

z/OS



Infoprint Server Customization

Note:

Before using this information and the product it supports, read the information in "Notices" on page 487.

This edition applies to these IBM products:

- Version 2 Release 1 of z/OS (5650-ZOS)
- Version 2 Release 1 of Infoprint Coaxial Printer Support for z/OS (5655-N62)

This edition applies to all subsequent releases and modifications until otherwise indicated in new editions.

© **Copyright IBM Corporation 1995, 2013.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Figures	ix
--------------------------	-----------

Tables	xi
-------------------------	-----------

About this document **xiii**

Who should read this document	xiii
How to read syntax diagrams	xiii
Where to find more information	xiv
Preventive Service Planning information	xiv
Infoprint Server migration information	xiv
Documents	xv
Infoprint Server online help	xvi

How to send your comments to IBM **xvii**

If you have a technical problem	xvii
---	------

Summary of changes **xix**

z/OS Version 2 Release 1	xix
------------------------------------	-----

Chapter 1. Introducing Infoprint Server **1**

Printer Inventory Manager.	3
Windows client	5
Print Interface	6
Print Interface with an SAP R/3 application server on the z/OS system	9
Print Interface with an SAP R/3 application server on a remote system	11
NetSpool	12
IP PrintWay	15
Transform Interface.	19
Transforms	20
Infoprint Central	21
SNMP subagent	22

Chapter 2. Customization roadmap **25**

Chapter 3. Customizing the Printer Inventory Manager. **27**

Identifying software requirements for Printer Inventory Manager	28
Enabling the Infoprint Server feature	28
Updating the BPXPRMxx member of SYS1.PARMLIB	28
Making Language Environment and C++ runtime libraries available	29
Creating Infoprint Server directories	30
/etc/Printsrv directory	30
/var/Printsrv directory	31
Creating the Infoprint Server configuration file (aopd.conf)	34
Steps for creating the Infoprint Server configuration file (aopd.conf)	41

Steps for editing the Infoprint Server configuration file (aopd.conf)	42
Setting environment variables for the Printer Inventory Manager	43
Setting up security for the Printer Inventory and operator commands	47
AOP.ADMINISTRATOR profile.	48
AOP.CONFIGURATION profile.	49
AOPADMIN and AOPOPER groups	49
AOPCONF group	49
Steps for setting up security for the Printer Inventory, common message log, and operator commands.	50
Setting permissions for directories and executable files -- aopsetup	54
Steps for running aopsetup	55
Customizing startup and shutdown commands and procedures	56
Editing the aopstart EXEC	57
Creating a user ID to assign to the AOPSTART and AOPSTOP procedures	60
Defining the AOPSTART and AOPSTOP procedures to RACF	61
Editing the AOPSTART startup procedure	61
Editing the AOPSTOP shutdown procedure	63
Starting Infoprint Server daemons automatically	65
Enabling Infoprint Server ISPF panels	66
Defining libraries in the TSO logon procedure.	66
Displaying the Infoprint Server panel option	67
Modifying the AOPINIT EXEC	67
Enabling dynamic configuration	68
Dynamic attributes	69
Cross-system coupling facility (XCF)	72
Steps for enabling dynamic configuration	72
Steps for disabling dynamic configuration	74
Steps for determining whether dynamic configuration is enabled	75
Editing the system configuration definition for Printer Inventory Manager	76
Steps for editing the system configuration definition	80
Sending non-console messages to the hardcopy log	84
Creating the message configuration file (aopmsg.conf)	86
Editing the message configuration file (aopmsg.conf)	87
Setting up the MVS system logger for the common message log	88
Deciding whether to use the MVS system logger	89
Deciding the type of log stream	89
Deciding on a naming convention for log streams	90
Determining how long to retain messages	90
Determining the size of log-stream data sets and interim storage	91
Setting up security for log streams.	91
Defining log streams to the MVS system logger	94

Activating a log stream in Infoprint Server	99
Switching from one log stream to another	101
Deactivating a log stream in Infoprint Server	102
Handling system logger errors	103
Viewing messages in the log stream	104
Backing up and restoring the Printer Inventory	105
Steps for backing up the Printer Inventory to an HFS or zFS file	105
Steps for restoring the Printer Inventory from an HFS or zFS file	106
Steps for backing up the Printer Inventory to a generation data group (GDG)	107
Steps for restoring the Printer Inventory from a GDG	108
Saving messages in the common message log in the z/OS UNIX file system	110
Reducing the size of the common message log in the z/OS UNIX file system	111
Setting Workload Manager goals	112

Chapter 4. Customizing Print Interface 115

Identifying software requirements for Print Interface	116
Common customization for Print Interface	116
Customizing z/OS Communications Server	117
Updating the BPXPRMxx member of SYS1.PARMLIB	119
Working with JES spool limits	119
Editing the aopd.conf file for Print Interface	120
Writing a filter program	121
Customizing JES to specify no blank truncation	127
Setting environment variables for Print Interface	127
Editing the system configuration definition for Print Interface	128
Customizing the z/OS UNIX printing commands	130
Customizing the AOPPRINT JCL procedure	130
Customizing the Print Interface LPD	132
Customizing z/OS Communications Server (TCP/IP)	133
Editing the aopd.conf file for the LPD	134
Editing the system configuration definition for the LPD	135
Using the LPD compatibility filter	136
Customizing LPR clients	137
Customizing the z/OS SMB server to use Print Interface	138
Customizing the z/OS SMB server	139
Customizing the Print Interface IPP Server	140
Ensuring sufficient memory is available to start the IPP Server	140
Editing the aopd.conf file for the IPP Server	141
Setting environment variables for the IPP Server	142
Editing the system configuration definition for the IPP Server	142
Customizing the IPP workstation client	143
Customizing SAP R/3	144
Defining an SAP user for the callback daemon	146
Creating the SAP callback daemon configuration file (aopsapd.conf)	146
Setting environment variables for the SAP OMS	148

Defining a ROMS and LOMS to the SAP R/3 application server	148
Modifying the LOMS to specify job attributes	149
Creating an SAP R/3 output device	150
Customizing the Print Interface subsystem	153
Editing the aopd.conf file for the Print Interface subsystem	154
Increasing the maximum number of active transform daemons	154

Chapter 5. Customizing Transform Interface 157

Identifying software requirements for Transform Interface	158
Editing the aopd.conf file for the Transform Manager	158
Creating the transform configuration file (aopxfd.conf)	159
Transform classes	159
Format of an entry in the transform configuration file	160
Example of a transform entry	162
Steps for creating the transform configuration file	162
Steps for editing the transform configuration file	164
Creating the remote transform configuration file (aoprxf.conf)	165
Format of the remote transform configuration file	165
Steps for creating the remote transform configuration file	166
Setting environment variables for Transform Interface	167
Customizing the TCP/IP network firewall	168
Setting up security for transform resources	168
Creating a code page conversion table	169

Chapter 6. Customizing NetSpool 171

Identifying software requirements for NetSpool	172
Updating the BPXPRMxx member of SYS1.PARMLIB	172
Editing the aopd.conf file for NetSpool	172
Customizing the NetSpool message log	174
Customizing the NetSpool startup procedure	174
Determining the number of NetSpool tasks to start	175
Using the Japanese version	175
Using an extended MCS console	176
Creating a user ID to assign to the NetSpool startup procedure	176
Defining the NetSpool startup procedure to RACF	176
Editing the NetSpool startup procedure	177
Working with JES spool limits	180
Changing Language Environment runtime options	181
Writing NetSpool exit and filter programs	181
Programming considerations	182
APIPP1X2 control block	183
Beginning of File exit for SCS data streams (APIPP1X2)	185

Beginning of File exit for 3270 data streams (APIUBF3)	187
Transparent Data Control exit (APIPPTD2)	189
Graphic Escape exit (APIUGEX)	192
Sample exits and macro	193
Installing the exits	194
Modifying the exits	194
Setting environment variables for NetSpool	194
Editing the system configuration definition for NetSpool	195

Chapter 7. Customizing IP PrintWay 199

Comparing IP PrintWay extended mode to basic mode	202
Identifying software requirements for IP PrintWay	205
Editing the aopd.conf file for IP PrintWay (extended mode)	205
Increasing the space available in /var/Printsrv directory (extended mode)	207
Updating the BPXPRMxx member of SYS1.PARMLIB (extended mode)	207
Creating IP PrintWay job-selection rules (extended mode)	208
Creating job-selection rules in the Printer Inventory (extended mode)	211
Starting and stopping job-selection rules (extended mode)	213
Authorizing IP PrintWay to select data sets from the JES spool (extended mode)	214
Authorizing IP PrintWay to write SMF type 6 records (extended mode)	215
Setting environment variables for IP PrintWay (extended mode)	215
Editing the system configuration definition for IP PrintWay (extended mode)	223
APF-authorizing SYS1.IMAGELIB (extended mode)	230
Customizing from AFP transforms for IP PrintWay (extended mode)	230
Defining printer host names in the DNS	231
Customizing z/OS Communications Server	232
Customizing the TCPIP.DATA data set	232
Customizing the PROFILE.TCPIP data set	233
Customizing z/OS UNIX sendmail	234
Creating a VTAM APPL resource definition for IP PrintWay	236
Writing IP PrintWay exit and filter programs	237
Comparing IP PrintWay basic mode and extended mode exits	238
Resubmit for filtering considerations (basic mode)	240
Programming considerations	242
Common control blocks	243
Routing exit (ANFUXRTG)	247
Begin Data Set exit	258
Record exit	269
End Data Set exit	276
SMF exit (ANFUXSMF)	285
Message exit (ANFUXMSG) — basic mode	290
Response Notification exit (ANFUXRSP)	292
Using the block letter program (ANFUBLK)	298
IP PrintWay sample exits	299

Installing IP PrintWay exits	300
Modifying IP PrintWay exits	301
Updating the BPXPRMxx member of SYS1.PARMLIB (basic mode)	301
APF-authorizing SYS1.IMAGELIB (basic mode)	301
Defining the IP PrintWay functional subsystem (basic mode)	302
Determining how many functional subsystems to define (basic mode)	302
Determining how many functional subsystem applications to define (basic mode)	303
Determining work-selection criteria (basic mode)	303
Specifying a default font and page definition for data transforms (basic mode)	304
Defining the IP PrintWay functional subsystem in JES2 (basic mode)	305
Defining the IP PrintWay functional subsystem in JES3 (basic mode)	309
Creating FSS and FSA definitions in the Printer Inventory (basic mode)	312
Creating FSS definitions (basic mode)	312
Creating FSA definitions (basic mode)	315
Creating the transmission-queue data set (basic mode)	316
Sharing the transmission-queue data set (basic mode)	316
Allocating and initializing the transmission-queue data set (basic mode)	317
Restricting access to the transmission-queue data set (basic mode)	319
Creating the IP PrintWay message-log data set (basic mode)	319
Customizing the IP PrintWay startup procedure (basic mode)	320
Creating a user ID to assign to the IP PrintWay startup procedure (basic mode)	320
Defining the IP PrintWay startup procedure to RACF (basic mode)	321
Editing the ANFWPROC startup procedure (basic mode)	321
Changing Language Environment runtime options (basic mode)	324
Customizing PSF for sharing network printers	324

Chapter 8. Customizing Infoprint Central 327

Identifying software requirements for Infoprint Central	328
Editing the aopd.conf file for Infoprint Central	329
Initializing the search database	331
Customizing the z/OS HTTP Server	332
Editing the z/OS HTTP Server configuration file (httpd.conf)	333
Editing the z/OS HTTP Server environment variables file (httpd.envvars)	335
Setting up security for Infoprint Central	341
Authorizing the AOOPER group to RACF profiles	341
Defining Infoprint Central users as z/OS UNIX users	344

Activating the PRINTSRV class	344
Authorizing users to read the Printer Inventory	344
Setting up security for printers	345
Setting up security for NetSpool logical units (LUs)	353
Setting up security for IP PrintWay job-selection rules	354
Setting up security for print jobs	356
Setting up security for Infoprint Server daemons	362
Setting up security for printer and printer pool definitions	362
Requesting RACF notification and auditing	362
Editing the system configuration definition for Infoprint Central	363
Customizing web browsers	365
Logging on to Infoprint Central from the web	366
Customizing PSF for z/OS to use the Printer Inventory	366
Customizing JES3 console standards (CONSTD statement)	366

Chapter 9. Installing and customizing the Infoprint Server Windows client. 369

Chapter 10. Customizing the SNMP subagent 373

Identifying software requirements	373
Customizing and starting the z/OS Communications Server SNMP agent	374
Editing the aopd.conf file for the SNMP subagent	374
Editing the system configuration definition for the SNMP subagent	375
Enabling SNMP reporting in the Printer Inventory	376
Customizing the SNMP manager	377

Chapter 11. Customizing PSF for z/OS to use the Printer Inventory 379

Identifying software requirements	379
Creating PSF FSS and FSA definitions in the Printer Inventory	379
Customizing PSF for z/OS to use the Printer Inventory	380
Customizing PSF for z/OS to write messages to the common message log	380

Appendix A. Checklist of Infoprint Server customization tasks 383

Appendix B. Infoprint Server files. 389

aopd.conf -- Infoprint Server configuration file	389
Purpose	389
Format	389
Usage notes	400
aopmsg.conf -- Infoprint Server message configuration file	401
Purpose	401
Format	401
Usage notes	402

aoprxf.conf -- Infoprint Server remote transform configuration file	403
Purpose	403
Format	403
Usage notes	403
aopxfd.conf -- Infoprint Server transform configuration file	404
Purpose	404
Format	404
Example	406
Usage notes	406
aopsapd.conf -- Infoprint Server SAP callback daemon configuration file	407
Purpose	407
Format	407
Usage notes	407
Examples -- aopsapd.conf	408

Appendix C. Environment variables 409

Appendix D. Configuration attributes 435

Appendix E. AOPBATCH program 455

Format	455
Description	455
Parameters	456
Usage notes	457
Exit values	457
Examples -- AOPBATCH	458
Sample jobs	458

Appendix F. Application programming interface (API) 459

API files	459
API data structures	459
DocumentInfo - information about a document in a job	460
JobInfo - information about an Infoprint Server print job and each document in the job	461
JobToken - identifies an Infoprint Server print job	461
PrinterInfo - information about an Infoprint Server printer	461
QueryToken - identifies an Infoprint Server query request	462
Transform - information about a transform connection	462
API functions	462
Determining the API version number	462
Initializing and terminating the API	462
Creating a print job	463
Canceling a print job	463
Querying print jobs	463
Querying printers	463
Connecting to the Infoprint Server Transform Manager	463
Handling errors	464
AbortPrintFile -- terminate an Infoprint Server print job	464

BeginEnumJobs -- begin an enumeration of Infoprint Server print jobs for a printer	464
BeginEnumPrinters -- begin an enumeration of Infoprint Server printers.	465
BufferSizeRequired -- return the required size of the buffer after a buffer overflow error	465
CancelJob -- cancel a print job	466
ClosePrintFile -- close an Infoprint Server print job	466
CreatePrintFile -- create an Infoprint Server print job	467
EndEnumJobs -- end the enumeration of Infoprint Server print jobs for a printer	468
EndEnumPrinters -- end the enumeration of Infoprint Server printers.	468
EnumJobs -- enumerate Infoprint Server print jobs for a printer	468
EnumPrinters -- enumerate Infoprint Server printers	469
ErrorNumber -- return the API error code	470
ErrorString -- return a pointer to the API error message	471
GetJobInfo -- return the status of an Infoprint Server print job.	471
GetPrinterInfo -- return information about an Infoprint Server printer	472
InitAPI -- initialize the API.	473
ReleaseTransform -- release a connection with a transform daemon.	473
RequestTransform -- establish a connection with a transform daemon	474
SetTerminationHandler -- register a termination function	475
TermAPI -- terminate the API	475
WritePrintFile -- write data to an Infoprint Server print job.	475

