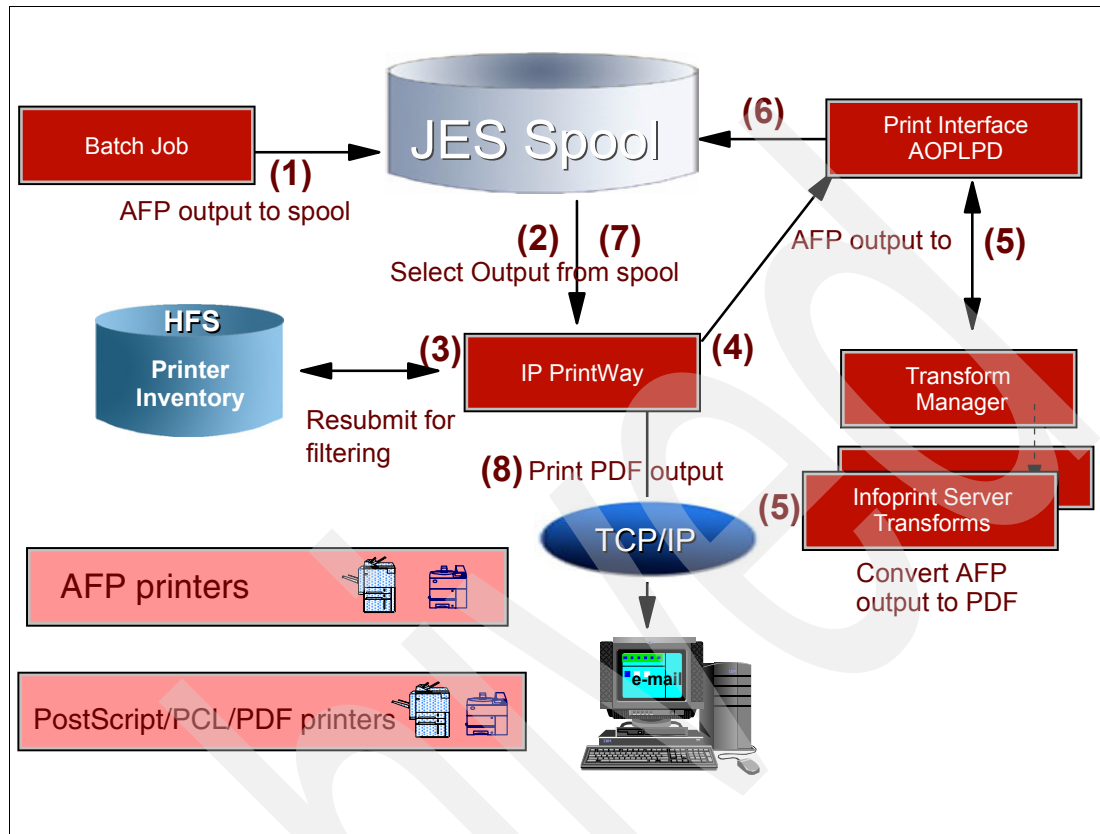


Figure 8-15 Flow of batch job output to the e-mail address

### Batch job output flow

Figure 8-15 describes the flow that takes place when the batch job is submitted and the final destination is an e-mail address, as follow:

1. The batch job is submitted and during its execution it creates an AFP output data set on the JES spool and the job completes execution.
2. The AFP data set on the spool is selected by an IP PrintWay basic mode writer based on its output class and output destination.
3. IP PrintWay basic mode determines from the Printer Inventory that the resubmit for filtering flag is on and for AFP output that a data stream transform should be used to convert the output to a PDF.
4. IP PrintWay basic mode passes the transform request to the Print Interface for processing.
5. Print Interface passes the request to the Transform Manager and the proper transform is selected to transform the data from AFP to PDF.
6. The PDF created is then placed back on the JES spool
7. IP PrintWay is notified to select the PDF output from the JES spool.
8. IP PrintWay transmits the PDF output to the IP addresses specified in the protocol definition for the EMAIL10 printer.

## 8.16 AOPBATCH Program for Data Transforms

### ❑ AOPBATCH program in SYS1.LINKLIB

```
EXEC PGM=AOPBATCH,PARM='/transform_name transform_options'

//AOPBATCH JOB ...
//TRANSFRM EXEC PGM=AOPBATCH,
//    PARM=/ps2afp -o //DD:OUTPUT //DD:INPUT
//INPUT DD DSN=HLQ.INPUT.PS,DISP=SHR
//OUTPUT DD DSN=HLQ.OUTPUT.AFP,DISP=(NEW,CATLG,DELETE),
//    RECFM=VBM,LRECL=32756,BLKSIZE=32760, SPACE=(CYL,(1,1))
//STDOUT DD SYSOUT=*
//STDERR DD SYSOUT=*
//STDENV DD *
PATH=/usr/lpp/Printsrv/bin:/usr/bin
LIBPATH=/usr/lpp/Printsrv/lib:/usr/lib
NLSPATH=/usr/lpp/Printsrv/En_US/%N:/usr/lib/nls/msg/En_US/%N
AOPCONF=/etc/Printsrv/aopd.conf

➤ Transforms: afp2pcl, afp2pdf, afp2ps, pcl2afp, pdf2afp, ps2afp, sap2afp
➤ With XML Extender: xml2afp and xml2pdf
➤ With XT Extender: x2afp
```

Figure 8-16 Transforming data with the AOPBATCH program

### Transforming data with the AOPBATCH program

Infoprint Server provides the AOPBATCH program in SYS1.LINKLIB. The AOPBATCH program lets you submit a batch job to transform data to and from the AFP data format using the Infoprint transform products or another optional transform product, such as IBM Infoprint XML Extender for z/OS (5655-J66) or IBM Infoprint XT Extender for z/OS (5655-J65). XT is the IBM Xerox Transform technology. The AOPBATCH program parameters are the name of the transform command, followed by transform options and arguments, in this format:

```
EXEC PGM=AOPBATCH,PARM='/transform_name transform_options'
```

#### /(slash)

The optional slash indicates that the PARM data that follows is input to AOPBATCH. If you omit the initial slash, your PARM data might be interpreted as C++ run-time options. You must include the initial slash if any of the PARM data itself includes a slash.

#### transform\_name

The name of an executable transform program that resides in an HFS file. The name of the transform program is case-sensitive. You can specify names: afp2pcl, afp2pdf, afp2ps, pcl2afp, pdf2afp, ps2afp, and sap2afp. If you have XML Extender, you can also use xml2afp and xml2pdf. If you have XT Extender, you can also use x2afp.

If the transform program does not reside in one of the directories specified in the PATH environment variable, also specify the path name. You can use the STDENV DD statement to set the PATH environment variable if the default value set by AOPBATCH is not suitable.

## transform\_options

Options and arguments accepted by transform. You must specify the transform input data set or file as a transform argument, and you must specify the -o transform option to identify where you want the transform to write its output. (This is because the transform cannot write its output to standard output and cannot read input from standard input.) To identify the transform input and output data set or file, you can specify either a DD statement name or a data set or file name. You must specify a DD statement name if you want to write the transform output to an MVS data set that does not already exist.

Specify the names of DD statements to the transform in the format:

### //DD:DDname

The name of the DD statement is DDname.

When you specify an MVS data set name in the -o options, code two slashes before the data set name and enclose the data set name in two sets of single quotation marks if you specify a fully qualified data set name. If you do not enclose the data set name in quotation marks, C/C++ assumes that the data set name is not fully qualified and adds a high-level qualifier to the name you specify:

- If you are running under TSO (batch or interactive), z/OS C/C++ appends the TSO user prefix.
- If you are running under MVS batch or IMS (batch or online), z/OS C/C++ appends the RACF user ID.
- If your system does not use RACF, C/C++ does not add a high-level qualifier.