Programming considerations	476
Redefining API function names	476
Linking to API functions	477
APF-authorizing your application	477
Writing a termination handler function.	477
Customization requirements	478

Appendix G. Migrating from IP PrintWay basic mode to extended mode 479

Appendix H. Exploiting new functions in z/OS V2R1 483

Dynamic configuration	483
MVS system logger	483
SMF type 6 record enhancements.	484

Appendix I. Accessibility 485

Using assistive technologies	485
Keyboard navigation of the user interface	485

Notices 487

Policy for unsupported hardware.	488
Minimum supported hardware	488
Programming interfaces	489
Trademarks	489

Glossary 491

Index 499

Figures

1. Infoprint Server system diagram	1	16. Sample UNIX filter	124
2. Printer Inventory Manager system diagram	3	17. AOPPRINT JCL procedure in SYS1.IBM.PROCLIB	131
3. Print Interface system diagram	6	18. NetSpool startup procedure — SYS1.IBM.PROCLIB(APIJPJCL).	178
4. Printing SAP R/3 documents with an SAP R/3 application server running on a z/OS system	10	19. Language Environment runtime options for NetSpool	181
5. Printing SAP R/3 documents with an SAP R/3 application server running on a non-z/OS system	11	20. Format of APIPP1X2 control block	184
6. NetSpool system diagram.	13	21. Sample APPL statement for IP PrintWay extended mode	236
7. IP PrintWay system diagram	16	22. Sample APPL statement for IP PrintWay basic mode	237
8. Transform Interface system diagram	19	23. Format of the IP PrintWay exit parameter area (ANFUEXTP)	245
9. SNMP subagent system diagram	23	24. Sample JES2 initialization statements	306
10. AOPSTART startup procedure — SYS1.IBM.PROCLIB(AOPSTART)	63	25. Sample JES3 initialization statements	309
11. AOPSTOP shutdown procedure — SYS1.IBM.PROCLIB(AOPSTOP).	65	26. IDCAMS parameters to allocate a transmission-queue data set — SYS1.SAMPLIB(ANFDEAL).	318
12. Sample JCL to define a DASD-only log stream to the system logger	96	27. IP PrintWay startup procedure — SYS1.IBM.PROCLIB(ANFWPROC)	322
13. Sample JCL to define a coupling-facility structure to the system logger	98	28. Language Environment runtime options for IP PrintWay	324
14. Sample JCL to define a coupling-facility structure in the CFRM policy	99		
15. Sample JCL to define a coupling-facility log stream to the system logger	99		

Tables

1. Documents for Infoprint Server	xv	34. How IP PrintWay extended mode differs from basic mode	204
2. Documents for IBM transform products	xvi	35. IP PrintWay job-selection rules	208
3. Documents for Ricoh products	xvi	36. Contents of ANFUEXTP fields when resubmit for filtering option is selected (IP PrintWay basic mode).	241
4. Transforms that work with Infoprint Server	20	37. IP PrintWay FSS definition	313
5. Summary of Infoprint Server components	25	38. Customization tasks for Infoprint Central	328
6. Customization tasks for Printer Inventory Manager	27	39. Other components to customize for Infoprint Central.	328
7. Number of documents in the Historical Inventory	40	40. Environment variables in httpd.envvars	335
8. Where to define environment variables for the Printer Inventory Manager	44	41. Profiles in the OPERCMDS class and minimum access required for the AOPOPER group— JES2 systems.	342
9. RACF access for Infoprint Server	47	42. Profiles in the OPERCMDS and JESSPOOL classes and minimum access required for the AOPOPER group— JES3 systems	342
10. Dynamic attributes and ISPF field names	69	43. Infoprint Central printer actions and the minimum access required to profiles in the PRINTSRV class	345
11. Attributes in system configuration definition and the equivalent attributes in aopd.conf and aopmsg.conf and equivalent environment variables	70	44. Infoprint Central printer actions and minimum access required to profiles in the OPERCMDS class — JES2 systems	350
12. Number of documents in the Historical Inventory	79	45. Infoprint Central printer actions and the minimum access required to profiles in the OPERCMDS class — JES3 systems	351
13. Customization tasks for system logger	88	46. Infoprint Central NetSpool LU actions and the minimum access required to the AOP.NETSPOOL profile in the PRINTSRV class	353
14. DASD-only log-stream parameters for Infoprint Server	95	47. Infoprint Central job-selection rule actions and minimum access required to the profile in the PRINTSRV class	354
15. Coupling-facility structure parameters for Infoprint Server	97	48. Infoprint Central print job actions and minimum access required to profiles in the OPERCMDS class – JES2.	357
16. Coupling-facility log-stream parameters for Infoprint Server	97	49. Infoprint Central print job actions and the minimum access required to profiles in the OPERCMDS class – JES3.	358
17. Customization tasks for Print Interface	115	50. Infoprint Central print job actions and the minimum access required to profiles in the JESSPOOL class.	360
18. Common customization tasks for Print Interface	117	51. Infoprint Central daemon actions and the minimum access required to the AOP.DAEMON profile in the PRINTSRV class	362
19. Customization tasks for Print Interface LPD	132	52. Comparison of Windows printing methods	369
20. Customization tasks for SMB	139	53. Customization tasks for SNMP subagent	373
21. Customization tasks for Print Interface IPP Server	140	54. Customization tasks for PSF for z/OS	379
22. IPP attributes and corresponding Infoprint Server attributes	144	55. Number of documents in the Historical Inventory	396
23. Customization tasks for Print Interface when SAP R/3 application server runs on same z/OS system.	145	56. Where to set environment variables	409
24. Customization tasks for Print Interface when SAP R/3 application server does not run on same z/OS system.	145	57. Number of documents in the Historical Inventory	449
25. Customization tasks for Print Interface subsystem	153		
26. Customization tasks for Transform Interface	157		
27. Other components to customize for Transform Interface	157		
28. Customization tasks for NetSpool.	171		
29. Other components to customize for NetSpool	171		
30. Customization tasks for IP PrintWay extended mode	199		
31. Customization tasks for IP PrintWay basic mode	200		
32. Customization tasks for IP PrintWay protocols	201		
33. Other components to customize for IP PrintWay	201		

About this document

This document helps you customize Infoprint Server in z/OS® Version 2 Release 1. It describes the different components of Infoprint Server and how to customize each component.

Who should read this document

This document is for system programmers and administrators responsible for customizing Infoprint Server for their installation. You should be familiar with z/OS UNIX System Services, TCP/IP, the z/OS job entry subsystem (JES), and z/OS job control language (JCL).

This document also contains reference information, including:

- Syntax of Infoprint Server configuration files
- Environment variables used by Infoprint Server
- Attributes in the Infoprint Server system configuration definition
- AOPBATCH utility program
- Infoprint Server callable services
- Actions to take when migrating from IP PrintWay™ basic mode to extended mode
- Summary of new functions in z/OS V2R1 and how to use them

How to read syntax diagrams

This section explains the general notations that this document uses in syntax diagrams. For ease of reading, this document breaks some examples into several lines. However, when you enter a command, enter it all on one line. Do not press Enter until you have typed the entire command.

This notation:	Means:	You enter:	For example:	
			This document shows:	You enter:
Apostrophes	String	As shown	SEND '123'	SEND '123'
Bold	Keyword	As shown	CLASS	CLASS
Braces	List of items	The braces and one or more items from the list	{GT10 GT12}	{GT10 GT12}
Brackets	Optional item	One item or no items	aopstop [now]	aopstop
Comma	Separator	As shown	DISPLAY C,K	DISPLAY C,K
Ellipsis	Repeatable item	One or more items	<i>filename ...</i>	file1 file2
Lowercase	Item the system defines	As shown, in lowercase	lp	lp
Lowercase italics	Variable item	A value for the item	MOUNT <i>devnum</i>	MOUNT A30
Parentheses	List of items	The parentheses and one or more items from the list	(GT10,GT12)	(GT10,GT12)

This notation:	Means:	You enter:	For example:	
			This document shows:	You enter:
Special characters	Various symbols	As shown	%filter-options	%filter-options
Underline	Default	The item, or you can omit it	K T <u>REF</u>	K T
Uppercase	Item the system defines	As shown, in uppercase	PRMODE	PRMODE
Vertical bar	UNIX pipe (the output of the first is input to the second)	As shown	ls lp	ls lp
Vertical bar in braces	Required choice	One item	{NOW FOREVER}	FOREVER
Vertical bar in brackets	Optional choice	One item or no items	[PORTNO PRTQUEUE]	PORTNO

Where to find more information

This section describes where to find information related to z/OS and Infoprint Server.

Preventive Service Planning information

Before installing Infoprint Server, you should review the current Preventive Service Planning (PSP) information, also called the *PSP bucket*. You should also periodically review the current PSP information. The PSP upgrade ID depends on your z/OS operating system, for example: ZOSV2R1. The subset for Infoprint Server is: INFOPRINT.

To obtain the current PSP bucket, contact the IBM® Support Center or use z/OS SoftwareXcel (IBMLink). If you obtained z/OS as part of a CBPDO, HOLDDATA and PSP information is included on the CBPDO tape. However, this information might not be current if the CBPDO tape was shipped several weeks prior to installation.

Infoprint Server migration information

This section lists documents that can help you migrate from previous releases of z/OS and from IP PrintWay basic mode to IP PrintWay extended mode.

Document	Form number
z/OS Migration	GA32-0889
Describes the tasks required to migrate to z/OS V2R1. Includes the migration tasks for Infoprint Server.	
z/OS Summary of Message and Interface Changes	SA23-2300
Describes new and changed messages and interfaces in z/OS V2R1. Includes the messages and interfaces for Infoprint Server.	

Document	Form number
<i>z/OS Infoprint Server Customization</i>	SA38-0691
Describes the tasks required to migrate to IP PrintWay extended mode from IP PrintWay basic mode. Also, describes the tasks required to use the new functions introduced in z/OS V2R1.	

Documents

This section lists related documents that can help you use Infoprint Server and transform products. For documents for all z/OS products, see *z/OS Information Roadmap*.

Table 1. Documents for Infoprint Server

Document	Form number
<i>z/OS Infoprint Server Introduction</i>	SA38-0692
Introduces Infoprint Server. This document contains printing scenarios that show how you can use Infoprint Server in your installation.	
<i>z/OS Infoprint Server Customization</i>	SA38-0691
Describes customization tasks for Infoprint Server. This document describes Infoprint Server environment variables, configuration files, startup procedures, how to write exit routines and filter programs, and how to use the Infoprint Server API.	
<i>z/OS Infoprint Server Operation and Administration</i>	SA38-0693
Describes operator procedures and administrative tasks for Infoprint Server. This document describes how to start and stop Infoprint Server and how operators can use Infoprint Central. It describes how administrators can create entries in the Printer Inventory using either ISPF panels or the Printer Inventory Definition Utility (PIDU) program and define NetSpool printer LUs to VTAM®.	
<i>z/OS Infoprint Server Printer Inventory for PSF</i>	SA38-0694
Describes the Printer Inventory for PSF for PSF customers who do not purchase an Infoprint Server license. It describes the tasks required to customize Infoprint Server, start and stop Infoprint Server, create PSF FSS and FSA definitions in the Printer Inventory, and diagnose problems in Infoprint Server.	
<i>z/OS Infoprint Server User's Guide</i>	SA38-0695
Describes user tasks for Infoprint Server. This document describes how to submit print jobs from remote systems (including Windows systems), the local z/OS system, and Virtual Telecommunications Access Method (VTAM) applications. It describes z/OS UNIX commands; the AOPPRINT JCL procedure; the AOPBATCH program; DD and OUTPUT JCL parameters that Infoprint Server supports; and how to download and install the Infoprint Port Monitor for Windows.	
<i>z/OS Infoprint Server Messages and Diagnosis</i>	GA32-0927
Describes messages from Infoprint Server. This document also describes how to use Infoprint Server tracing facilities to diagnose and report errors.	

Table 2. Documents for IBM transform products

Document	Form number
<p><i>IBM Infoprint Transforms to AFP for z/OS</i></p> <p>Describes IBM Infoprint Transforms to AFP for z/OS (5655-N60)</p>	G550-0443
<p><i>IBM Print Transforms from AFP for Infoprint Server for z/OS</i></p> <p>Describes these products:</p> <ul style="list-style-type: none"> • IBM Print Transform from AFP to PCL for Infoprint Server for z/OS (5655-TF2) • IBM Print Transform from AFP to PDF for Infoprint Server for z/OS (5655-TF1) • IBM Print Transform from AFP to PostScript for Infoprint Server for z/OS (5655-TF3) 	G325-2634
<p><i>IBM Infoprint XT for z/OS</i></p> <p>Describes IBM Infoprint XT for z/OS (5655-O15)</p>	GI11-9492

Table 3. Documents for Ricoh products

Ricoh Document
<p><i>InfoPrint Transform Manager for AIX®, Linux, and Windows</i></p> <p>Describes InfoPrint Transform Manager for AIX, Linux, and Windows</p>
<p><i>InfoPrint Transform Manager for AIX, Linux, and Windows: afp2pdf Transform Installation and User's Guide</i></p> <p>Describes the InfoPrint Transform Manager for Linux AFP to PDF transform</p>

Infoprint Server online help

The Infoprint Server ISPF panels contain integrated online help for each panel and field. To view the help, place your cursor on a panel or in a field on a panel and press the Help function key (F1).

Infoprint Central contains an integrated online help system. To view the help system, select the question mark (?) in the left navigation bar. To view the help for a particular Web page, select the question marks (?) or the **Help** buttons on the Web page.

How to send your comments to IBM

We appreciate your input on this publication. Feel free to comment on the clarity, accuracy, and completeness of the information or provide any other feedback that you have.

Use one of the following methods to send your comments:

1. Send an email to mhvrcfs@us.ibm.com.
2. Send an email from <http://www.ibm.com/systems/z/os/zos/webqs.html>
3. Mail the comments to the following address:
IBM Corporation
Attention: MHVRCFS Reader Comments
Department H6MA, Building 707
2455 South Road
Poughkeepsie, NY 12601-5400
US
4. Fax the comments to us, as follows:
From the United States and Canada: 1+845+432-9405
From all other countries: Your international access code +1+845+432-9405

Include the following information:

- Your name and address
- Your email address
- Your telephone or fax number
- The publication title and order number:
z/OS V2R1.0 Infoprint Server Customization
SA38-0691-00
- The topic and page number related to your comment.
- The text of your comment.

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

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

If you have a technical problem

Do not use the feedback methods that are listed for sending comments. Instead, take one of the following actions:

- Contact your IBM service representative.
- Call IBM technical support.
- Visit the IBM Support Portal at <http://www.ibm.com/systems/z/support/>.

Summary of changes

This section describes the release enhancements that were made.

z/OS Version 2 Release 1

Refer to the following publications for specific enhancements for z/OS Version 2 Release 1:

- *z/OS Summary of Message and Interface Changes*, SA23-2300
- *z/OS Introduction and Release Guide*, GA32-0887
- *z/OS Planning for Installation*, GA32-0890
- *z/OS Migration*, GA32-0889

Chapter 1. Introducing Infoprint Server

Infoprint Server provides support for LAN and host printing on your z/OS system. It works together with data stream transforms that other transform products provide. Figure 1 shows how most of the components of Infoprint Server fit into your system. The components of Infoprint Server and the transform products are shaded. Components of Infoprint Server are displayed in yellow, while transform products are displayed in blue. Following the figure is a description of each component.

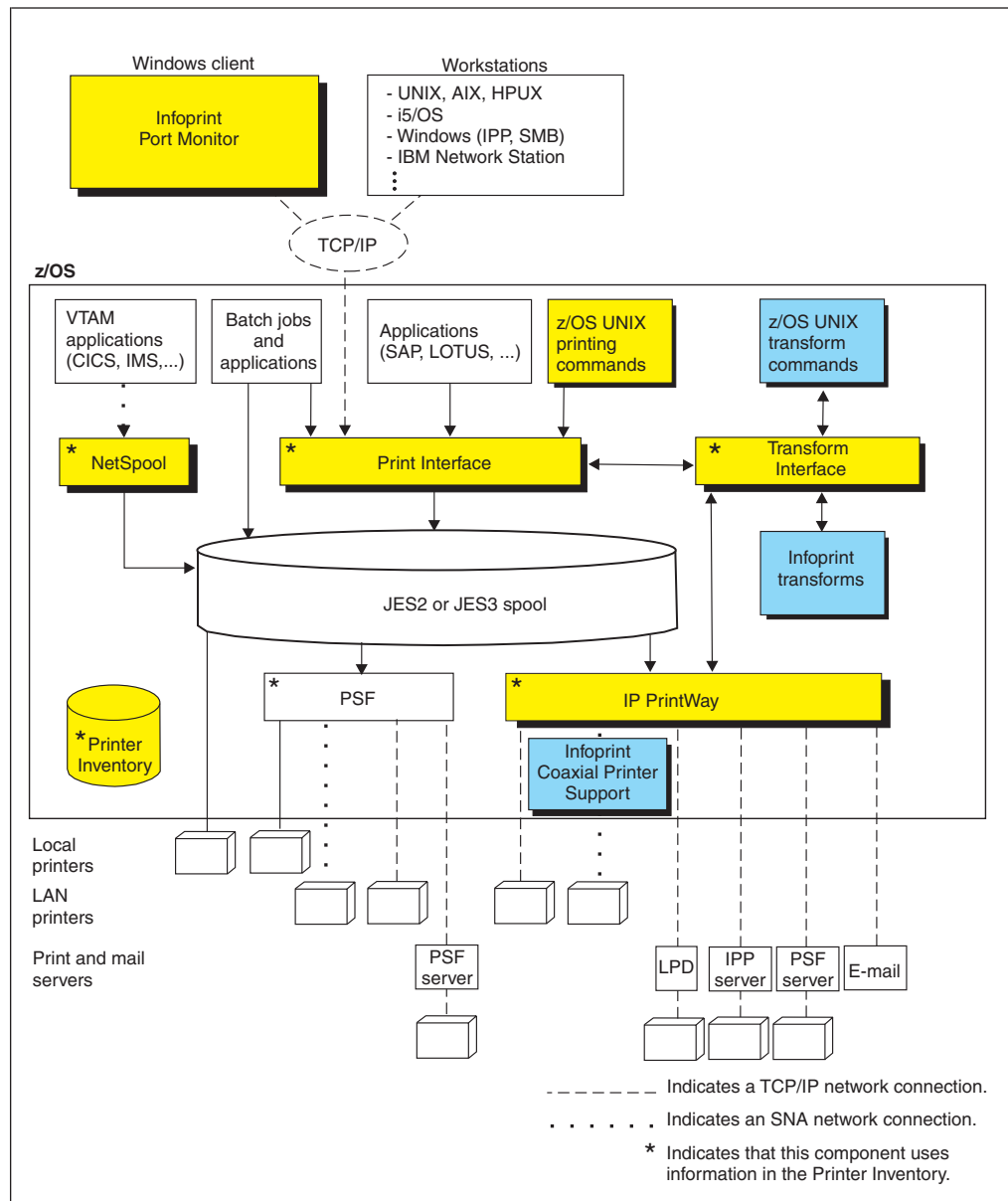


Figure 1. Infoprint Server system diagram

Printer Inventory and Printer Inventory Manager

The Printer Inventory Manager controls the Printer Inventory. The Printer Inventory consists of files in a z/OS UNIX file system (HFS or zFS) that contain information about printers and print jobs. The Printer Inventory also contains information about system configuration.

Infoprint Server Windows Client

The Infoprint Server Windows client consists of the Infoprint Port Monitor, which sends print requests and job attributes to Print Interface.

Print Interface

Print Interface processes print requests from remote clients and from the local z/OS system and allocates output data sets on the JES spool. Print Interface accepts various data formats and can transform input data streams to EBCDIC line data, ASCII text data, AFP, PCL, PostScript, PDF, or other data formats that the printer accepts. A separate transform product is required for some transforms.

NetSpool

NetSpool processes print requests from VTAM applications, such as CICS® and IMS™, and allocates output data sets on the JES spool. NetSpool accepts SCS, 3270, and binary data streams and can transform input data streams to EBCDIC line data, PCL, PDF, AFP, or other data formats that the printer accepts. A separate transform product is required for some transforms. However, a separate transform product is not required to convert input data streams to the line or PCL formats.

IP PrintWay

IP PrintWay transmits data sets from the JES spool to printers or print servers in a TCP/IP or SNA network and to email destinations. IP PrintWay accepts various data formats and can transform input data streams to ASCII text data, PCL, PostScript, PDF, or other data formats that the printer accepts. A separate transform product is required for some transforms.

Transform Interface

Transform Interface communicates with other transform products to transform data to another data format. It manages the transforms that run on z/OS, and it sends data to transforms that run on other systems, including Linux, AIX, and Windows systems.

Infoprint Central (not shown in figure)

Infoprint Central is a web-based application that lets help desk operators work with print jobs (output data sets) on the JES spool, printers controlled by IP PrintWay extended mode or PSF, and NetSpool logical units. It also lets operators check system status and view printer definitions.

Simple Network Management Protocol (SNMP) subagent (not shown in figure)

The SNMP subagent lets you use an SNMP manager to view printer characteristics and printer status for printers that PSF controls and that do not have internal SNMP agents or that are not TCP/IP-attached to PSF.

Transforms (separate products)

Transform products can transform data streams from one data format to another. These products are separate from Infoprint Server.

PSF for z/OS (separate product)

PSF for z/OS can use system configuration information stored in the Printer Inventory. In addition, PSF V4R4 and later can write messages for print jobs and printers to the Infoprint Server common message log.

Printer Inventory Manager

The Printer Inventory Manager component of Infoprint Server controls the Printer Inventory. The Printer Inventory consists of Hierarchical File System (HFS) or z/OS File System (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 objects in the Printer Inventory are:

Printer definitions

Contain information about printers and email destinations.

Printer pool definitions

Contain information about groups of printer definitions that you want to broadcast data to (applies only to NetSpool).

FSS definitions

Contain configuration information for IP PrintWay basic mode functional subsystems (FSSs) and PSF FSSs.

FSA definitions

Contain configuration information for IP PrintWay basic mode functional subsystem applications (FSAs) and PSF FSAs.

Job selection rules

Contain rules that IP PrintWay extended mode uses to determine which print jobs to select from the JES spool for printing.

System configuration definition

Contains configuration information for Infoprint Server. Infoprint Server creates this definition automatically. Administrators can modify it if the dynamic configuration function is enabled.

Figure 2 shows how the administrator can create objects in the Printer Inventory and which components of Infoprint Server use the Printer Inventory.

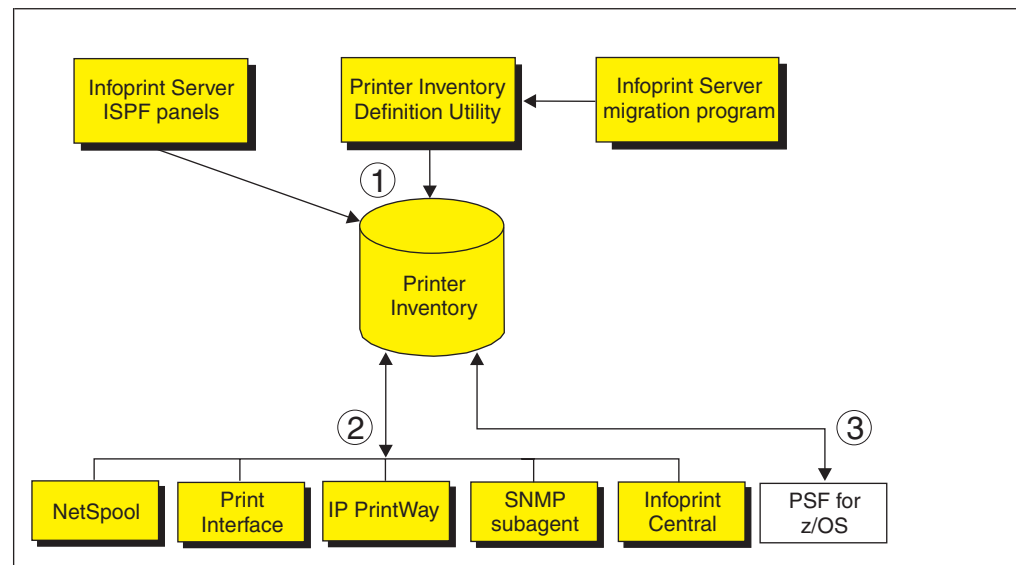


Figure 2. Printer Inventory Manager system diagram

1. The administrator can use Infoprint Server ISPF panels and the Printer Inventory Definition Utility (PIDU) to create and maintain the Printer Inventory. PIDU is useful for creating many printer definitions at the same time and for backing up the Printer Inventory.
2. These Infoprint Server components use information in the Printer Inventory:
 - 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.
3. PSF and the AFP Download Plus feature of PSF can, as an option, use configuration information that is specified in FSS and FSA definitions in the Printer Inventory. The system programmer can alternatively specify this configuration information in PSF and AFP Download Plus startup procedures and 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 printers. (Operators can work with AFP Download Plus senders in the same way they work with PSF printers.) For example, operators can start and stop (that is, drain) printers, change JES work-selection criteria, and turn TCP/IP-connected printers online and offline.
 - PSF V4R4 and later can write messages for print jobs and printers to the Infoprint Server common message log. Authorized users can view these messages in Infoprint Central. 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.

Additional functions that the Printer Inventory Manager provides are:

- **Common message log**

The common message log lets you see messages from Infoprint Server, from some transform products, and from PSF for z/OS V4R4 and later in one place. The log contains messages from all components of Infoprint Server except for IP PrintWay basic mode. For information about whether a transform can write messages to the common message log, see the documentation for the transform.

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.

You can view and process messages in the common message log in these ways:

- Authorized users can use Infoprint Central to view messages for selected print jobs, IP PrintWay printers, PSF printers, and NetSpool LUs.
- 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.

- **Hardcopy log**

Infoprint Server administrators can select messages to send to the z/OS system log (SYSLOG) or operations log (OPERLOG) for message automation.

- **Historical Inventory**

The Historical Inventory contains information about print jobs 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 view information about these print jobs.

- **Migration program**

The Infoprint Server migration program copies configuration information from PSF and AFP Download Plus startup procedures to the Printer Inventory. It automatically creates PSF FSS and FSA definitions in the Printer Inventory.

- **Security**

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

- **Printer Inventory for PSF**

PSF customers who do not purchase an Infoprint Server license can specify PSF and AFP Download Plus configuration information in the Printer Inventory. For information, see *z/OS Infoprint Server Printer Inventory for PSF*.

Windows client

The Infoprint Server Windows client consists of this program:

Infoprint Port Monitor for Windows

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. After the Infoprint Port Monitor is installed and configured on the Windows system, it automatically sends documents to the Print Interface component of Infoprint Server.

Infoprint Port Monitor V3.1 runs on Windows Server 2003, Windows Server 2008, Windows 7, Windows Vista (Business Edition, Enterprise Edition), and Windows XP.

Note: Infoprint Port Monitor V3.1 can run on 32-bit Windows systems or on 64-bit Windows systems that are based on x64 architecture. However, it cannot run on 64-bit Windows systems that are based on Intel Itanium architecture.

Infoprint Port Monitor V3.1 is shipped with Infoprint Server. You can also download Infoprint Port Monitor V3.1 at <http://www-01.ibm.com/support/docview.wss?uid=psd1P1000597>.

Note: Infoprint Server also supports printing from a Windows system with the Server Message Block (SMB) protocol and the Internet Printing Protocol (IPP). To use the SMB and IPP protocols, Windows users do not need to install the Infoprint Port Monitor.

You might want to use these products if your installation has AFP printers or prints AFP documents:

IBM AFP Printer Driver for Windows

Creates output files in AFP format. Files in AFP format can be printed on AFP printers. The 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.

IBM AFP Viewer plug-in for Windows

Lets you view documents in AFP format. For example, documents downloaded from the z/OS system or documents on the web. The viewer also lets you print AFP documents on AFP printers and on non-AFP printers.

Note: You can download the AFP Printer Driver for Windows and the AFP Viewer plug-in for Windows from the IBM website at <http://www.ibm.com>. On the IBM website, search for "AFP Printer Driver" and "AFP Viewer plug-in".

Print Interface

The Print Interface component of Infoprint Server processes print requests received from both remote clients and local users. Figure 3 shows the steps that occur from the time Print Interface receives a print request until it allocates an output data set on the JES spool. An explanation of each step follows.