The AOPBATCH program uses the following standard DD statements:

### STDENV

Specifies environment variables for use by the transform. You can specify the environment variables in-stream in the JCL, in an MVS data set, or in a UNIX file. Specify the environment variables in the format `variable=value`, with one environment variable per line or record. Sequence numbers in columns 73 - 80 in data specified with the STDENV DD statement are ignored and not treated as part of the data.

If you omit the STDENV DD statement or do not specify one of the environment variables, AOPBATCH sets these default values, which are suitable for running Infoprint Server programs if your installation installed Infoprint Server files in the default directories:

```
PATH=/usr/lpp/Printsrv/bin:/bin:/usr/bin
```

```
LIBPATH=/usr/lpp/Printsrv/lib:/lib:/usr/lib
```

```
NLSPATH=/usr/lpp/Printsrv/%L/%N:/usr/lpp/Printsrv/En_US/%N:/usr/lib/nls/msg/%L/%N
```

AOPBATCH also sets the HOME environment variable to the user's home directory and sets the LOGIN variable to the user ID.

**Note:** Do not specify the `_BPX_SHAREAS` environment variable. AOPBATCH will set it appropriately.

### STDERR

Specifies the system output data set where error messages are to be written. The data set can be an MVS data set or a UNIX file.

### STDOUT

Specifies the system output data set where informational messages are to be written. The data set can be an MVS data set or a UNIX file.

## 8.17 AOPPRINT JCL procedure

```
//VAINIA JOB (999,POK),EXPERT,MSGLEVEL=1,MSGCLASS=T
//EMAIL EXEC AOPPRINT,PRINTER='Etst',
//  OPTIONS='attributes=//DD:MAILADDR'
//STDENV DD DSN=PRINTWAY.ENV.VARS,DISP=SHR
//SYSIN DD PATH='/tmp/lp.hlp',PATHOPTS=ORDONLY
//MAILADDR DD *
mail-to-addresses={'vaini@fi.ibm.com'}
mail-cc-addresses={'rogers@us.ibm.com'}
mail-bcc-addresses={'boss@us.ibm.com'}
mail-from-name="Me"
mail-reply-address='vaini@fi.ibm.com'
mail-file-name="lp man"
title-text="Ditto"
/*

//JOB2 JOB .....
//PRINTPS EXEC AOPPRINT,PRINTER='POKE'
//SYSIN DD DSN=FILE1.LISTPS,DISP=SHR
```

Figure 8-17 Sample JCL for AOPPRINT

### Sample JCL for AOPPRINT

With the AOPPRINT procedure, you can send the file to an e-mail destination instead of to a printer. You can specify the e-mail addresses of the primary and secondary recipients in one of these ways:

- ▶ An administrator can specify the e-mail address list of the recipients in the printer definition in the Printer Inventory.
- ▶ You can specify the e-mail addresses of the recipients using the job attributes shown in Figure 8-17 and described in “Sending output to an e-mail destination” on page 399.

The example in Figure 8-17 shows how to send a file to an e-mail destination. Assumptions:

- ▶ The administrator has set up a printer definition Etst in the Printer Inventory.
- ▶ The e-mail protocol is selected and an e-mail address is specified in the printer definition mail for the primary recipient. However, the e-mail address you specify in the mail-to-addresses job attribute overrides the e-mail address in the printer definition.
- ▶ The AFP to PDF transform is specified in printer definition Etst.
- ▶ AFP resources libraries are specified either in the printer definition or in the transform configuration file, aopxfd.conf. If the AFP resource libraries are not specified elsewhere, specify the libraries in the resource-library job attribute.

The STDENV DD statement specifies an MVS data set that contains environment variables to be used by the AOPPRINT program.

See also “AOPPRINT JCL procedure” on page 160.

## 8.18 Creating an alias file

- ❑ Create a file using the name defined by your administrator, for example create file
  - **OUTPUT** statement: **MAILTO=branches**
- ❑ **/etc/aliases** file:
  - **branches**: "include:/u/myuserid/branches.list"
- ❑ A user creates the e-mail addresses in the file, **branches.list** by using for example, **oedit**.
  - **oedit /u/myuserid/branches.list**
    - **branch01 @xyz.com,branch02 @xyz.com,**
    - **...**
    - **branch56 @xyz.com,branch57 @xyz.com**

Figure 8-18 Creating alias files

### Alias files

An administrator can specify one or more alias names in the printer definition, and you can specify alias names in some JCL parameters. An alias name is a name defined to z/OS UNIX sendmail that represents one or more actual e-mail addresses. For example, alias name *branches* might represent the e-mail addresses of all employees in *branch 02*.

The actual e-mail address list can be defined in any UNIX file. If the file is one that you can edit, the administrator does not need to change the printer definition whenever you need to change the address list and you do not need to change your JCL. For example, your administrator can specify that the actual e-mail address list for alias *dept123* is in file */u/myuserid/branches.list*.

### Creating alias files

For each address list, your administrator must do these tasks:

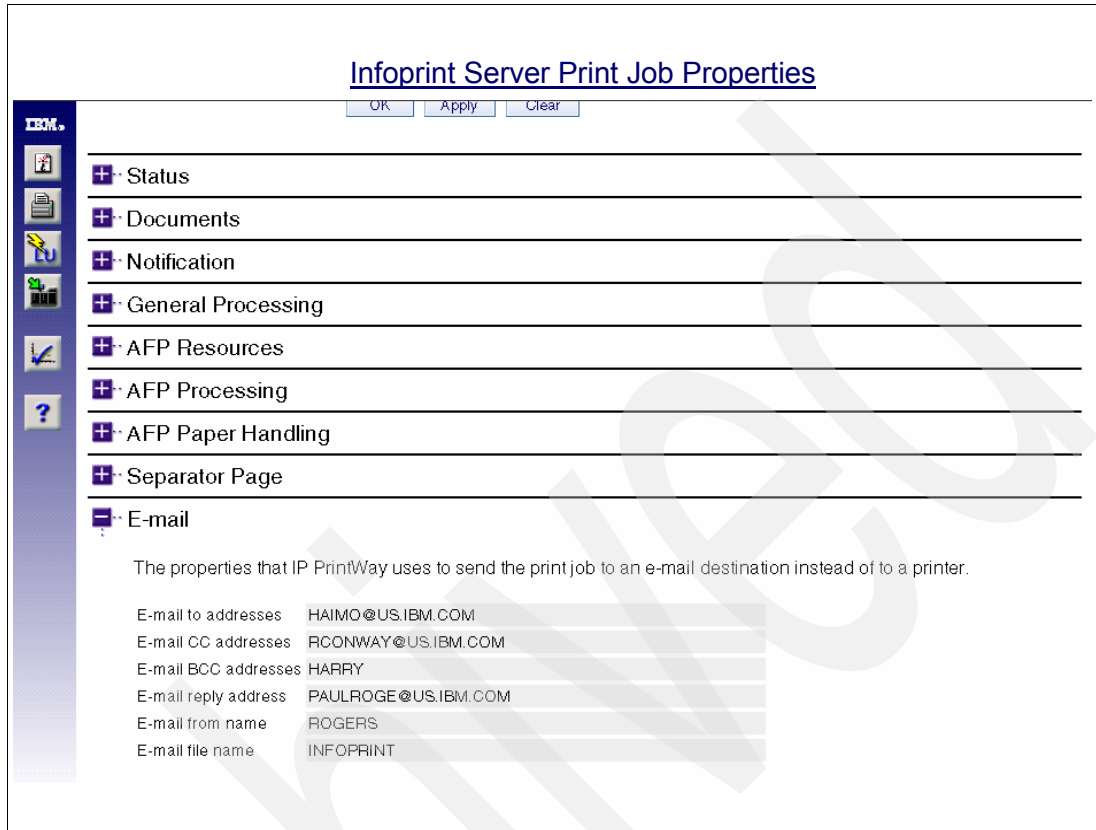
- ▶ Create a printer definition and specify a sendmail alias name in the definition.
- ▶ In the sendmail aliases file */etc/aliases*, specify the same alias name and specify the name of a file to contain the actual e-mail address list. Ask your administrator for the name of this file so that you can create it.

### User creates

A user creates the e-mail addresses in the file *branches.list* by using, for example, **oedit**.

```
oedit /u/myuserid/branches.list
```

## 8.19 Infoprint Central e-mail display



**Infoprint Server Print Job Properties**

OK Apply Clear

IBM

- Status
- Documents
- Notification
- General Processing
- AFP Resources
- AFP Processing
- AFP Paper Handling
- Separator Page
- E-mail**

The properties that IP PrintWay uses to send the print job to an e-mail destination instead of to a printer.

E-mail to addresses: HAIMO@US.IBM.COM  
E-mail CC addresses: RCONWAY@US.IBM.COM  
E-mail BCC addresses: HARRY  
E-mail reply address: PAULROGE@US.IBM.COM  
E-mail from name: ROGERS  
E-mail file name: INFOPRINT

Figure 8-19 Infoprint Server Print Job Properties display of e-mail destinations from a JES batch job

### JES print job properties

This panel displays all the properties of a JES print job. The E-mail section of the panel displays the properties that IP PrintWay will use to send the output data set to the e-mail destinations that were specified in the batch job. The batch job JCL is as follows:

```
//ROGERSW JOB (POK,999),MSGCLASS=T,NOTIFY=ROGERS
//PW OUTPUT MAILTO=HAIMO@US.IBM.COM,
// MAILFROM='ROGERS',
// REPLYTO='PAULROGE@US.IBM.COM',
// MAILFILE=INFOPRINT,
// MAILCC='RCONWAY@US.IBM.COM',
// MAILBCC=HARRY
//PRINT EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=J,OUTPUT=(*.PW)
//SYSUT2 DD SYSOUT=F,OUTPUT=(*.PW)
//SYSUT1 DD DISP=SHR,DSN=ROGERS.JCL.VERS5(GENER)
//SYSIN DD DUMMY
```

The properties that IP PrintWay uses to send the print job to an e-mail destination instead of to a printer may be displayed as hexadecimal representation when invalid characters entered in the e-mail to addresses. For example:

E-mail to addresses x'A5818995896A86894B8982944B839694'  
c'vaini|fi.ibm.com' = x'A5818995896A86894B8982944B839694'

The '@' character is erroneously typed as '|' in the e-mail address.

## 8.20 Printing from workstations overview

- ❑ With Infoprint Server, you can print from your Windows system to any printer defined in the Infoprint Server Printer Inventory
- ❑ Methods to submit a print request:
  - Any Windows application that supports printing, such as a word processor or a viewer application
    - IBM Infoprint Port Monitor for Windows
    - Server Message Block (SMB) protocol
    - Internet Printing (IPP)
  - LPR command
    - TSO/E - z/VM - Windows 2000/XP - OS/400 - AIX

Figure 8-20 Workstation printing using Infoprint Server

### Workstation printing

With Infoprint Server, you can print from your Windows system to any printer defined in the Infoprint Server Printer Inventory.

Any Windows application that supports printing, such as a word processor or a viewer application. To print from a Windows application, you can use the following:

- ▶ IBM Infoprint Port Monitor for Windows
- ▶ The Server Message Block (SMB) protocol
- ▶ The Internet Printing Protocol (IPP)

### Windows print command

Some Windows systems provide a **print** command, which you can use to print on a printer you defined in your Windows system. To use the **print** command, you or your Windows administrator must define the z/OS printer as a Windows shared printer. The **print** command lets you print a file directly without using a Windows application that supports printing. You can specify this command and these options when you print using Infoprint Server:

```
print /d:\\servername\print_share file
```

### Windows and the lpr command

Some Windows systems provide an LPR command, which you can use to print on any printer defined to Infoprint Server in the Printer Inventory. The **LPR** command lets you print a file directly without using a Windows application that supports printing. To use the LPR

command, the administrator must have configured the Infoprint Server LPD to listen at port 515. If the Infoprint Server LPD listens at a different port, use the **print** command instead. To use the **print** command, you must configure the Infoprint Port Monitor to submit print requests to the port at which the Infoprint Server LPD is listening.

### **lprafp command**

The **lprafp** command lets you submit print requests from Windows systems and UNIX systems (such as HP-UX and SunOS) and specify Infoprint Server job attributes. For example, when you use the **lprafp** command, you can specify the name of a form definition in the form-definition attribute and the number of copies in the copies attribute.



## 8.21 Adding a z/OS printer to a Windows environment(1 of 5)

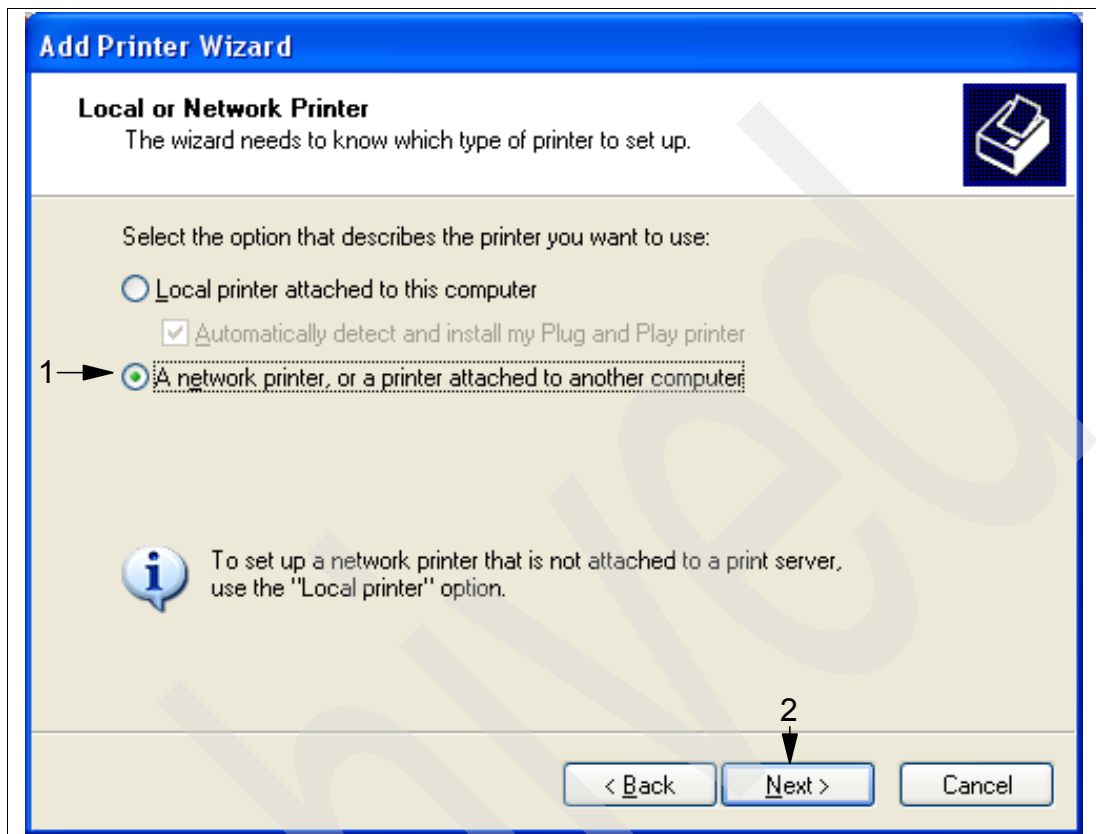


Figure 8-21 Adding a z/OS printer to a Windows environment - 1

### Adding a z/OS printer to a Windows environment (IPP protocol)

When you add a network printer you need to know:

- ▶ The URI of a printer defined in the Printer Inventory has this format:

`http://host:port/servlet/IPPServlet/printername`

where:

host            The host name or IP address of the z/OS system.

port            The port number where the IPP server is listening. The default is that the IPP server listens at port 631. Ask your administrator the port number to use.

printername    The name of the printer definition in the Printer Inventory.