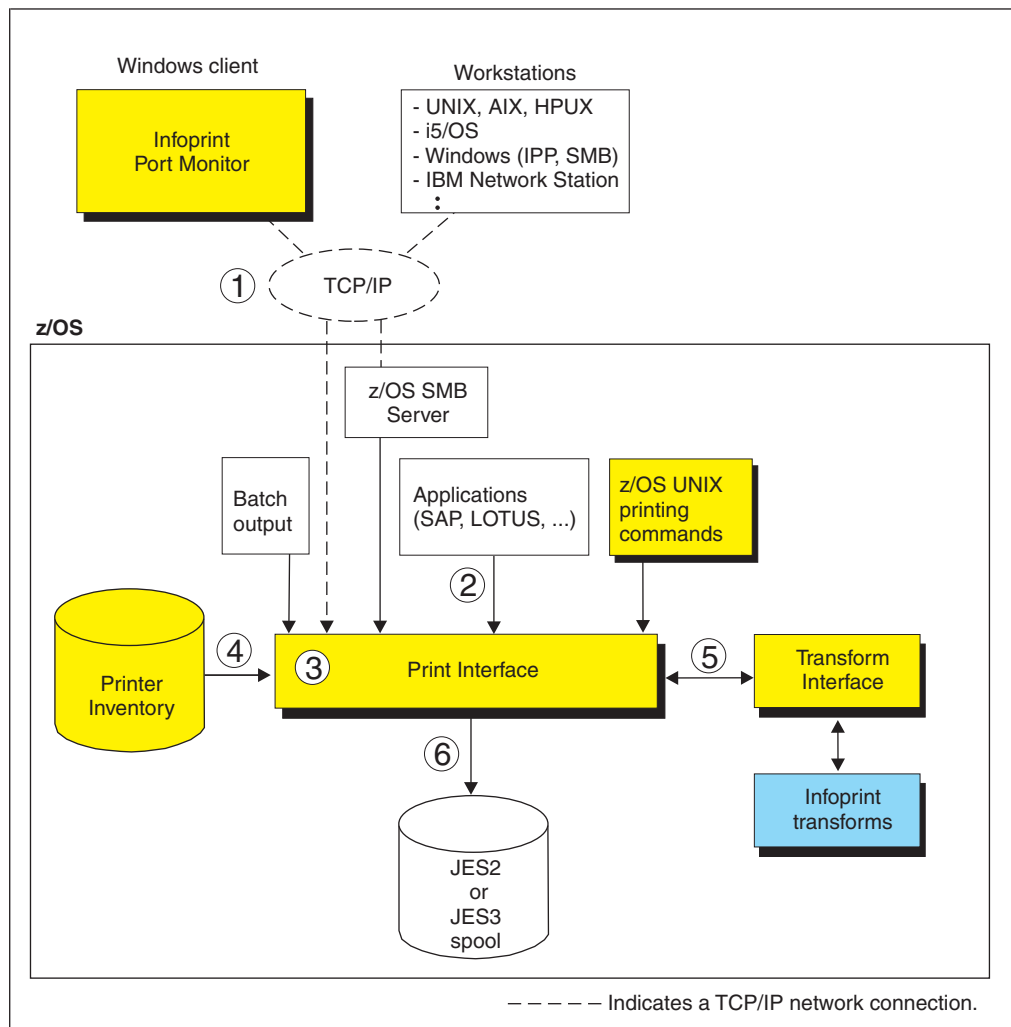


Figure 3. Print Interface system diagram

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

- LPR protocol: The LPR protocol is defined by RFC 1179. Clients that use this protocol include:
 - The Infoprint Port Monitor for Windows. Infoprint Server provides this client (see “Windows client” on page 5).
 - The **lprafp** command for Windows and UNIX systems. This command lets you specify Infoprint Server job attributes. You can download the **lprafp** command from the IBM downloads website at <http://www.ibm.com/systems/z/os/zos/downloads/>. On the IBM downloads website, in the “As-is z/OS Downloads” section, click **Infoprint Server for z/OS**.
 - TCP/IP commands such as **lpr**, **enq**, and **lpq**.
 - An SAP R/3 application server that runs on a remote system.
 - Internet Printing Protocol (IPP): IPP is a standard protocol for printing over the Internet. An IPP client must run in the remote system.
 - Server Message Block (SMB) printing protocol: 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.
2. Users can submit print requests from the local z/OS system using one of these methods:
 - Print Interface subsystem: Using the Print Interface subsystem, you can transform and print output data created by a batch application with minimal changes to your JCL.
 - AOPPRINT: Using the AOPPRINT JCL procedure, you can print existing MVS™ data sets and UNIX files.
 - z/OS UNIX 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 command line or from a UNIX application.
 - Infoprint Server SAP Output Management System (OMS): Using the Infoprint Server 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.
 3. 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, and text.
 4. Each print request specifies the name of a printer definition in the Printer Inventory. The printer definition can describe a printer or an email destination. Print Interface uses information in the printer definition to determine items, such as how to process the data or whether to transform the data.
 5. Print Interface can, in most cases, automatically detect the data format of the input data and validate that the printer accepts that data format. Print Interface can convert data between EBCDIC and ASCII. Also, it can use other transform products to transform data into other formats for printing or emailing.
 6. For each print request, Print Interface dynamically allocates an output data set on the JES2 or JES3 spool. In a JES2 environment, you can request that Print Interface allocate output data sets on the spool of the primary JES2 subsystem or a secondary JES2 subsystem. Print Interface uses 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 PSF or IP PrintWay.
- Advanced Function Presentation (AFP) parameters, such as the name of a form definition and page definition. PSF uses these parameters when printing data on AFP printers.

Some additional functions that Print Interface provides include:

- **Transforms**

Print Interface can transform data to line data or text format without using a separate transform product. It can:

- Transform text data into line data for printing on AFP printers.
- Transform line data (for example, in a sequential data set or a partitioned data set) into text data for printing on a network printer or emailing.

Print Interface can also transform data to AFP format for printing on AFP printers, and it can transform data to PCL, PostScript, or PDF format for printing or emailing. To do these transforms, Print Interface can use the transform products described in “Transforms” on page 20.

- **Validation of print requests**

Print Interface can validate, with some exceptions, that the document can print as requested on the selected printer. For example, 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.

- **Notification of completion**

Print Interface can notify users on the local z/OS system when processing of a document is complete and the data set has been removed from the JES spool. It can also notify users who request mail notification with a command, such as `lpr`, that uses the LPR to LPD protocol.

- **Operator control**

The operator can use tools such as SDSF and Infoprint Central to find output data sets that Print Interface writes to the JES spool. Infoprint Central can display additional status such as whether the output data sets (called *print jobs*) have completed successfully, have been retained due to failed transmission to LAN printers, or were deleted before printing.

- **Identification of output**

Print Interface retains the user ID of the job submitter for printing on separator pages and for display on the JES spool, so that the user ID can be printed on separator pages and the operator can see the name of the job submitter when the data set is on the JES spool.

- **Double-byte character set (DBCS) support**

Print Interface can convert DBCS data from one code page to another before writing the data to the JES spool.

- **Filters**

You can write filter programs to modify data before Print Interface writes the data to the JES spool. For example, a filter can add separator pages or transform data from one format to another.

- **SAP Output Management System (OMS)**

Print Interface provides an SAP OMS with a Callback daemon to support printing with the SAP R/3 Application Server for z/OS. The OMS and Callback daemon let SAP users print, cancel jobs, obtain job status, and receive immediate notification about job events.

The SAP-certified functions that Print Interface provides are:

- OMS Polling Interface
- OMS/XOM Callback Interface
- OMS Operations Supplement

For more information about SAP certification, visit the SAP website at <http://www.sap.com>. "Print Interface with an SAP R/3 application server on the z/OS system" describes this support.

The Print Interface LPD can print documents received from an SAP R/3 application server running on another system, such as an AIX or Windows system. "Print Interface with an SAP R/3 application server on a remote system" on page 11 describes this support.

Print Interface with an SAP R/3 application server on the z/OS system

Figure 4 on page 10 shows how the Print Interface SAP Output Management System (OMS), Callback daemon, and the SAP R/3 Application Server for z/OS fit into your system. When the SAP R/3 Application Server runs on the same z/OS system, the Print Interface OMS receives print and status requests, and the Callback daemon provides immediate notification of job events. The SAP R/3 Application Server for z/OS and its spool work process must run on the same system as Infoprint Server. The Print Interface Callback daemon can return notification of job events to SAP R/3 application servers running on other SAP R/3 systems, provided that the SAP spool work process runs on the z/OS system.

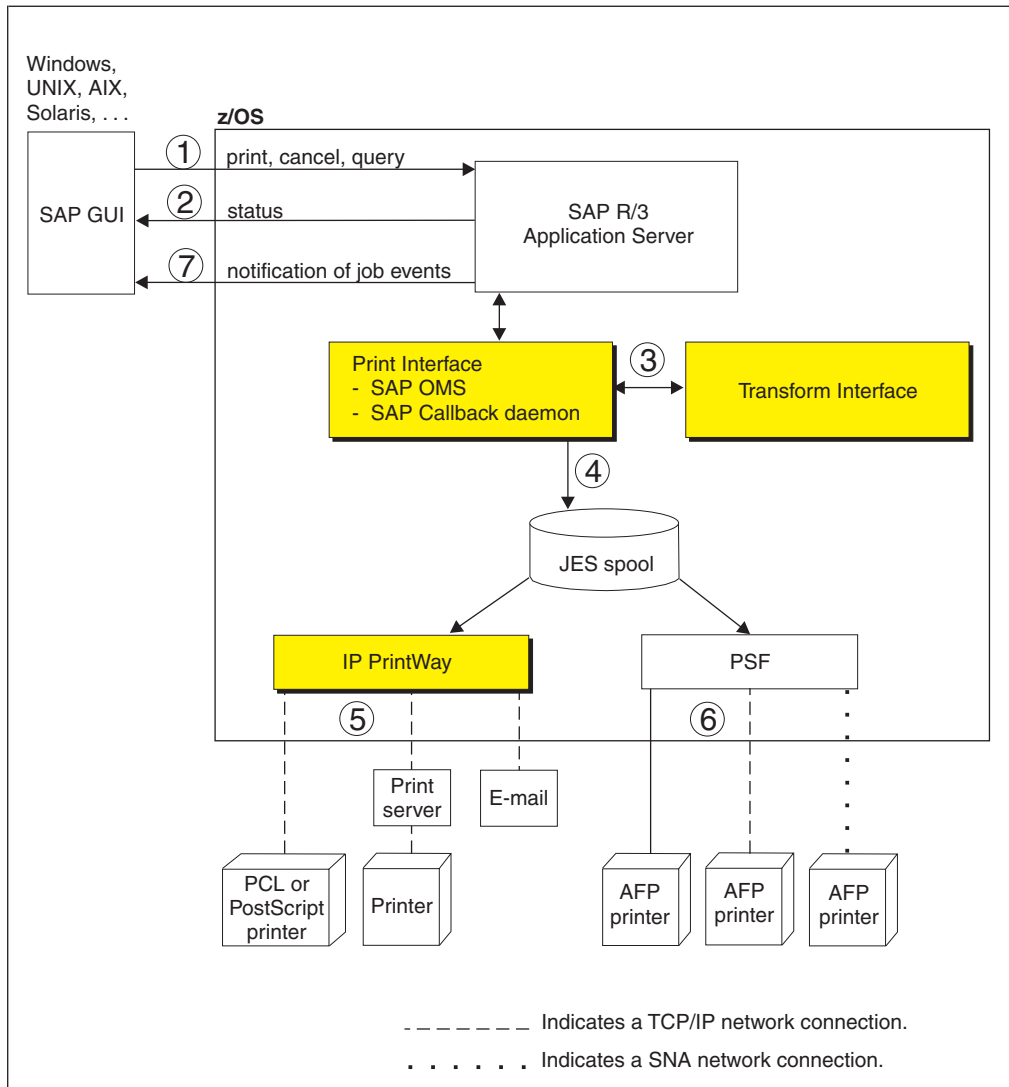


Figure 4. Printing SAP R/3 documents with an SAP R/3 application server running on a z/OS system

1. From an SAP R/3 GUI, users make print, cancel, job query, and device query requests to the SAP R/3 Application Server. These print requests specify the name of an SAP R/3 output device defined to the SAP R/3 system. The SAP administrator associates each output device with a printer definition in the Infoprint Server Printer Inventory.
2. For a status request, the Print Interface SAP OMS returns the status of a print job or a list of print jobs that the printer is processing.
3. For a print request, Print Interface automatically detects the data format. It calls Transform Interface to transform the data to another format if the administrator has specified a filter in the printer definition. For example, if the data format is PCL or PostScript, Transform Interface can call the PCL to AFP transform or the PostScript to AFP transform to transform the data to AFP format for printing on an AFP printer.
4. Print Interface creates an output data set on the JES spool. From the JES spool, IP PrintWay, PSF, or JES can print the document.
5. IP PrintWay selects data sets from the JES spool and transmits them to remote printers, print servers, or email destinations.

6. PSF selects data sets from the JES spool and prints them on AFP printers. The printers can be local, TCP/IP-attached, or SNA-attached.
7. As print jobs complete (successfully or unsuccessfully), the Print Interface SAP callback daemon sends notification back to the SAP R/3 system.

Print Interface with an SAP R/3 application server on a remote system

Figure 5 shows how the Print Interface LPD and an SAP R/3 application server running on a remote system (such as AIX or Windows) fit into your system. When no SAP R/3 application server is running on the z/OS system, the Infoprint Server LPD (as opposed to the Infoprint Server OMS) receives print and status requests, and notification of job events does not occur.

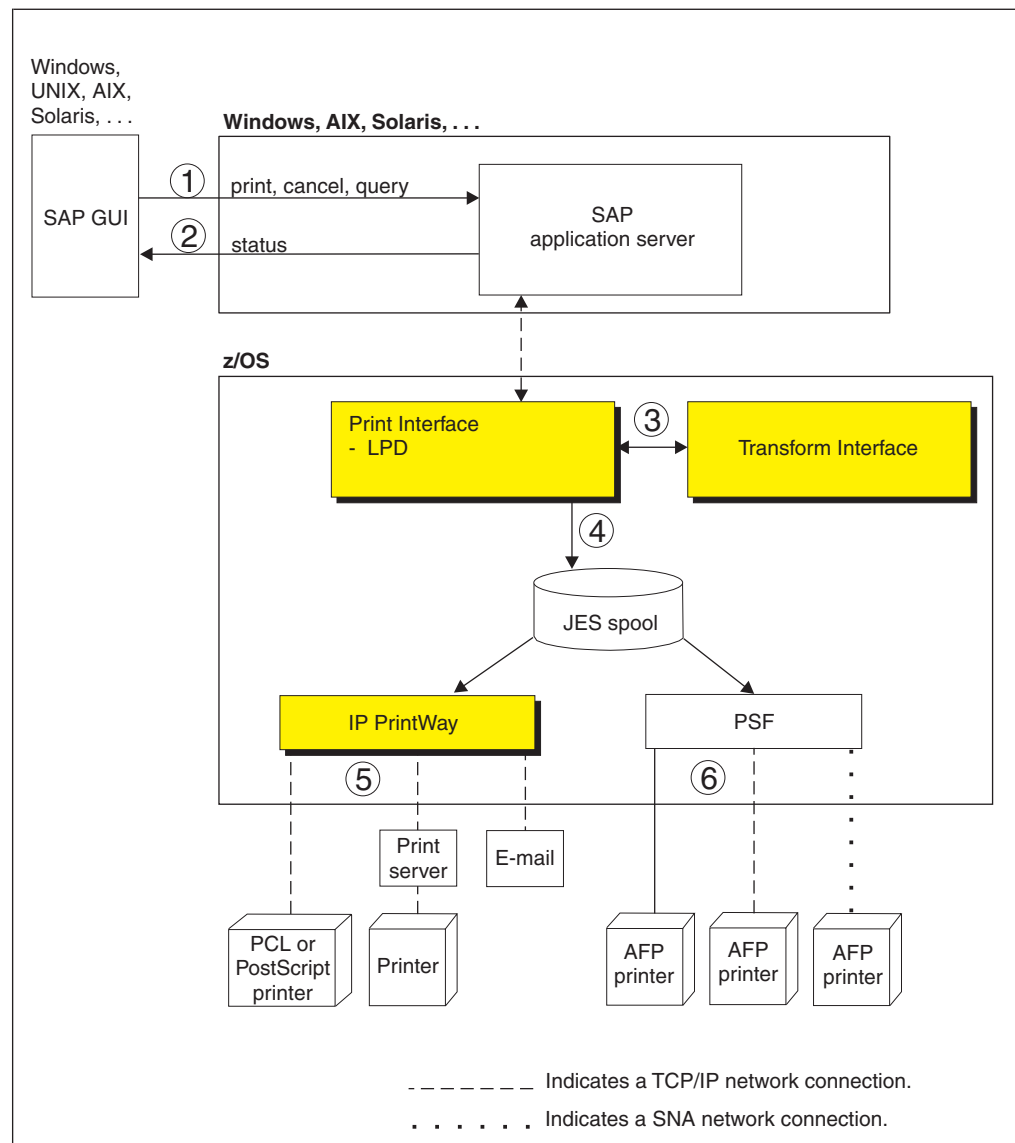


Figure 5. Printing SAP R/3 documents with an SAP R/3 application server running on a non-z/OS system

1. From an SAP R/3 GUI, users make print, cancel, job query, and device query requests to an SAP R/3 application server running on a remote system such as Windows, AIX, or Solaris. These print requests specify the name of an SAP R/3

output device that is defined to the SAP R/3 system. The SAP administrator associates each output device with a printer definition in the Infoprint Server Printer Inventory.

2. For a status request, the Print Interface LPD returns the status of a print job or a list of print jobs that the printer is processing.
3. For a print request, Print Interface automatically detects the data format. It calls Transform Interface to transform the data to another format if the administrator has specified a filter in the printer definition. For example, if the data format is PCL or PostScript, Transform Interface can call the PCL to AFP transform or the PostScript to AFP transform to transform the data to AFP format for printing on an AFP printer. The transform you need depends on the type of output the SAP R/3 output device creates.
4. Print Interface creates an output data set on the JES spool. From the JES spool, IP PrintWay, PSF, or JES can print the document, or IP PrintWay can send it to an email destination.
5. IP PrintWay selects data sets from the JES spool and transmits them to remote printers or print servers or sends them to email destinations.
6. PSF selects data sets from the JES spool and prints them on AFP printers. The printers can be local, TCP/IP-attached, or SNA-attached.

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 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.

Figure 6 on page 13 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.

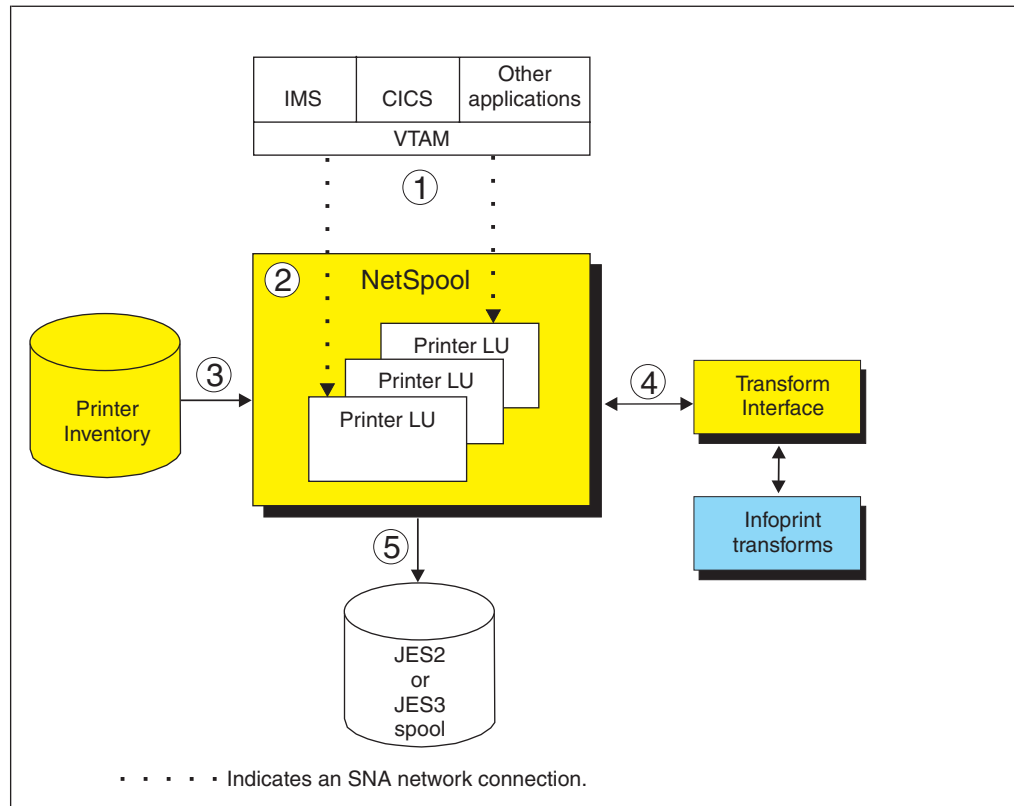


Figure 6. NetSpool system diagram

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
 - 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
2. NetSpool runs as a VTAM application on the same or different z/OS system. 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 transform SCS and 3270 data streams to EBCDIC line data or ASCII PCL data streams. Also, it can use other transform products to convert line data to other formats for printing or emailing.
5. NetSpool dynamically allocates output data sets on the JES2 or JES3 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 AFP printers.

- Distribution information, such as name and address, which can be printed on output header pages.

Additional functions that NetSpool provides are:

- **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. Line data can be printed on AFP printers.
- Transform SCS and 3270 data streams into PCL data streams. PCL data can be printed on network printers.

For information about how NetSpool transforms SCS and 3270 data streams to line or PCL format, see *z/OS Infoprint Server User's Guide*.

NetSpool can transform data to PDF or PostScript format using one of the transform products described in "Transforms" on page 20. (NetSpool first transforms SCS and 3270 data to line data, and then uses the transform).

- **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 extended 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 at the same time. Also, you can print output and send it to email destinations at the same time. 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 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. 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.

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 email 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.

IP PrintWay basic mode and IP PrintWay extended mode use the same printer definitions in the Printer Inventory.

IP PrintWay can use one of these protocols 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.
- **Email:** IP PrintWay can use the z/OS UNIX sendmail function to send print output to one or more email addresses. IP PrintWay can optionally transform the output to another format using a separate transform product. It can send the output as an email attachment or include text and line data inline in the body of the email. You can specify the recipients of the email in a header in the document, in JCL parameters on the OUTPUT JCL statement, in Infoprint Server job attributes, or in the printer definition. You can also specify a standard greeting in the printer definition.

Tip: IP PrintWay basic mode cannot transmit data sets larger than 2 gigabytes to a remote LPD. Also, depending on the IP PrintWay options the administrator selects in the printer definition, IP PrintWay basic mode might not be able to transmit data sets larger than 2 gigabytes to a remote printer that uses the IPP or direct-sockets printing protocol, or to email destinations. This restriction does not apply to IP PrintWay extended mode.

Figure 7 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 email destination and then deletes them from the JES spool. An explanation of each step follows.

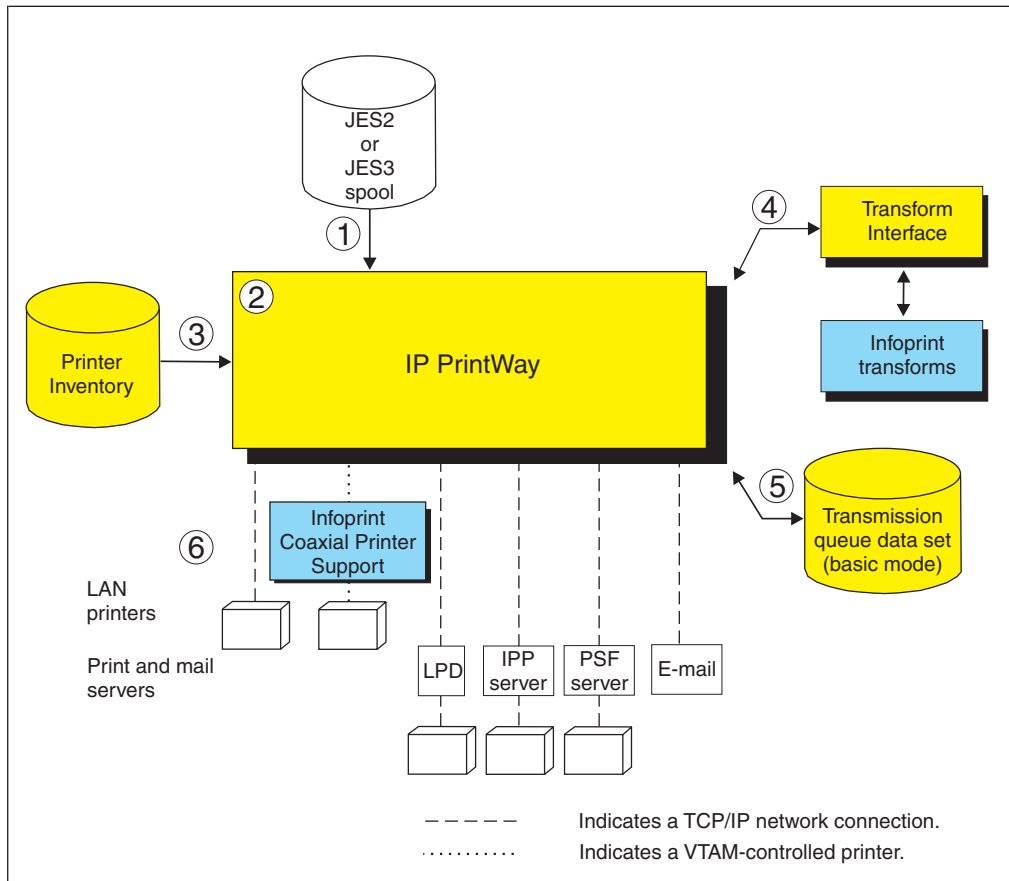


Figure 7. IP PrintWay system diagram

1. IP PrintWay selects output data sets from the JES2 or JES3 spool. IP PrintWay can select output 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 JES2 or JES3 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.
In a JES2 environment, you can configure IP PrintWay to select output data sets from the spool of either the JES2 primary subsystem or a JES2 secondary subsystem.
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 email destination. IP PrintWay basic mode can transform data by resubmitting the data stream to Print Interface. To do transforms, IP PrintWay uses transforms that other 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 email destination using the protocol selected in the printer definition: LPR, direct sockets, IPP, email, 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.

Additional functions that IP PrintWay provides include:

- **Retry**

IP PrintWay can retry an unsuccessful transmission for 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 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 (PJP) 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 emailing without using a separate transform product.

IP PrintWay can transform data to PCL, PDF, or PostScript format for printing or emailing using one of the transform products described in "Transforms" on page 20.

- **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 is processing or has retained on the JES spool.

- **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 for the administrator 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 records additional information in the type 6 record. For example, IP PrintWay extended mode includes accounting information from the JOB JCL statement and the number of pages that printed successfully if the printer supports Hewlett Packard (HP) Printer Job Language (PJP) and the direct sockets printing protocol. The administrator must request in the printer definition that IP PrintWay extended mode record the number of pages.

- **Exits and filters**

IP PrintWay lets you write exits to customize IP PrintWay processing. For example, exits can change the IP address of the remote printer, add separator pages, specify printer commands in document headers and trailers, and modify SMF accounting records. 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. Although the data sets are a single file, each data set starts printing on a new page.

- **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. The administrator

can specify printer commands in the printer definition or in the Begin Data Set exit. The Begin Data Set exit lets you specify different printer commands for each data set.

- **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.

Transform Interface

The Transform Interface component of Infoprint Server communicates with transforms that 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. For a list of the transform products that Transform Interface supports, see “Transforms” on page 20.

Figure 8 shows how Transform Interface fits into your system. An explanation of each step follows.

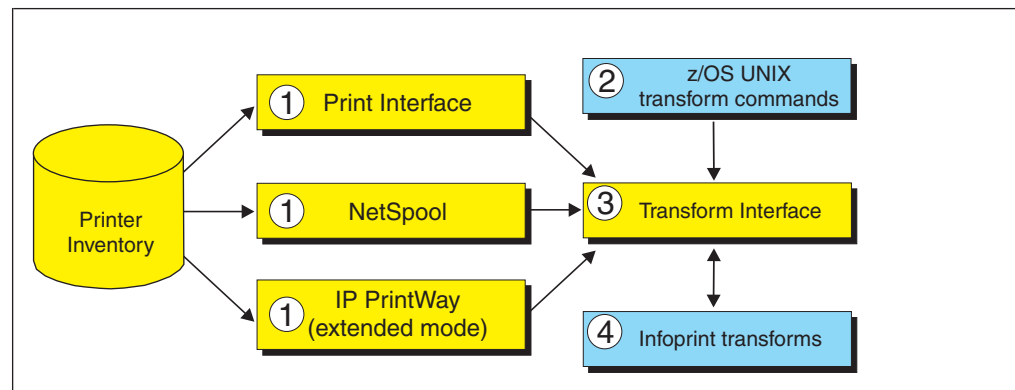


Figure 8. Transform Interface system diagram

1. 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 other transform products provide these transform filters:
 - Some transform products provide transform filters. For example, Infoprint Transforms to AFP for z/OS provides the **pdf2afp.dll** filter, which transforms data from PDF to AFP.
 - **aoprxf.so**: This filter sends data to Ricoh InfoPrint Transform Manager for Linux to be transformed.
 - **aoprform.dll**: This filter sends data to Ricoh InfoPrint Manager for AIX or Windows for transform.

Tip: IP PrintWay basic mode transforms data by resubmitting it to the Print Interface component. This function is called the *resubmit for filtering* function.
2. z/OS job submitters can specify one of these z/OS UNIX commands to transform data without printing it:

- A transform command: IBM transform products provide z/OS UNIX transform commands (such as, **pdf2afp**).
 - **remotexf** command: This command sends data to Ricoh InfoPrint Transform Manager for Linux for transform.
3. 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.
 - If the transform runs on a non-z/OS system, the Infoprint Server Transform Interface sends the data to the transform.
 4. The transform converts the data from one format to another.

Transforms

IBM provides several separately priced transform products that convert data from one format to another. These products run on the z/OS system. Ricoh provides other transform products that run on other systems.

Table 4 lists the transform products that you can use with Infoprint Server, the transforms that each product provides, and where you can find more information about the transforms and how to customize them.

Table 4. Transforms that work with Infoprint Server

Transform product	Transform	For information, see:
IBM Infoprint Transforms to AFP for z/OS (5655-N60)	PCL to AFP PDF to AFP PostScript to AFP SAP to AFP	<i>IBM Infoprint Transforms to AFP for z/OS</i>
IBM Print Transform from AFP to PCL for Infoprint Server for z/OS (5655-TF2)	AFP to PCL	<i>IBM Print Transforms from AFP for Infoprint Server for z/OS</i>
IBM Print Transform from AFP to PDF for Infoprint Server for z/OS (5655-TF1)	AFP to PDF	<i>IBM Print Transforms from AFP for Infoprint Server for z/OS</i>
IBM Print Transform from AFP to PostScript for Infoprint Server for z/OS (5655-TF3)	AFP to PostScript	<i>IBM Print Transforms from AFP for Infoprint Server for z/OS</i>
IBM Infoprint XT for z/OS (5655-O15)	Xerox to AFP	<i>IBM Infoprint XT for z/OS</i>
Ricoh InfoPrint Transform Manager for Linux	AFP to PDF GIF to AFP JPEG to AFP PCL to AFP PDF to AFP PostScript to AFP TIFF to AFP	<i>InfoPrint Transform Manager for Linux</i> <i>InfoPrint Transform Manager for Linux: afp2pdf Transform Installation and User's Guide</i>
Ricoh InfoPrint Manager for AIX	PCL to AFP PDF to AFP PostScript to AFP	<i>InfoPrint Manager for AIX: Procedures</i>
Ricoh InfoPrint Manager for Windows	PCL to AFP PDF to AFP PostScript to AFP	<i>InfoPrint Manager for Windows: Procedures</i>

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.

With Infoprint Central you can:

- **Work with print jobs:** You can work with print jobs on the JES spool. Plus, you can see more information about print jobs that Infoprint Server or PSF V4R4 and later has processed. For example, you can see whether a 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 in the common message log for the print job.

Tip: A print job consist of one or more output data sets submitted together in the same batch job. In JES2, a print job is called an *output group*.

- **Work with printers:** You can work with printers that are defined in the Printer Inventory – including AFP printers that PSF for z/OS 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 also work with PSF AFP Download Plus senders as if they were PSF printers. An AFP Download Plus sender sends print jobs to a receiver on another system for printing, emailing, or faxing.

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. 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, see all messages from PSF V4R4 and later for that one printer, and link to the printer's web page. If the PSF printer is an AFP Download Plus sender, you can do a limited set of PSF printer actions.
- 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.

You can find LUs by name. After you find a NetSpool LU, 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. Also, you can 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. Also, you can start and stop IP PrintWay job selection rules to change which print jobs IP PrintWay extended mode 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 do 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.

Tip: 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.

SNMP subagent

The Infoprint Server 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.

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

You can monitor PSF printers that do not contain internal SNMP agents, such as the InfoPrint 3900 printer, and also PSF printers that have internal SNMP agents but are not TCP/IP-attached to PSF. You can also monitor PSF TCP/IP-attached printers that contain internal SNMP agents. However, consider defining PSF printers that have internal SNMP agents directly to the SNMP manager. When you define a printer directly to the SNMP manager, you can also monitor printer statistics and change some printer characteristics. See the documentation for your printers to determine if they have internal SNMP agents.

Figure 9 on page 23 shows how the SNMP subagent fits into your system. An explanation of each step follows.