Be careful to note the *exact* spelling of the URI. Uppercase and lowercase letters are not equivalent.

- ▶ The manufacturer and model number of the printer.
- ▶ The name of the printer driver to use.

The printer driver converts your documents to a format that the printer understands. This driver might be available online, or your administrator might give it to you on a diskette or CD-ROM.

## AFP printer driver

The AFP printer driver converts your documents to AFP format, which lets you print on IBM AFP printers. You can download the AFP Printer Driver at no charge from the Web at:

<http://www.ibm.com/printers/download.html>

If you want to use the AFP driver, you need to know the model number and resolution of the AFP printer.

If either the PCL to AFP transform or the PostScript to AFP transform is installed on the z/OS system, you can use a PCL or PostScript driver instead of the AFP driver.

## Add Printer Wizard

When the Add Printer Wizard asks you whether to install the printer as a local printer or a network printer:

1. Select **Network printer**, as shown in Figure 8-21.

On Windows XP, select **A network printer, or a printer attached to another computer**.

2. Click **Next**.

**Note:** The panels shown in Figure 8-21 and Figure 8-22 are from Windows XP.

## 8.22 Adding a z/OS printer to a Windows environment (2 of 5)

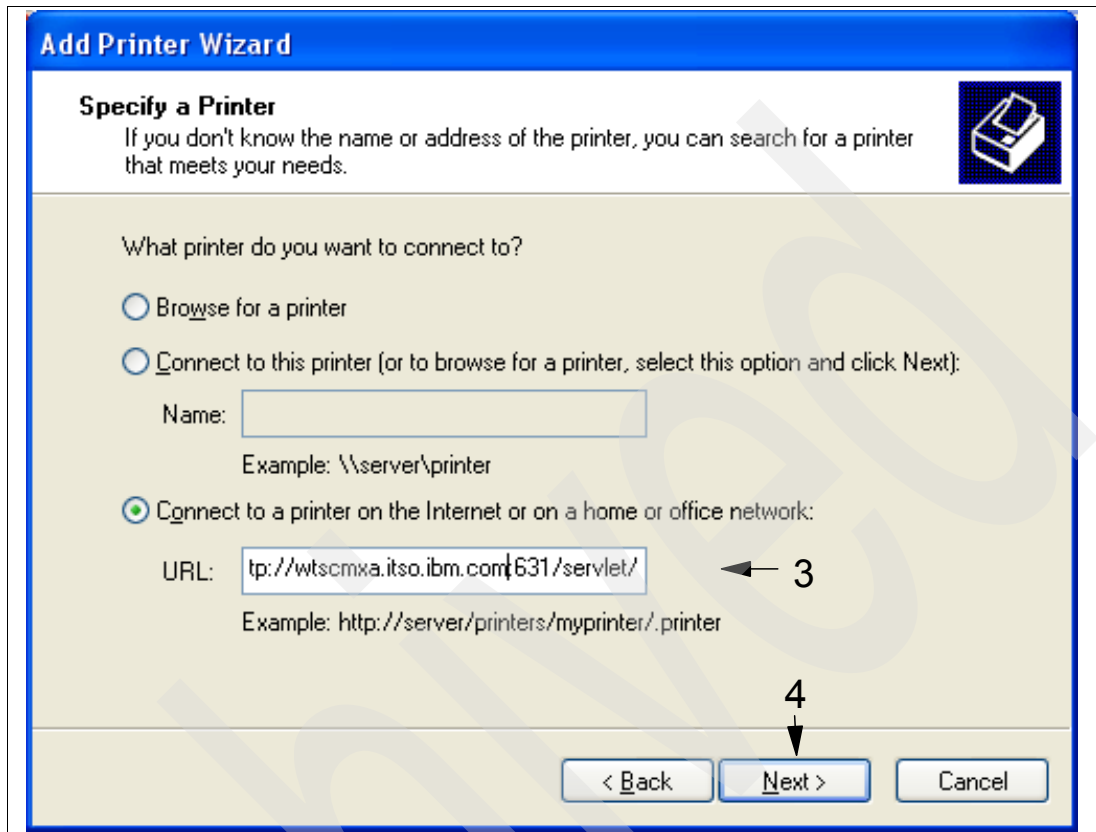


Figure 8-22 Adding a z/OS printer to a Windows environment - 2

### Adding a z/OS printer to a Windows environment (2 of 5)

Use the standard Windows procedure to add a printer, and to continue defining the connection to the host for an IPP connection with these special steps:

3. When the Add Printer Wizard asks you how to connect, select **Connect to a printer on the Internet**. Enter the URI of the printer in the URL field. The URI entered is:

`http://wtscmx.itso.ibm.com:631/servlet/IPPServlet/T0IAZE`

where:

<code>wtscmx.itso.ibm.com</code>	The host IP address which can either be an IP address or a host name
<code>631</code>	The IPP port number
<code>servlet/IPPServlet</code>	Required syntax
<code>T0IAZE</code>	The printer name in the Printer Inventory

4. After typing in the URI, click **Next**.

## 8.23 Adding a z/OS printer to a Windows environment (3 of 5)

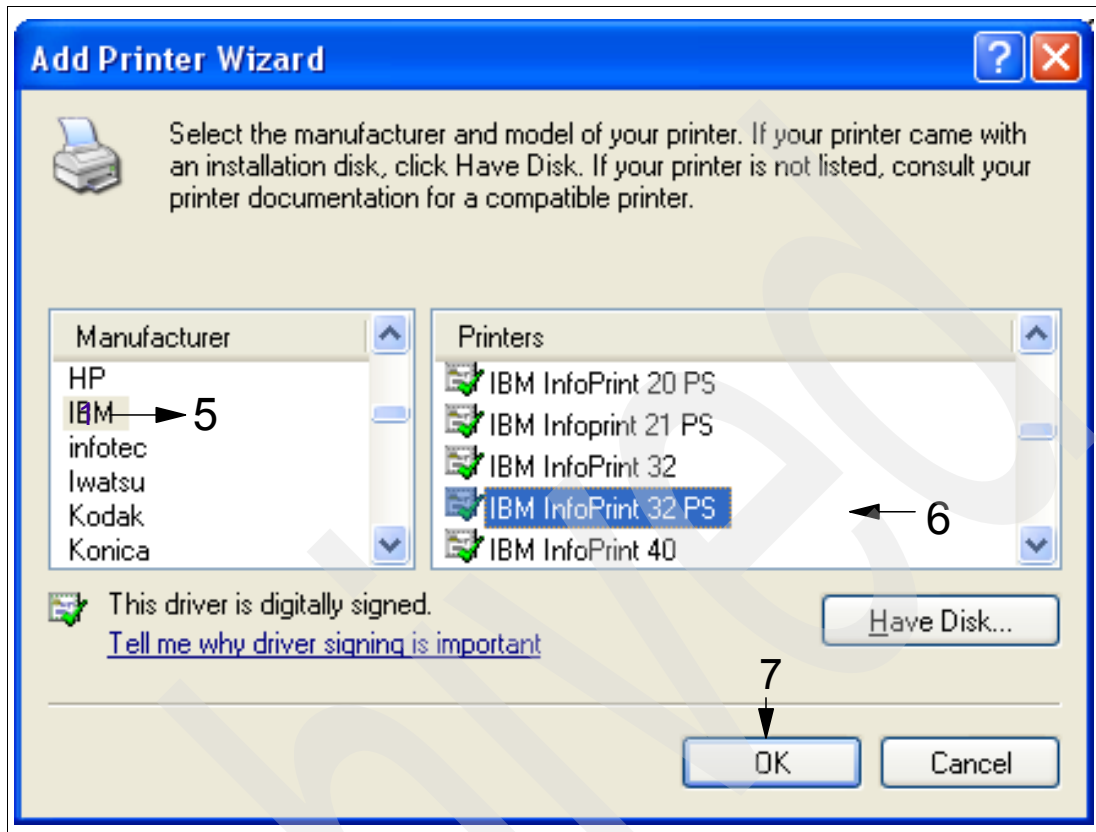


Figure 8-23 Adding a z/OS printer to a Windows environment - 3

### Adding a z/OS printer to a Windows environment (3 of 5)

Specify additional information about the printer that was identified on the previous panel:

5. Select the printer manufacturer.
6. Select the printer type.
7. When the Add Printer Wizard prompts you whether to install the printer driver, click **OK**.

While Windows is installing the printer driver, the Add Printer Wizard returns to the panel where you entered the URI. For some time, it might seem that nothing is happening. Do not cancel the job. Wait for the Add Printer Wizard to go on to the next panel.

## 8.24 Adding a z/OS printer to a Windows environment (4 of 5)



Figure 8-24 Adding a z/OS printer to a Windows environment - 4

### Adding a z/OS printer to a Windows environment (4 of 5)

This panel allows you to specify whether you would like to make this the default printer for your session.

8. Select either **Yes** or **No** for the default printer.
9. Click **Next**.

## 8.25 Adding a z/OS printer to a Windows environment (5 of 5)

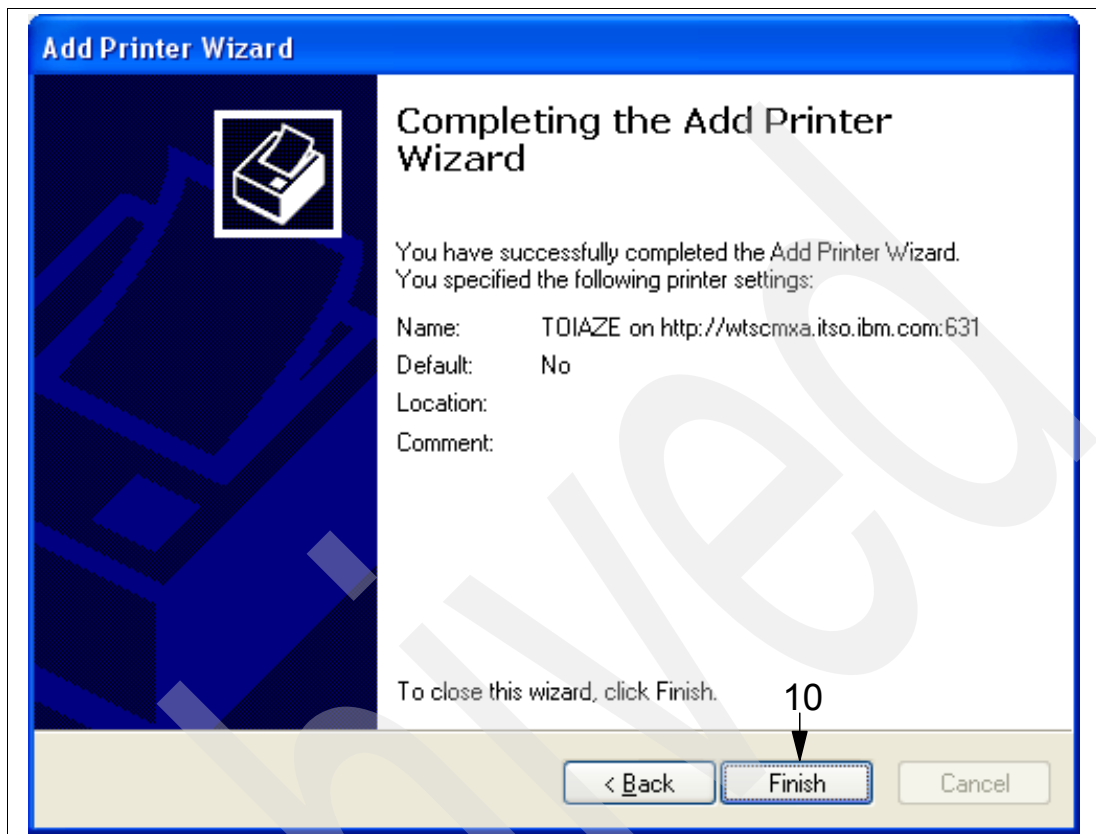


Figure 8-25 Adding a z/OS printer to a Windows environment - 5

### Adding a z/OS printer to a Windows environment (5 of 5)

On the confirmation panel:

10. Click **Finish**.

Now that the printer driver is loaded, the printer being defined is ready for use.

## 8.26 Printing from Windows with the z/OS printer

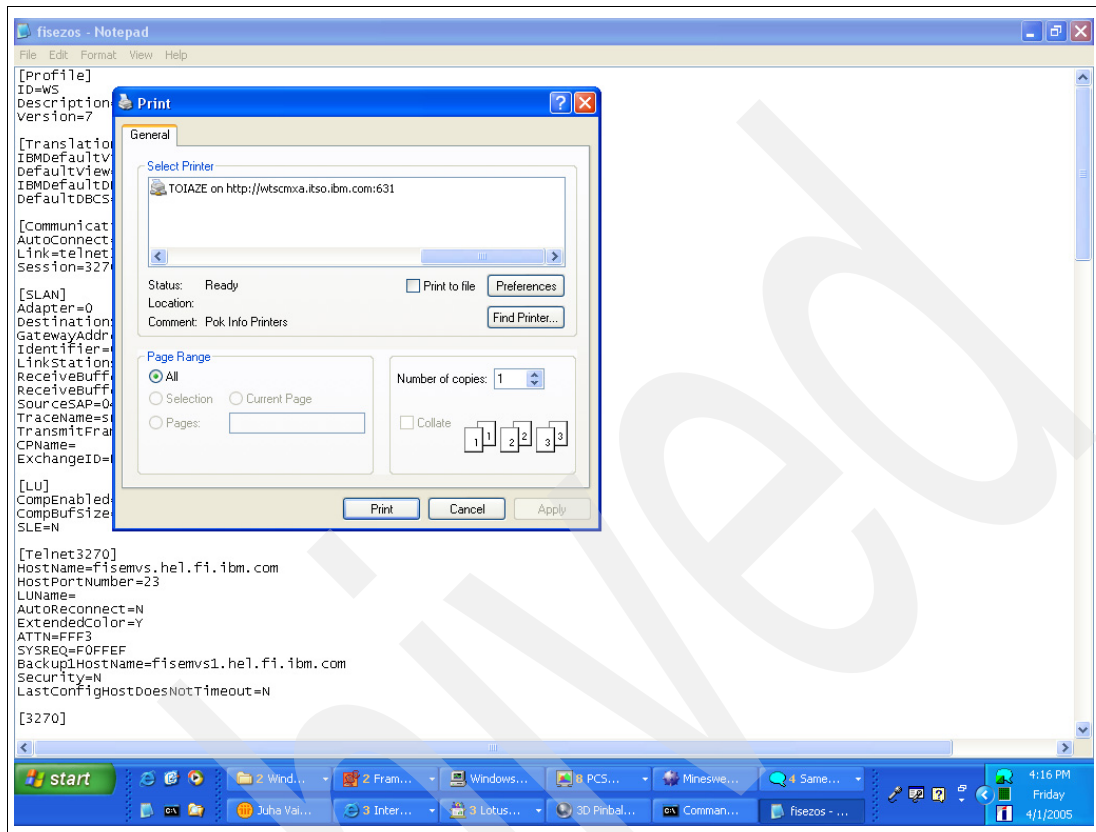


Figure 8-26 Printing from Windows with the z/OS printer

### Printing from Windows with the z/OS printer

Now that the printer is defined in the Add Wizard printers, you just need to select it from a list of printers to send a document to the host to Infoprint Server.

The example in Figure 8-26 shows how to print a Notepad document to the z/OS printer.