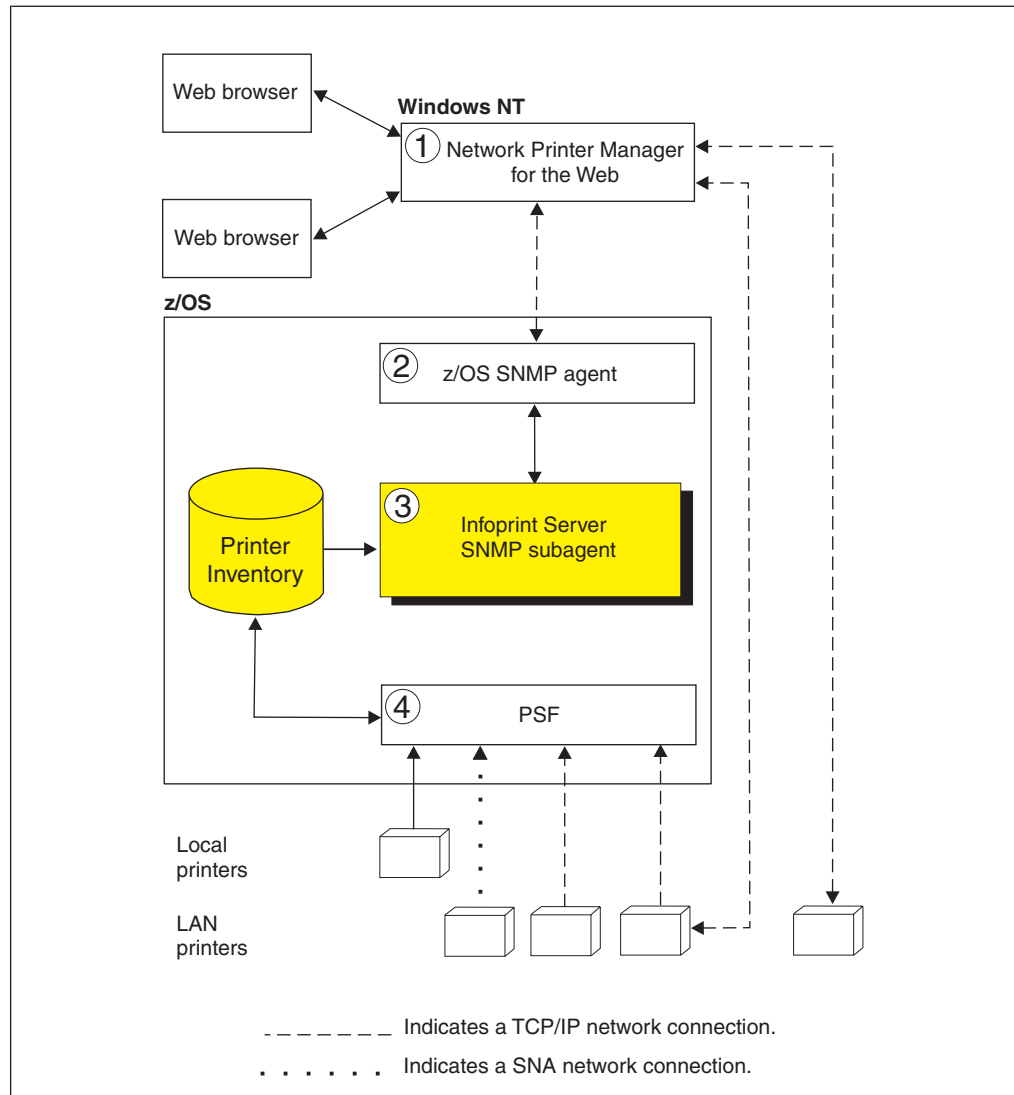


Figure 9. SNMP subagent system diagram

1. To monitor PSF printers, you must install an SNMP manager. You can install any SNMP manager that communicates with an SNMP agent that supports the general printer MIB (defined in RFC 1759) and has implemented support for multiple printers defined with one IP address.

One SNMP manager that provides the required support is Ricoh Network Printer Manager (NPM) for the web, which is shown in Figure 9.

Administrators can use a Java-enabled version of Microsoft Internet Explorer to monitor PSF printers, while the NPM server runs on a Windows system. Note that NPM limits the number of PSF printers an administrator can monitor at the same time. See the NPM online help for the maximum number of printers NPM lets you monitor.

As shown in the figure, the SNMP manager (NPM) can also communicate directly with any TCP/IP-attached printer that contains an internal SNMP agent. PSF can control this printer, but it is not necessary.

2. The SNMP agent, part of the z/OS Communications Server, communicates with the SNMP manager and with the Infoprint Server SNMP subagent.
3. The Infoprint Server SNMP subagent communicates with PSF through the Printer Inventory. PSF stores printer characteristics and printer status in the

Printer Inventory for any printer that has SNMP-reporting enabled. (The administrator enables SNMP reporting in the FSA definition for the PSF printer).

The SNMP subagent transmits the information stored by PSF to the SNMP manager through the z/OS SNMP agent. The SNMP subagent also notifies the SNMP manager immediately when PSF detects a change to printer characteristics or printer status.

4. PSF obtains printer characteristics and printer status from any PSF-controlled printer. The PSF printer can be channel-attached, TCP/IP-attached, or SNA-attached. However, the PSF printer cannot be attached using the Download for z/OS feature of PSF.

If a PSF-controlled printer is shared with another printing application, and is not connected to PSF when an intervention required situation occurs, PSF cannot report the change in printer status.

Chapter 2. Customization roadmap

This information can help you determine which Infoprint Server components you must customize to use the different functions that Infoprint Server provides. Table 5 lists the functions that Infoprint Server provides, the Infoprint Server components you need to customize, and the sections that describe the customization tasks.

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.

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.

Table 5. Summary of Infoprint Server components

Infoprint Server function	Components	See page	
Receive print requests from these sources, and create output data sets on the JES spool: <ul style="list-style-type: none"> • 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 lp, lpstat, and cancel commands • The AOPPRINT JCL procedure • Any Windows application that supports printing • The Infoprint Server Application Programming Interface • Batch z/OS job that specify the Print Interface subsystem on a DD statement 	Printer Inventory Manager	27	
	Print Interface	115	
	Infoprint Port Monitor for Windows (optional)	369	
Receive print requests from VTAM applications (such as CICS and IMS), and create output data sets on the JES spool	Printer Inventory Manager	27	
	NetSpool	171	
Send output data sets from the JES spool to an ASCII printer, a print server, or an email destination	Printer Inventory Manager	27	
	IP PrintWay, basic or extended mode	199	
	Print Interface (if you use IP PrintWay basic mode and transforms)	115	
Specify PSF and AFP Download Plus configuration information in FSS and FSA definitions in the Printer Inventory; write PSF and AFP Download messages in the Infoprint Server common message log	Printer Inventory Manager	27	
	PSF for z/OS	379	
Transform data from one format to another	Printer Inventory Manager	27	
	Transform Interface	157	
Use Infoprint Central	Printer Inventory Manager	27	
	Infoprint Central	327	
View status of PSF printers using an SNMP manager	Printer Inventory Manager	27	
	SNMP subagent	373	

Chapter 3. Customizing the Printer Inventory Manager

This information describes how to customize the Printer Inventory Manager component of Infoprint Server. You must customize the Printer Inventory Manager to use Infoprint Server.

The Printer Inventory Manager controls the Printer Inventory. It provides Infoprint Server ISPF panels and the Printer Inventory Definition Utility program (PIDU), which the administrator can use to create and manage entries in the Printer Inventory. The Printer Inventory Manager also controls the Infoprint Server common message log and Historical Inventory.

For an overview of the Printer Inventory Manager and how this component fits into your system, see “Printer Inventory Manager” on page 3.

Note: The Printer Inventory *cannot* be shared by any Infoprint Server components running at the *same* or *different* levels on other systems. For sysplex considerations, see “/var/Printsrv directory” on page 31.

Table 6 summarizes the tasks to customize the Printer Inventory Manager. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies to your installation.

Table 6. Customization tasks for Printer Inventory Manager

Task	Condition	See page
Identifying software requirements for Printer Inventory Manager	Required	28
Enabling the Infoprint Server feature	Optional: If the Infoprint Server feature was not in your original z/OS order	28
Updating the BPXPRMxx member of SYS1.PARMLIB	Required	28
Making Language Environment and C++ runtime libraries available	Optional: If required libraries are not in LNKLST	29
Creating Infoprint Server directories	Required	30
Creating the Infoprint Server configuration file (aopd.conf)	Required	34
Setting environment variables for the Printer Inventory Manager	Required	43
Setting up security for the Printer Inventory and operator commands	Required	47
Setting permissions for directories and executable files -- aopsetup	Required	54
Customizing startup and shutdown commands and procedures	Optional: To specify environment variables or to start and stop Infoprint Server with JCL procedures	56
Enabling Infoprint Server ISPF panels	Required	66

Table 6. Customization tasks for Printer Inventory Manager (continued)

Task	Condition	See page
Enabling dynamic configuration	Optional: To customize Infoprint Server without restarting Infoprint Server	68
Editing the system configuration definition for Printer Inventory Manager	Optional: To specify dynamic attributes	76
Sending non-console messages to the hardcopy log	Optional: To send messages to the hardcopy log	84
Setting up the MVS system logger for the common message log	Optional: To use the MVS system logger for the common message log	88
Backing up and restoring the Printer Inventory	Required	105
Saving messages in the common message log in the z/OS UNIX file system	Optional: To save the common message log	110
Reducing the size of the common message log in the z/OS UNIX file system	Optional: To free space in the z/OS UNIX file system	111
Setting Workload Manager goals	Optional: To prioritize Infoprint Server work	112

Related task: Some customization tasks require that you start Infoprint Server daemons. For information about how to start Infoprint Server, see *z/OS Infoprint Server Operation and Administration*.

Identifying software requirements for Printer Inventory Manager

This additional IBM software beyond the z/OS base elements is required:

- To print output from Infoprint Server on printers that PSF controls, PSF for z/OS (5655-M32)
- z/OS Security Server Resource Access Control Facility (RACF) or another access control manager with equivalent function

Enabling the Infoprint Server feature

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 5650-ZOS. For information about dynamic enablement, see *z/OS Planning for Installation*.

Updating the BPXPRMxx member of SYS1.PARMLIB

Infoprint Server uses z/OS UNIX domain sockets. The BPXPRMxx member of SYS1.PARMLIB must define an AF_UNIX file system, for example:

```
FILESYSTYPE TYPE(UDS) ENTRYPPOINT(BPXTUINT)
NETWORK DOMAINNAME(AF_UNIX)
          DOMAINNUMBER(1)
          TYPE(UDS)
```

For AF_UNIX file systems, the MAXSOCKETS parameter is ignored if specified. The maximum number of AF_UNIX sockets is 10000.

Related information: For more information about defining AF_UNIX file systems, see *z/OS UNIX System Services Planning*.

Making Language Environment and C++ runtime libraries available

These Language Environment® and C++ runtime libraries must be available to Infoprint Server and IBM transforms:

- *hlq.SCEERUN*
- *hlq.SCEERUN2*
- *hlq.SCLBDLL*

The *hlq.SCEERUN* and *hlq.SCLBDLL* data sets must be APF-authorized.

Steps for making runtime libraries available:

1. Add the *hlq.SCEERUN*, *hlq.SCEERUN2*, and *hlq.SCLBDLL* libraries to the system LNKLST. Also, consider placing the *hlq.SCEELPA* data set (which contains key modules) in the link pack area (LPA) for better performance.

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

2. If you do not add the *hlq.SCEERUN* and *hlq.SCLBDLL* libraries to LNKLST, do these steps:
 - Add the *hlq.SCEERUN* and *hlq.SCLBDLL* libraries to the STEPLIBLIST facility of the BPXPRMxx *hlq.PARMLIB* member.
 - Add this statement to the */etc/rc* file:

```
export STEPLIB=hlq.SCEERUN:hlq.SCLBDLL
```

For example, if you use the standard high-level qualifiers, add this statement:

```
export STEPLIB=CEE.SCEERUN:CBC.SCLBDLL
```

- Remove these statements from the */etc/profile* file:

```
if [-z "$STEPLIB"] && tty -s;  
then  
  export STEPLIB=none  
  exec sh -L  
fi
```

Replace the statements that you removed in the */etc/profile* file with this statement:

```
export STEPLIB=hlq.SCEERUN:hlq.SCLBDLL
```

For example, if you use the standard high-level qualifiers, add this statement:

```
export STEPLIB=CEE.SCEERUN:CBC.SCLBDLL
```

- Uncomment and edit this statement in the **aopstart** EXEC:

```
/*n=n+1;env.n='STEPLIB=hlq.SCEERUN:hlq.SCLBDLL' */
```

For example, if you use the standard high-level qualifiers, code this statement:

```
n=n+1;env.n='STEPLIB=CEE.SCEERUN:CBC.SCLBDLL'
```

For information, see “Editing the aopstart EXEC” on page 57.

- Add the *hlq.SCEERUN* and *hlq.SCLBDLL* libraries to the TSO logon procedures for administrators who use Infoprint Server ISPF panels. For information, see “Defining libraries in the TSO logon procedure” on page 66.
- Add the *hlq.SCEERUN* and *SCLBDLL* libraries to JCL that starts AOPBATCH. The AOPSTART and AOPPRINT JCL procedures use AOPBATCH.
- Add the *hlq.SCEERUN* and *SCLBDLL* libraries to the NetSpool and IP PrintWay basic mode startup procedures. For more information, see “Customizing the NetSpool startup procedure” on page 174 and “Customizing the IP PrintWay startup procedure (basic mode)” on page 320.

Related information: For more information about how to access the Language Environment runtime libraries and improve the performance of runtime libraries, see *z/OS UNIX System Services Planning*.

Creating Infoprint Server directories

The Printer Inventory Manager uses these directories, which can be in a Hierarchical File System (HFS) or a z/OS File System (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 Printer Inventory files and other Infoprint Server files

Related information: For information about zFS, see *z/OS Distributed File Service zFS Administration*.

/etc/Printsrv directory

The **/etc/Printsrv** directory is the default location for all Infoprint Server configuration files. This directory is created automatically when you install Infoprint Server.

The **/etc/Printsrv** directory is created with these owners and permissions:

- Owned by UID of 0
- Readable and executable by everyone
- Writeable only by users with an effective UID of 0

The permission bits, in octal, are 755.

If desired, you can change the group owner to the RACF group you create for Infoprint Server administrators (AOPADMIN) and change the permissions to allow the AOPADMIN group to write files in the directory. This lets users who are members of the AOPADMIN group create configuration files.

Related task: If you create Infoprint Server configuration files in a directory other than the **/etc/Printsrv** directory, specify the location of the configuration files in these environment variables in the **aopstart** EXEC:

- AOPCONF
- AOPMSG_CONF
- AOPRXF_CONF
- AOPSAPD_CONF

- AOPXFD_CONF

Related information: For a summary of all environment variables, see Appendix C, “Environment variables,” on page 409.

/var/Printsrv directory

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**).

Tip: The **aopsetup** shell script creates the **/var/Printsrv** directory (or the directory that is named in the **base-directory** attribute) and sets the appropriate permissions for it. For information about how to run **aopsetup** in a later step, see “Setting permissions for directories and executable files -- aopsetup” on page 54.

Guidelines:

- Mount a separate file system at the **/var/Printsrv** mount point because some subdirectories can become large. If your system is part of a sysplex, the **/var/Printsrv** file system must be system-specific and designated NOAUTOMOVE in the BPXPRMxx parmlib member.

Example: This example shows a sample BPXPRMxx member in SYS1.PARMLIB:

```
MOUNT FILESYSTEM('my.var.printsrv.filesystem')
      TYPE(ZFS)
      MODE(RDWR)
      NOAUTOMOVE
      MOUNTPOINT('/&SYSNAME./var/Printsrv')
```

If you specify a zFS, it must be HFS-compatible. For information about zFS, see *z/OS Distributed File Service zFS Administration*.

- Make sure that the file system that contains the **/var/Printsrv** directory has enough space. For information about how much space is required, see “Calculating DASD space requirements for the **/var/Printsrv** directory” on page 32.
- 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
 - Executable by everyone

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 consists of files in the **/var/Printsrv** directory with file extension **.v2db**. For example, these Printer Inventory files reside in the **/var/Printsrv** directory:

- **master.v2db**
- **jestoken.v2db**
- **pwjestoken.v2db**

Other Printer Inventory files with the **.v2db** extension reside in **/var/Printsrv** subdirectories.