1. On the **File** menu click **Print**.
2. On the **General** tab select the printer and options you want and click **Print**.

To change the look of your printed document, on the **File** menu click **Page Setup**.

## 8.27 Using LPR command to submit print requests

- ❑ The LPR command is available on most platforms
- ❑ Examples:
  - **z/OS syntax**
    - LPR dsname ( H host P printer other\_parameters
  - **z/VM syntax**
    - LPR fn fm ft (H host P printer other\_parameters
  - **Window syntax**
    - lpr -S server -P printer [-C class] [-J job] [-o option] [-x] [-d] filename
  - **OS/400 syntax**
    - LPR RMTSYS(host) PRTQ('printqueue') FILE(filename) other\_...
  - **AIX syntax**
    - lpr -P remotequeue file
    - To print, you must also configure a remote queue

Figure 8-27 Using LPR command to submit print requests

### Using LPR command to submit print requests

You can use the LPR command to submit a print request. From an AIX system, you can also use the **enq** command.

To use an LPR or **enq** command you need to know:

- ▶ The host name or IP address of the z/OS system on which Infoprint Server is running.
- ▶ The name of the printer definition created for the printer. This name is case-sensitive. You must specify this name as the name of the printer or print queue.

The OS/400® LPR command and the AIX **enq** command let you specify Infoprint Server job attributes.

After receiving a print request, Infoprint Server returns either an error message or a job identifier. The job identifier indicates that Infoprint Server has accepted the print request. You can use the job identifier to query the status of the print request or to cancel the print request.

The LPQ command allows you to request a list of the printer queue on a remote printer from the LPD server controlling that printer.



## 8.28 z/OS LPR command syntax

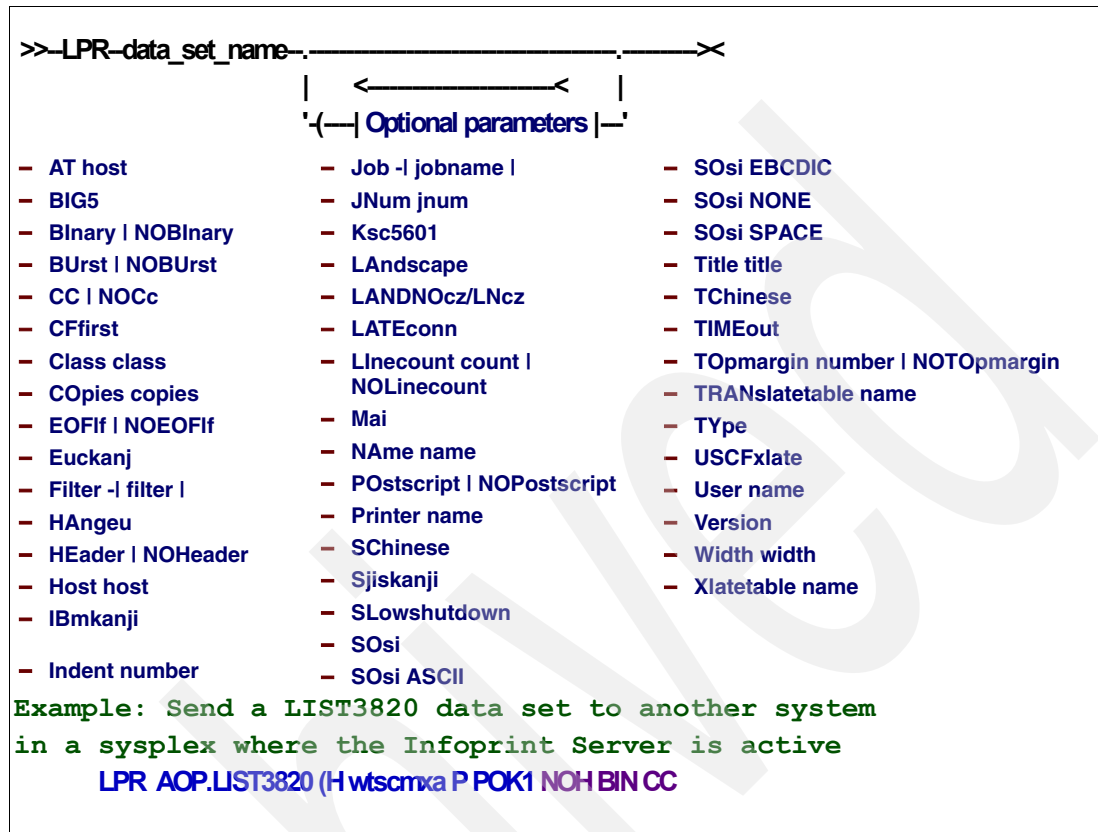


Figure 8-28 z/OS LPR command syntax

### z/OS LPR command syntax

For a description of the optional parameters of the LPR TSO/E command, enter the command `HELP LPR` in your TSO session.

**Note:** The MVS LPD server always converts data sets in ASCII to EBCDIC, and there is no option to disable this conversion.

In the example in Figure 8-28, the options `NOHeader`, `BInary`, and `CC` were used.

- ▶ The `BINARY` option causes LPR to send the data without translation and without any indication of record boundaries.
- ▶ The `NOHEADER` option prevents the client from inserting page headers.
- ▶ The `CC` option causes the remote system to interpret the first character of each line as carriage control. This option is the default if the record format is `FA`, `FBA`, `FBM`, `FM`, `VA`, `VBA`, `VBM`, or `VM`. The characters used to specify these record formats have the following meanings:

- |   |  |
|---|--|
| F | Fixed record length                                |
| V | Variable record length                             |
| B | Blocked records                                    |
| A | Records containing ISO/ANSI control characters     |
| M | Records containing machine code control characters |

### **LIST3820 data sets**

The LIST3820 data sets have record format VBM, so the CC option was not really required. However, the NOH and BIN options are required. IP Printway needs to understand the format of the print data in order to process it correctly. The NOH option prevents the LPR command from inserting data set header text and the BIN option preserves the data format (the X'5A' M character), thus allowing IP Printway to properly determine the format of the data.

## 8.29 SDSF view of IP PrintWay data sets

- ❑ Considerations when using SDSF
- ❑ Default job-prefix in configuration file - PS
  - Jobname, JobID, and Owner fields
  - Printer DEST and SYSOUT class
  - Batch jobs' print data sets do not get a PS job-prefix
- ❑ SDSF can show all print data sets currently on JES spool
- ❑ But SDSF does not show IP PrintWay processing status

*Figure 8-29 SDSF and IP PrintWay data sets on JES spool*

### **SDSF view of IP PrintWay data sets**

The SFSF Output Queue panel allows the user to display information about SYSOUT data sets for jobs, started tasks, and TSO users on any non held JES2 output queue.

The SDSF Held Output panel shows the user information about SYSOUT data sets for jobs, started tasks, and TSO users on any held JES2 output queue.

The HOLD=YES parameter on a SYSOUT DD statement tells the system to hold a SYSOUT data set until it is released by the system operator. SDSF shows these SYSOUT data sets on the Held Output panel, not on the Output Queue panel.

SYSOUT data sets that are created by batch jobs and directed to IP PrintWay do not get a PS job-prefix. SDSF does not show these data sets on the Output Queue panel when "JobID EQ PS\*" filtering is used to limit data until IP Printway selects the data sets.

When Print Interface creates an output data set on the JES spool, it assigns the job-prefix for each document to be printed.

### 8.30 JES2 filter to display PS output

[illegible]

Figure 8-30 JES2 filter to Display PS output

## Locating output data sets

The Infoprint Server job ID starts with the 2-character prefix that is defined in the Infoprint Server configuration file, `aopd.conf`. The default prefix is PS. The job submitter or VTAM application can specify a job ID in the `sysout-job-id` job attribute. If specified, this job ID is used instead. The `1p` command returns this job ID to the job submitter.

SDSF does not display jobs with a job ID prefix of PS. Therefore, it is necessary to define a filter to view output data sets with a job ID prefix of PS, as shown in Figure 8-30.

## 8.31 Viewing Infoprint Server messages

- ❑ **Infoprint Server (common message log)**
  - Messages from most Infoprint Server components
  - **aoplogu** command
- ❑ **IP PrintWay basic mode**
  - IP PrintWay basic mode messages are in IP PrintWay message-log
  - **Infoprint Server ISPF session - Option 6**
- ❑ **NetSpool**
  - NetSpool sends all messages to the operator console
  - Common message log
  - NetSpool message log data set (if specified in NetSpool JCL)
- ❑ **Data transforms**
  - Each transform daemon instance writes messages to its own log
  - **/var/Printsrv/xfd/** subdirectory
- ❑ **z/OS UNIX sendmail**
  - e-mail for a local system - messages recorded in the IP PrintWay message log (basic mode) or the common message log (extended mode).
  - e-mail for a remote system - no error messages recorded locally

Figure 8-31 Viewing Infoprint Server messages

### Viewing Infoprint Server messages

Infoprint Server writes messages for:

- ▶ Infoprint Server (common message log) can be viewed using the **aoplogu** command.
- ▶ IP PrintWay basic mode places messages in the IP PrintWay messages log.

In addition, messages are maintained for the following components.

### NetSpool

NetSpool sends messages to several locations:

- ▶ NetSpool sends all messages to the operator console.
- ▶ NetSpool sends most messages, with the exception of messages written at initialization and termination, to the Infoprint Server common message log if it is enabled.
- ▶ NetSpool sends some messages, with the exception of messages written at initialization and termination and job-related messages, to the NetSpool message log data set if it is specified in the NetSpool startup procedure. You can view messages in this data set by browsing it.

### Data transforms

The transform daemons (such as those provided by Infoprint Server Transforms and Infoprint XML Extender for z/OS) send messages to message logs located in the xfd subdirectory in

the base directory defined in the Infoprint Server configuration file `aopd.conf`. The default base directory is `/var/Printsrv/`.

Each instance of a transform daemon writes messages to its own message log. The message log is cleared when you restart the Transform Manager.

The file names of the message logs have the format `transform.n.stderr`, where:

- |           |   |
|-----------|---|
| transform | The name of the transform with the error. The transform name includes the name of the transform class if one is defined. For example: <code>pcl2afp</code> or <code>pcl2afp_letter_300</code> . |
| n         | The instance of the transform daemon.   |

## **z/OS UNIX sendmail**

When IP PrintWay transmits output to an e-mail destination, it uses the z/OS UNIX sendmail facility. Sendmail returns an error message when it cannot deliver an e-mail:

- ▶ If the target e-mail address is for a local system, the sendmail error message is recorded in the IP PrintWay message log (basic mode) or the common message log (extended mode).
- ▶ If the target e-mail address is for a remote system, no error message is recorded in the IP PrintWay message log (basic mode) or common message log (extended mode). Sendmail instead sends a message about the failed delivery to one of these users:
  - If the e-mail address was specified in a sendmail alias, sendmail notifies the owner of the alias.
  - If the e-mail address was specified directly in the printer definition, or if no owner is specified for the alias, sendmail notifies the user associated with the IP PrintWay startup procedure. This user ID is AOPSTC if your installation used the user ID suggested in z/OS Infoprint Server Customization.

You might need to wait several days before sendmail returns an error message to you about an e-mail that could not be delivered to a remote system. How long you need to wait depends in part on how long it takes the remote system to notify sendmail that an e-mail is not deliverable and in part on how your installation has customized sendmail. For more information about customizing sendmail timeout values, see z/OS Infoprint Server Customization.

To receive messages from sendmail, run the z/OS UNIX `mail` or `mailx` command.

## 8.32 Viewing messages with aoplogu command

### ❑ **aoplogu** z/OS UNIX command

**aoplogu** [-b time] [-e time] or **aoplogu** -l time

The format of time is:

year:month:day:hour:minute

month:day:hour:minute

day:hour:minute

hour:minute

minute

### ❑ **Environment variables**

- **AOPCONF** - Default /etc/Printsrv/aopd.conf
- **LANG** - The language used for messages
- **LC\_ALL** - Locale for message time and date format
- **LC\_TIME** - Locale for message time and date format
- **NLSPATH** - Directory for Infoprint Server message catalogs
- **TZ** - Time zone displayed in common log messages.

Figure 8-32 Viewing Infoprint Server common message log

### **Viewing Infoprint Server common message log**

The common message log lets you see messages from most Infoprint Server components in one place. The log contains messages from all components of Infoprint Server except for IP PrintWay basic mode. It does not contain messages from Infoprint Server Transforms or other transform products. IP PrintWay extended mode writes its messages only to the common message log. Other components, such as NetSpool and Print Interface, write their messages to other locations such as the NetSpool message-log data set and the system console log, as well as to the common message log.

Your administrator must customize the common message log to specify how many days worth of messages to keep. The default is that no messages are kept in the common message log.

Infoprint Central lets authorized users view messages in the common message log for selected print jobs and IP PrintWay extended mode printers. In addition, Infoprint Server administrators can use the **aoplogu** command to select messages in a particular time range and copy them to a file or view them on the terminal.

### **aoplogu** command

The **aoplogu** command lets administrators view messages in the common message log. You can specify that you want to view:

- All the messages.

- The messages in a range of times. For example, the messages from 6 May 2004 to 10 May 2004, or the messages beginning on 6 May 2004.
- The most recent messages for a length of time. For example, the messages for the last day or for the last two minutes.

### **aoplogu command format**

The results are written to standard output (stdout).

Syntax:

```
aoplogu [-b time] [-e time]
```

or

```
aoplogu -l time
```

The format of time is:

```
year:month:day:hour:minute | month:day:hour:minute | day:hour:minute |
hour:minute
minute
```

Options are as follows:

- b** The beginning time of the range of messages that you want to view. The default is the oldest message that is available in the common message log.
- e** The ending time of the range of messages that you want to view. The default is the most recent message that is available in the common message log.
- l** The most recent messages for the length of time that you specify.

To use the **aoplogu** command, you must be defined to RACF as a z/OS UNIX user and be connected to the RACF AOPADMIN group. When you specify the **-b** or **-e** option, the defaults for the year, month, day, and hour are the current year, month, day, and hour. You can enter any value for the units of time, beginning with 0. The time is inclusive. For example, **aoplogu -b 0 -e 0** includes all messages that occurred in the first minute of the current hour.



## 8.33 Format of common message log messages

### Common message log message format

date time priority:level user:id [job: id] [dsn:name] [filename:name]  
[output\_device:name] [job\_selection\_rule:name] [program:name]  
msg:message

### Example

04/05/05 10:06:16 (UTC-5) priority:info user:VAINI job:JOB06180  
dsn:VAINI.VAINIPJ.JOB06180.D0000102.? program:aopoutd  
msg:AOP3401I IP PrintWay deleted the document from the JES spool.