Note: Printer Inventory files with extension `.v2db` are in Version 2 format. Printer Inventory files with extension `.db` are in Version 1 format. If you migrated to Infoprint Server V2R1 from an earlier version, the `/var/Printsrv` directory might contain Printer Inventory both Version 1 and Version 2 files. Infoprint Server V2R1 does not read or update Printer Inventory Version 1 files. You can delete the Version 1 files (extension `.db`).

Calculating DASD space requirements for the `/var/Printsrv` directory

The amount of space that is required for the file system at the `/var/Printsrv` mount point depends on various factors, such as the number of printers you have, the size of data sets you process, whether you use data-stream transforms, the number of days worth of historical information you keep, and (if you do not use the MVS system logger for messages) the number of days worth of messages you keep. Use the information in this section to estimate how much space you need to allocate for the file system. Also, other factors can influence how much space is required.

Tip: The Printer Inventory, Historical Inventory, search database, and message log database are optimized for rapid direct access to objects. As you add objects to these databases, files in the databases increase in size. When you remove objects, the files do not decrease in size because the Printer Inventory Manager designates as available the space that had been occupied by the removed objects. When you add objects in the future, the Printer Inventory Manager uses available space in the files. The files increase in size only when they do not contain sufficient available storage. So, the size of files can be characterized as a high-water mark.

Guideline: Allocate at least 750 MB of DASD space for the file system that is 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 large files or if you want to keep many days worth of messages and historical information, you might need more space.

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

- **`/var/Printsrv`:** This directory contains:
 - Printer Inventory files `master.v2db`, `jestoken.v2db`, and `pwjestoken.v2db`.
 - 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)

Tip: To save space in this directory:

- Configure clients that send data to the Print Interface LPD to send the control file first. Not all clients can be configured to do so.
 - Delete Language Environment dumps in this directory that begin with “CEEDUMP” after you provide them to IBM support if necessary. These dumps can be large.
- **`/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.

To calculate the maximum size of this directory, multiply the size of the largest possible output data set by the number of output data sets in a job. Multiply the result by the number of printers that can be printing at the same time. Add 10 MB.

Tip: To save space in this directory, print very large data sets (such as 4 GB or larger) on one or two printers so that large data sets do not print at the same time.

- **/var/Printsrv/logs** and **/var/Printsrv/logdb:** These directories contain message logs and the common message log database.

The number of days that are specified in the **log-retention** attribute indirectly determines the size of these directories. For every 1000 data sets processed in the specified number of days, allow 3 MB in the **logs** directory and 2 MB in the **logdb** directory.

Note: If you use the MVS system logger for the common message log, messages are not stored in the **logs** subdirectory. However, data is still stored in the **logdb** subdirectory.

Tips:

- Because the common message log can contain a large amount of data, start with a value of 1 day in the **log-retention** attribute. Increase the number of days only if there is enough space in the file system to keep more historical print job information and messages.
- If the size of the message files grows too large, you can decrease the size. However, if you do this, Infoprint Central cannot display messages that were written to the log before you decreased its size. For more information, see “Reducing the size of the common message log in the z/OS UNIX file system” on page 111.

- **/var/Printsrv/hinv:** This directory contains the Historical Inventory. The default maximum size is 10 MB. To change the maximum size, specify the **max-historical-inventory-size** attribute.
- **/var/Printsrv/searchdb:** This directory contains the search database. Allow five times the amount of space as the maximum for the **/var/Printsrv/hinv** directory. Also, allow an extra 12 K for each IP PrintWay and PSF printer.
- **/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.

Tip: To save space in this directory:

- Specify an alternate trace directory in the AOPTRACEDIR environment variable before you activate tracing.
- Erase trace files after the problem is solved.

- **/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.

Creating the Infoprint Server configuration file (aopd.conf)

The Infoprint Server configuration file (**aopd.conf**) is a required file. You must create it before starting Infoprint Server for the first time. The default location of this file is **/etc/Printsrv/aopd.conf**.

If you change the **aopd.conf** file while Infoprint Server is running, in most cases you must stop all Infoprint Server daemons and restart Infoprint Server before the changes can take effect. An exception is if you change the **start-daemons** attribute. For more information, see the description of the **start-daemons** attribute.

To configure the Printer Inventory Manager, you can specify these attributes in the **aopd.conf** file:

base-directory = *path*

The directory path in which Infoprint Server creates Printer Inventory files and other files. This attribute is required.

The directory path can contain up to 95 characters. However, if you use Infoprint Server transforms, the maximum length of the directory path might be less, depending on the length of the transform class names you define. If the transform class names can be up to 20 characters, the maximum length of the directory path is 60 characters.

Rules:

- Do not change this attribute while Infoprint Server (including NetSpool and IP PrintWay basic mode) is running. Also, if PSF for z/OS uses the Printer Inventory, do not change this attribute while PSF is running.
- If you use Infoprint Central and change this attribute, restart the z/OS HTTP Server.

Default: None.

Example: To use the suggested name for the base directory:

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

dynamic-configuration = **yes** | **no**

Indicates whether the dynamic configuration function is enabled. If dynamic configuration is enabled, you can change many configuration attributes in the system configuration definition in the Printer Inventory while Infoprint Server is running, without stopping and restarting Infoprint Server.

yes Dynamic configuration is enabled.

no Dynamic configuration is not enabled.

Tip: For more information about dynamic configuration, see “Enabling dynamic configuration” on page 68.

Rules:

- If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.
- If you use Infoprint Central and change this attribute, restart the z/OS HTTP Server.

Default: `dynamic-configuration = no`

Example: `dynamic-configuration = yes`

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. It *cannot* be a lowercase letter or numeric. The remaining characters can be uppercase letters or numeric. 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 and AFP Download Plus startup procedures if you want PSF or AFP Download Plus to use information that is specified in FSS and FSA definitions in the Infoprint Server Printer Inventory.

Rules:

- Do not change this attribute while Infoprint Server (including NetSpool and IP PrintWay basic mode) is running. Also, if PSF for z/OS uses the Printer Inventory, do not change this attribute while PSF is running.
- If you use Infoprint Central and change this attribute, restart the z/OS HTTP Server.

Default: inventory = AOP1

jes-name = *name*

The 1-4 character name of the JES2 secondary subsystem that Infoprint Server works with. If this attribute is specified, Print Interface and NetSpool create output data sets on the spool of the specified subsystem, and IP PrintWay extended mode selects output data sets for printing from the spool of the specified subsystem. Infoprint Server converts any lowercase letters to uppercase.

Tip: If you want Infoprint Server to work with the JES2 or JES3 primary subsystem, you can either omit this attribute or specify the name of the primary subsystem.

Rules:

- If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.
- If you use Infoprint Central and change this attribute, restart the z/OS HTTP Server.

Default: None. Infoprint Server works with the JES2 or JES3 primary subsystem.

Example: jes-name = JESX

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:

ippd The IPP Server daemon, **aopippd**, starts. It processes print jobs that are submitted by an Internet Printing Protocol (IPP) client.

lpd The line printer daemon, **aoplpd**, starts. It processes print jobs that are submitted with commands, such as **lpr**, and by the Infoprint Port Monitor for Windows.

netd	The NetSpool daemon, aopnetd , starts. It supports printing from VTAM applications such as CICS and IMS.
outd	The IP PrintWay extended mode daemons, aopoutd and aopwsmd , start. They select output data sets from the JES spool and send them to remote printers in a TCP/IP network or to email destinations.
snmpd	The Simple Network Management Protocol (SNMP) subagent daemon, aopsnmpd , starts. It provides status information to the z/OS SNMP agent about printers controlled by PSF.
ssid	The Infoprint Central daemon, aopssid , starts. It communicates between Infoprint Central and JES.
subd	The Print Interface subsystem daemon, aopsubd , starts. It processes output data sets when the DD JCL statement requests the subsystem on the SUBSYS parameter.
xfd	The Transform Manager, aopxfd , starts. It manages data stream transforms.

The 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.

Rule: If you add or remove a value from **start-daemons** while Infoprint Server is running, you must start Infoprint Server again. You do not need to stop any daemons before you restart Infoprint Server unless you add value **ssid** (see “Note”).

When you restart Infoprint Server:

- Daemons that were removed from **start-daemons** are stopped.
- Daemons that were added to **start-daemons** are started.
- Daemons that were not removed or added continue to run without interruption.

To run the AOPSTART JCL procedure to start daemons that you added to the **start-daemons** attribute, enter this MVS START command:

```
START AOPSTART
```

Note: Stop all Infoprint Server daemons before you restart Infoprint Server if *all* of these conditions are met:

- You added value **ssid**.
- Dynamic configuration is not enabled.
- You did not specify the **log-retention** attribute or you specified **log-retention=0** the last time you started all Infoprint Server daemons.

Before you restart Infoprint Server, if **log-retention=0** is specified, change the value to 1 or higher.

For example, enter these MVS commands to run the AOPSTART and AOPSTOP JCL procedures:

```
START AOPSTOP
START AOPSTART
```

Default: `start-daemons = {1pd}`

Examples:

- In this example, all daemons start:
start-daemons = {ippd lpd netd outd snmpd ssid subd xfd}
- In this example, only the Printer Inventory Manager daemons start:
start-daemons = {}

xcf-group-qualifier = *q*

A one-character XCF group qualifier that Infoprint Server adds to the end of the XCF group name to make it unique in a sysplex. The character must be an alphanumeric or national (\$, #, and @) character. Enclose a national character in single or double quotation marks. A lowercase letter is converted to uppercase.

You must specify a qualifier only if you enable dynamic configuration and multiple instances of Infoprint Server run in a sysplex with the same Printer Inventory name. Keep in mind that in the XCF group name, the Printer Inventory name is always uppercase. Therefore, you need to specify a qualifier if the Printer Inventory names are the same except for case. For example, specify a qualifier if the Printer Inventories are named "AOP1" and "aop1".

As an alternative, you can change the Printer Inventory name so that it is unique for each instance of Infoprint Server running in the sysplex. (You specify the Printer Inventory name in the **inventory** attribute.)

Format of the XCF group name: AOP*pppp*[*q*]:

AOP Infoprint Server identifier.

pppp Printer Inventory name. Any lowercase letters are converted to uppercase.

q A one-character XCF group qualifier if specified in **xcf-group-qualifier**. This qualifier is optional.

Rule: If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: None

Example: xcf-group-qualifier = 1

If dynamic configuration is not enabled (**dynamic-configuration=no**), you can also specify these attributes in the **aopd.conf** file:

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*.

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
- 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. For information about how to create conversion tables, see information about code set conversion utilities in *z/OS XL C/C++ Programming Guide*.

Infoprint Server uses the code page specified in this field in these ways:

- The Printer Inventory Manager displays this code page in the **Printer code page** field in ISPF panels for new IP PrintWay printer definitions that use the LPR, direct sockets, or IPP protocol to transmit data sets to the remote system.
- Print Interface uses this code page as the default source (document) code page when it converts data streams from ASCII to EBCDIC. A document code page specified in the printer definition or job attribute overrides this value.
- Print Interface uses this code page as the target (printer) code page when it converts PostScript documents that were created with a non-ASCII code page.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: `ascii-codepage = IS08859-1`

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*.

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
- 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. For information about how to create conversion tables, see information about code set conversion utilities in *z/OS XL C/C++ Programming Guide*.

Infoprint Server uses this code page in these ways:

- The Printer Inventory Manager displays this code page in the **Printer code page** field in ISPF panels for new PSF printer definitions, general printer definitions, and IP PrintWay printer definitions that use the VTAM or email protocol to transmit data sets to the remote system.
- NetSpool uses this code page as the default source (document) code page when it converts data streams from EBCDIC to ASCII. A document code page in the printer definition or job attribute overrides this value.
- IP PrintWay extended mode uses this code page as the default source (document) code page when it converts data streams from EBCDIC to ASCII. A document code page in the printer definition overrides this value.
- Print Interface uses this code page as the source (document) code page when Print Interface cannot determine the user locale for the **lp** command.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server. Also restart the NetSpool started task.

Default: ebcdic-codepage = IBM-1047

log-retention = days

The number of days worth of messages that Infoprint Server retains in the common message log. Also, this attribute specifies the number of days for which Infoprint Central displays information about print jobs that have been removed from the JES spool.

You can specify a value from 0–59. A value of 0 means that Infoprint Server does not write any messages to the common message log and Infoprint Central does not display information about print jobs that have been removed from the JES spool.

Guidelines:

- If you do not start the Infoprint Central daemon, specify a value greater than 0 so that Infoprint Server writes messages to the common message log.
- If you specify a value greater than 0 or start the Infoprint Server daemon, you might need to increase the amount of space available to the file system mounted at the **/var** or **/var/Printsrv** mount point because the common message log and Historical Inventory can contain a large amount of data. See “Calculating DASD space requirements for the **/var/Printsrv** directory” on page 32.
- Because the common message log and Historical Inventory can contain a large amount of data, start with a value of 1 day. Increase the value by 1 day if there is enough space in the file system and if users request to see more messages or historical information.
- If the size of the common message log in the z/OS UNIX file system grows too large, you can decrease its size. For more information, see “Reducing the size of the common message log in the z/OS UNIX file system” on page 111.
- If you use Infoprint Central to view messages, set the TZ and either the LC_TIME or LC_ALL environment variables. These variables affect the date and time displayed in messages. See “Setting environment variables for the Printer Inventory Manager” on page 43.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: If you start the Infoprint Central daemon (**aopssid**), the default is 1 day. Otherwise, the default is 0. If you enable dynamic configuration, the default changes to 1 day regardless of whether you start **aopssid**.

Example: log-retention = 3

max-historical-inventory-size = *megabytes*

The maximum size in megabytes (MBs) of the Historical Inventory, `/var/Printsrv/hinv`. The Historical Inventory contains information about print jobs and documents (output data sets) 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. However, the number of days specified in the **log-retention** attribute limits the amount of information that Infoprint Server keeps in the Historical Inventory.

Table 7 shows the number of documents one MB in the Historical Inventory can contain.

Table 7. Number of documents in the Historical Inventory

If print jobs on the JES spool contain:	One MB in the Historical Inventory contains:
1 document in each print job	1350 documents
3 documents in each print job	1750 documents
5 documents in each print job	1850 documents
10 documents in each print job	1950 documents

Guideline: Use this formula to estimate the maximum size of the Historical Inventory:

Maximum size of Historical Inventory =
(Documents processed in one day X Days to retain information) /
Documents in one MB

For example:

- Documents processed in one day = 100,000
- Days to retain information = 3 (**log-retention** = 3)
- Documents in one MB = 1350 documents

The maximum size of the Historical Inventory = (100,000 X 3) / 1350 = 223 MB (rounded up)

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: max-historical-inventory-size = 10

Example: max-historical-inventory-size = 300

Related information: For information about:

- All attributes that you can specify in the **aopd.conf** file, see “aopd.conf -- Infoprint Server configuration file” on page 389.
- Enabling dynamic configuration, see “Enabling dynamic configuration” on page 68.

Steps for creating the Infoprint Server configuration file (aopd.conf)

These steps show how to create the **aopd.conf** configuration file in the **/etc/Printsrv** directory, which is the default directory for Infoprint Server configuration files. You must create the **aopd.conf** file before starting Infoprint Server for the first time.

To edit an existing **aopd.conf** file, follow the steps in “Steps for editing the Infoprint Server configuration file (aopd.conf)” on page 42.

Before you begin: You must have permission to create files in the **/etc/Printsrv** directory. If you did not change the owner and permissions for **/etc/Printsrv** after installation, you must have an effective UID of 0 to create files in this directory.

1. Switch to an effective UID of 0:

```
su
```

Tip: To use the **su** command, you must be permitted to the BPX.SUPERUSER profile in the FACILITY class in RACF.