Figure 8-33 Format of common message log messages

### Format of common message log messages

The **aoplogu** command displays messages in the common message log in this format:

date time priority:level user:id [job: id] [dsn:name] [filename:name]  
[output\_device:name] [job\_selection\_rule:name] [program:name] msg:message  
**date** The date the message was written.  
**dsn** The fully qualified data set name that JES assigns.  
**filename** The name of the file to be printed.  
**job** The job ID.  
**job\_selection\_rule** The name of an IP PrintWay job selection rule  
**output\_device** An identifier of the IP PrintWay extended mode printer  
**priority** The severity of the message:  
    **action** - A terminating situation occurred.  
    **error** - An error occurred.  
    **info** - This is an information message.  
    **severe** - A severe error occurred.  
    **warning** - A warning situation occurred.  
**program** The name of the Infoprint Server daemon or command  
**user** The z/OS user ID or the name of the user who submitted the print

## 8.34 Viewing IP PrintWay basic mode messages

```
A0PIPP1                      Infoprint Server: Printer Inventory Manager
Option ==> 6.

Printer Definitions
  1 Add                      Add a printer definition
  2 List                     List printer definitions
  3 Select                   Select printer definitions to list

Other Functions
  4 Other Definitions       Manage FSS, FSA, pool, and job selection definitions
  5 PrintWay Queue         View IP PrintWay basic mode transmission queue
  6 PrintWay Message       View IP PrintWay basic mode message log
  7 Configure              Change panel configuration

006927 2006050205523347-ANFM603I Data set:
006928                               WTSCPLX4.VAINI.VAINI1.JOB03758.D0000009.LIST#AA#
006929                               The data set is being released with a JES checkpoint
006930 2006050205523354-ANFM699I FSS: PRINTWAY The queue manager has stopped
006931 2006051511473996-ANFM004I Using message table ENU
006932 2006051511475798-ANFM108I The Inventory Server by the name of AOP1 is
no
006933                               started. IP Printway will not start until the
inventory
006934                               is running.
```

Figure 8-34 Viewing IP PrintWay basic mode messages

### Viewing IP PrintWay basic mode messages

IP PrintWay basic mode sends a message to the IP PrintWay message-log data set when it receives a data set from JES, when IP PrintWay successfully or unsuccessfully transmits the data set to the target printer or e-mail destination, and when IP PrintWay releases the data set to JES. Also, IP PrintWay writes other messages to this data set, and IP PrintWay installation exits can write messages to this data set.

The messages wrap around to the beginning when the data set becomes full. The time stamp preceding each message indicates when IP PrintWay wrote the message. IP PrintWay writes a string containing equal signs, =====, at the end of the last message written. An installation can write an IP PrintWay Message exit to suppress unwanted messages from the message-log data set.

To view the messages:

- ▶ Start an Infoprint Server ISPF session.
- ▶ On the main Infoprint Server ISPF panel, select: 7 Configure. Make sure that the Message log field contains the name of the IP PrintWay message log used by the IP PrintWay functional subsystem (FSS). This name must match the data set name specified in the IP PrintWay startup procedure.
- ▶ On the main Infoprint Server ISPF panel, select: 6 PrintWay Message.
- ▶ To find the latest message, press the REFRESH function key on the Infoprint Server ISPF panels, or search for a string of equal signs.

## 8.35 IP PrintWay and the JSPA

- ❑ The Infoprint begin-dataset-exit - (banner-exit)
  - Can extract the name and number of the job that created the data set to be printed from the JSPA pointed by XTPJSPAP in IP Printway basic mode
  - The JSPA control block is especially useful to the Begin Data Set Exit and End Data Set Exit in building header and trailer pages
    - IP PrintWay exit parameter area ( ANFUEXTP ) has field XTPJSPAP - points to JSPA
  - In IP Printway extended mode the address of the JES job separator page data area is not passed to exits in field XTPJSPAP - This field contains zeroes if IP PrintWay extended mode calls this exit

Figure 8-35 Using the JSPA for header and trailer pages

### Using the JSPA

The JES job separator page data area (IAZJSPA), called the JSPA, is a system control block that contains job and data set information. You might want to use information in the JSPA to create separator pages in an exit. IP PrintWay basic mode passes the address of the JSPA to exits in field XTPJSPAP of the ANFUEXTP control block. IP PrintWay extended mode passes the address of a partial JSPA to the exits in field XTPJSPAP because IP PrintWay extended mode cannot access the JES JSPA.

The partial JSPA that IP PrintWay extended mode creates contains information that is available to it. These fields contain valid values:

- ▶ JSPAJBNM
- ▶ JSPAJBID
- ▶ JSPJGRPN
- ▶ JSPJGRPD
- ▶ JSPJRMNO
- ▶ JSPJPNAM
- ▶ JSPJDSNM
- ▶ JSPJSOCL
- ▶ JSPJPRIOR
- ▶ JSPCEUID
- ▶ JSPCEDSN
- ▶ JSPCESEG

### **ANFUEXTP exit**

Your exit can retrieve the z/OS system name from either the JMR control block (basic mode) or the XTP\_ORIGINATING\_MVS\_SYSTEM and XTP\_ORIGINATING\_JES2\_SYSTEM fields in the ANFUEXTP control block (extended mode).

Your exit can check the XTPVERS field to determine whether the JSPA is the full JSPA that IP PrintWay basic mode provides or the partial JSPA that IP PrintWay extended mode provides. IP PrintWay extended mode sets the XTPVERS field to 1. IP PrintWay basic mode sets the XTPVERS field to 0.

# Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

## IBM Redbooks documents

For information about ordering these publications, see “How to get IBM Redbooks publications” on page 432. Note that some of the documents referenced here may be available in softcopy only.

- ▶ *z/OS Infoprint Server Implementation*, SG24-6234
- ▶ *UNIX System Services z/OS Version 1 Release 7 Implementation*, SG24-7035-01

## Other publications

These publications are also relevant as further information sources:

- ▶ *z/OS Infoprint Customization*, S544-5744
- ▶ *z/OS Infoprint Server User's Guide*, S544-5746
- ▶ *z/OS Infoprint Server Operation and Administration*, S544-5745
- ▶ *z/OS UNIX System Services Command Reference*, SA22-7802
- ▶ *z/OS XL C/C++ Programming Guide*, SC09-4765
- ▶ *z/OS Communications Server: IP Configuration Reference*, SC31-8776
- ▶ *z/OS HTTP Server Planning, Installing, and Using*, SC34-4826

## Online resources

These Web sites and URLs are also relevant as further information sources:

- ▶ Printing Systems Division Web site at:  
<http://www.ibm.com/printers/download.html>
- ▶ z/OS LookAt online facility from the Internet at:  
<http://www.ibm.com/eserver/zseries/zos/bkserv/lookat/>
- ▶ The Uniform Resource Identifier (URI) of printers defined in the Printer Inventory has format:  
<http://host:port/servlet/IPPServlet/printername>
- ▶ IBM provides the **lprafp** command on an “as is” basis. You can download the **lprafp** command at:  
[www.ibm.com/printers/downloads.html](http://www.ibm.com/printers/downloads.html)

## How to get IBM Redbooks publications

You can search for, view, or download Redbooks publications, Redpapers, Hints and Tips, draft publications and Additional materials, as well as order hardcopy books or CD-ROMs, at this Web site:

[ibm.com/redbooks](http://ibm.com/redbooks)

## Help from IBM

IBM Support and downloads

[ibm.com/support](http://ibm.com/support)

IBM Global Services

[ibm.com/services](http://ibm.com/services)



## ABCS of z/OS System Programming Volume 7

(0.5" spine)  
0.475" <-> 0.873"  
250 <-> 459 pages









# ABCs of z/OS System Programming

## Volume 7



### Infoprint Server

### IP PrintWay, NetSpool, Infoprint Server Transforms

### Infoprint Central

The ABCs of z/OS System Programming is an eleven volume collection that provides an introduction to the z/OS operating system and the hardware architecture. Whether you are a beginner or an experienced system programmer, the ABCs collection provides the information that you need to start your research into z/OS and related subjects. If you would like to become more familiar with z/OS in your current environment, or if you are evaluating platforms to consolidate your e-business applications, the ABCs collection will serve as a powerful technical tool.

This book describes the functions of the Infoprint Server. It will help you install, tailor, configure, and use the z/OS Version 1 Release 7 version of Infoprint Server. Topics covered in this volume are the following:

- ▶ Infoprint Server
- ▶ UNIX System Services overview
- ▶ Infoprint Server customization
- ▶ Print Interface
- ▶ IP PrintWay
- ▶ NetSpool
- ▶ Infoprint Central
- ▶ User interfaces to Infoprint Server

### INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

### BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

**For more information:**  
[ibm.com/redbooks](http://ibm.com/redbooks)