2. Copy the sample configuration file, **/usr/lpp/Printsrv/samples/aopd.conf**, to **/etc/Printsrv/aopd.conf**:

```
cp /usr/lpp/Printsrv/samples/aopd.conf /etc/Printsrv/aopd.conf
```

If you copy the sample configuration file to a different location, specify the full path name of the configuration file in the AOPCONF environment variable in the **aopstart** EXEC and in the **/etc/profile** file.

Tip: You can use the sample job that is provided in SYS1.SAMPLIB(AOPCPETC) to copy all Infoprint Server sample configuration files, including the Infoprint Server configuration file, from **/usr/lpp/Printsrv/samples** to the **/etc/Printsrv** directory. Review the comments in the sample job to determine which configuration files you require, and then run the job to create the configuration files. IBM suggests that a user with an effective UID of 0 run this job.

3. Change the owner of the file to UID 0:

```
chown 0 /etc/Printsrv/aopd.conf
```

4. Change the group owner to AOPADMIN. AOPADMIN is the suggested RACF group name for Infoprint Server administrators. However, you can use a different name for this group (see “AOPADMIN and AOOPER groups” on page 49).

```
chgrp AOPADMIN /etc/Printsrv/aopd.conf
```

5. Change the permissions of the file so that it is readable by everyone and writeable only by UID 0 and members of the AOPADMIN group.

```
chmod 664 /etc/Printsrv/aopd.conf
```

6. Edit the configuration file to specify configuration attributes and save your changes. For example, enter:

```
oedit /etc/Printsrv/aopd.conf
```

Edit this file using the IBM-1047 code page. For the syntax rules for editing the file, see “Syntax rules”.

Tip: You can select the ISPF “z/OS UNIX Shell” option (ISPF 13.11) to edit z/OS UNIX files.

-
7. If you switched to an effective UID of 0 in step 1, switch back to your own UID:
exit
-

Syntax rules: When you edit this file, follow these syntax rules:

- Use lowercase characters for the attributes and use uppercase or lowercase characters for the values. Use blank characters before or after the equal sign, if desired.
- If a value contains blank characters or special characters (such as {}-> =, # &), enclose the value in single or double quotation marks.
- Start comments with the # symbol.
- Include blank lines, if desired.

Steps for editing the Infoprint Server configuration file (aopd.conf)

These steps show the z/OS UNIX commands to use to edit an existing **aopd.conf** configuration file.

Note: To create the **aopd.conf** file, follow the steps in “Steps for creating the Infoprint Server configuration file (aopd.conf)” on page 41.

Before you begin: You must be a member of the RACF group for Infoprint Server administrators (AOPADMIN) or have an effective UID of 0.

1. Edit the configuration file to specify configuration attributes. Use your preferred editor and save your changes. For example, enter:
oedit /etc/Printsrv/aopd.conf

Edit this file using the IBM-1047 code page. For the syntax rules for editing the file, see “Syntax rules”.

Tip: You can select the ISPF “z/OS UNIX Shell” option (ISPF 13.11) to edit z/OS UNIX files.

-
2. If you change any attributes except for **start-daemons** while Infoprint Server is running, you must stop and restart Infoprint Server before the changes take effect:
 - a. Stop NetSpool and IP PrintWay basic mode. Also, if PSF uses the Printer Inventory, stop the PSF functional subsystem (FSS).
 - b. Stop Infoprint Server daemons and wait for all daemons to stop.
Example: Enter this MVS command to run the AOPSTOP JCL procedure:
START AOPSTOP
 - c. Restart the Infoprint Server daemons.
Example: Enter this MVS command to run the AOPSTART JCL procedure:

```
START AOPSTART
```

- d. Restart NetSpool and IP PrintWay basic mode. Also, if PSF uses the Printer Inventory, restart the PSF functional subsystem (FSS).

If you changed the **inventory** attribute, be sure to also change the Printer Inventory name in the NetSpool, IP PrintWay basic mode, and PSF start procedures.

Notes:

- a. If you change only the **start-daemons** attribute, in most cases you do not need to stop Infoprint Server daemons or any other programs before starting Infoprint Server again. When you start Infoprint Server again, any daemons that you removed from **start-daemons** are stopped and any daemons that you added are started. For more information, see the description of the **start-daemons** attribute.

Example: Enter this MVS command to run the AOPSTART JCL procedure:

```
START AOPSTART
```

- b. Do not change the **base-directory** and **inventory** attributes while Infoprint Server is running.

-
3. If you add or change any of these attributes and you run Infoprint Central, restart the z/OS HTTP Server:

- base-directory
- dynamic-configuration
- inventory
- jes-name

Tip: You can restart the z/OS HTTP Server while Infoprint Server is stopped.

Syntax rules: When you edit this file, follow these syntax rules:

- Use lowercase characters for the attributes and use uppercase or lowercase characters for the values. Use blank characters before or after the equal sign, if desired.
- If a value contains blank characters or special characters (such as {}-> =, # &), enclose the value in single or double quotation marks.
- Start comments with the # symbol.
- Include blank lines, if desired.

Setting environment variables for the Printer Inventory Manager

Infoprint Server uses environment variables to define its environment in the z/OS system. For a description of environment variables, see *z/OS UNIX System Services User's Guide*.

This section describes only those environment variables that affect the Printer Inventory Manager. Other environment variables affect other components of Infoprint Server, such as Print Interface and IP PrintWay extended mode. See Appendix C, "Environment variables," on page 409 for a summary of all the environment variables that Infoprint Server uses.

The Printer Inventory Manager, as well as other Infoprint Server daemons, uses environment variables that are specified in the **aopstart** EXEC. In addition, the daemons accept certain variables from the runtime 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 runtime environment. If you use the AOPSTART JCL procedure to start Infoprint Server daemons, you must specify these runtime variables in the STDENV data definition (DD) 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 commands, such as **aoplogu** and **pidu**, accept environment variables from the **/etc/profile** file.

For information about how to edit the **aopstart** EXEC, see “Editing the aopstart EXEC” on page 57.

To edit the **/etc/profile** file, you can use 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.

Examples:

- 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
```

Table 8 summarizes the variables that affect the Printer Inventory Manager, including the common message log and the **pidu** command. This table indicates where you need to set the variable. **Required** variables must be set. **Optional** variables must be set only if the default value is not suitable for your installation.

Table 8. Where to define environment variables for the Printer Inventory Manager

Set this environment variable	In the /etc/profile file	In the aopstart EXEC	In the AOPSTART procedure STDENV DD
AOPCONF	Optional	Optional	No
AOPMSG_CONF	Optional	Optional	No
AOPVALIDATEDB	Optional	Optional	Optional
LANG	Optional	Optional	Optional
LC_ALL	Optional	Optional	Optional
LC_CTYPE	Optional	Optional	Optional
LC_TIME	Optional	Optional	Optional
LIBPATH	Required	Required	No
MANPATH	Required	No	No
NLSPATH	Required	Required	No
PATH	Required	Required	No
TZ	Optional	Optional	Optional

Tip: If you change the environment variables while any Infoprint Server daemons are running, the settings do not change for the current login shell, daemons, or other processes already started.

These environment variables affect the behavior of the Printer Inventory Manager:

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 this configuration file or if you created the configuration file in the default location, you do not need to set this environment variable.

Default: `/etc/Printsrv/aopmsg.conf`

AOPVALIDATEDB

If this environment variable is 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: The environment variable is not set.

Example: `AOPVALIDATEDB=YES`

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. This locale overrides the locale in the `LC_TIME` variable.

Rule: In the z/OS HTTP environment variables file (`httpd.envvars`), do not specify a value that is associated with a code page other than IBM-1047.

Examples:

`aopstart EXEC: LC_ALL=Fr_FR.IBM-297`

`httpd.envvars` file: `LC_ALL=Fr_FR.IBM-1047`

Default: `C` (also called POSIX)

LC_CTYPE

The locale that determines the EBCDIC code page used to validate Infoprint Server attribute values.

Rules: In the z/OS HTTP Server environment variables file (`httpd.envvars`):

- Do not specify the `LC_CTYPE` environment variable.
- Specify the `AOPCENTRAL_CODEPAGE` environment variable with the same EBCDIC code page as specified in the `LC_CTYPE`

environment variable. For example:
AOPCENTRAL_CODEPAGE=Fr_FR.IBM-297

Example: LC_CTYPE=Fr_FR.IBM-297

Default: C (also called POSIX). The default code page is IBM-1047.

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)

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 is set in the **aopstart** EXEC. You must set this environment variable in the **/etc/profile** file.

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 are displayed.

Default: None.

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 file name.

Default: /usr/lpp/Printsrv/%L/%N:/usr/lpp/Printsrv/En_US/%N:/usr/lib/nls/msg/%L/%N

This default is set in the **aopstart** EXEC. You must specify this environment variable in the **/etc/profile** file.

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 is set in the **aopstart** EXEC. You must specify this environment variable in the **/etc/profile** file.

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.

For information about the allowed values, see *z/OS UNIX System Services Command Reference*.

Default: The z/OS HTTP Server default is EST5EDT (Eastern U.S. time zone). The default for the Printer Inventory Manager is GMT0.

Related information:

- For information about the optional tracing variables, see *z/OS Infoprint Server Messages and Diagnosis*.
- For information about how to edit the **aopstart** EXEC and AOPSTART JCL procedure, see “Customizing startup and shutdown commands and procedures” on page 56.
- For more information about the LANG, LC_ALL, LC_CTYPE, LC_TIME, and TZ environment variables, see *z/OS UNIX System Services Command Reference*.

Setting up security for the Printer Inventory and operator commands

This section describes how to set up security for the Printer Inventory, common message log, and operator commands. You can specify which users can read and update the Printer Inventory, read and write 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. For information about RACF, see *z/OS Security Server RACF Security Administrator’s Guide*.

Table 9 summarizes the RACF access that is required to do different functions. For additional security requirements if you use Infoprint Central or the MVS system logger, see:

- “Setting up security for Infoprint Central” on page 341
- “Setting up security for log streams” on page 91

Table 9. RACF access for Infoprint Server

Function	AOP.ADMINISTRATOR profile (PRINTSRV class)	AOP.CONFIGURATION profile (PRINTSRV class)	Member of AOPADMIN group	Member of AOOPER group
Use ISPF panels to view and update the Printer Inventory except for the system configuration definition (see Note)	<ul style="list-style-type: none"> • View: READ access • View and update: UPDATE access 	Not required	Not required	Not required
Use ISPF panels to view and update the system configuration definition (see Note)	<ul style="list-style-type: none"> • View: READ access • View and update: UPDATE access only if AOP.CONFIGURATION is not defined 	<ul style="list-style-type: none"> • View: READ access • View and update: UPDATE access 	Not required	Not required

Table 9. RACF access for Infoprint Server (continued)

Function	AOP.ADMINISTRATOR profile (PRINTSRV class)	AOP.CONFIGURATION profile (PRINTSRV class)	Member of AOPADMIN group	Member of AOOPER group
Use the PIDU program to view and update the Printer Inventory except for the system configuration definition	<ul style="list-style-type: none"> View: READ access View and update: UPDATE access 	Not required	Required	Not required
Use the PIDU program to view and update the system configuration definition	<ul style="list-style-type: none"> View: READ access View and update: UPDATE access only if AOP.CONFIGURATION is not defined 	<ul style="list-style-type: none"> View: READ access View and update: UPDATE access 	Required	Not required
Use Infoprint Central to view PSF printers, printer definitions, and IP PrintWay job selection rules	READ access	Not required	Not required	Not required
Use the aoplogu command to view messages in the common message log	Not required	Not required	Required	Not required
Start and stop Infoprint Server daemons	Not required	Not required	Not required	Required
Initialize the search database for Infoprint Central	Not required	Not required	Required	Not required
View transform messages in the transform's stderr file	Not required	Not required	Required	Not required
View trace information that is written to the default trace directory	Not required	Not required	Required	Not required
Note: To use the ISPF panels, users must have READ access to the Infoprint Server ISPF data sets.				

AOP.ADMINISTRATOR profile

The AOP.ADMINISTRATOR resource profile in the RACF PRINTSRV class is a required profile. It restricts access to objects in the Printer Inventory:

- READ or UPDATE access lets users view the Printer Inventory using ISPF panels, Infoprint Central, or the Printer Inventory Definition Utility (PIDU). Users do not need any access to the profile to simply to list names of printer definitions with the **lpstat** command or with Infoprint Port Monitor.
- UPDATE access lets users update all objects except for the system configuration definition in the Printer Inventory using ISPF panels or the PIDU program. If the AOP.CONFIGURATION resource profile is not defined in the PRINTSRV class, UPDATE access to the AOP.ADMINISTRATOR profile lets users update the system configuration definition.

Tips:

- If the AOP.ADMINISTRATOR profile does not exist in the PRINTSRV class, Infoprint Server checks the AOPADMIN profile in the FACILILTY class. READ or higher access to the AOPADMIN profile is equivalent to UPDATE access to the AOP.ADMINISTRATOR profile.
- RACF profiles with the prefix AOP in the PRINTSRV class are reserved by IBM. Do not create profiles for other purposes that start with AOP in the PRINTSRV class.

AOP.CONFIGURATION profile

The AOP.CONFIGURATION resource profile in the RACF PRINTSRV class is an optional profile. It lets you restrict access to the system configuration definition separately from other definition in the Printer Inventory:

- READ or UPDATE access to either the AOP.CONFIGURATION profile or the AOP.ADMINISTRATOR profile lets users view the system configuration definition.
- UPDATE access to the AOP.CONFIGURATION profile lets users update the system configuration definition.

If the AOP.CONFIGURATION profile does not exist, UPDATE access to the AOP.ADMINISTRATOR profile lets users update the system configuration definition as well as update other definitions in the Printer Inventory.

Tips:

- Consider defining the AOP.CONFIGURATION profile before you enable the dynamic configuration function because when dynamic configuration is enabled, users who have UPDATE access to the AOP.ADMINISTRATOR profile can update the system configuration definition.
- If the AOP.ADMINISTRATOR and AOP.CONFIGURATION profiles both do not exist in the PRINTSRV class, Infoprint Server checks the AOPADMIN profile in the FACILILTY class. READ or higher access to the AOPADMIN profile is equivalent to UPDATE access to the AOP.ADMINISTRATOR and AOP.CONFIGURATION profiles.
- RACF profiles with the prefix AOP in the PRINTSRV class are reserved by IBM. Do not create profiles for other purposes that start with AOP in the PRINTSRV class.

AOPADMIN and AOOPER groups

You must define these groups to RACF to restrict 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.

If you do not want to distinguish between administrators and operators, you need to define only one group.

You can choose any names for these groups.

AOPCONF group

You can optionally define the AOPCONF group to RACF for Infoprint Server configuration managers who can update the system configuration definition in the

Printer Inventory. Consider defining the AOPCONF group if you define the AOP.CONFIGURATION profile. If you define the AOPCONF group, you can give the group UPDATE access to the AOP.CONFIGURATION profile.

If you do not want to distinguish between Infoprint Server administrators (members of the AOPADMIN group) and Infoprint Server configuration managers, you do not need to define this group. Instead, you can give the AOPADMIN group or individual users UPDATE access to the AOP.CONFIGURATION profile if you define it.

You can choose any name for this group.

Steps for setting up security for the Printer Inventory, common message log, and operator commands

These steps show the RACF and z/OS UNIX commands to use to set up security for the Printer Inventory, common message log, and Infoprint Server operator commands.

Tip: The sample CLIST in SYS1.SAMPLIB(AOPRACF) contains the RACF commands for most of the steps that are described in this section. You must modify AOPRACF before you run it.

1. Define a group to RACF for Infoprint Server administrators.

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).

Example: This ADDGROUP command defines group AOPADMIN. For *group-identifier*, specify an integer that is different from other GIDs in your installation:

```
ADDGROUP (AOPADMIN) OMVS(GID(group-identifier))
```

If you use the MVS system logger for the common message log, Infoprint Server administrators must also have READ access to the profile that protects the log stream. For information, see “Setting up security for log streams” on page 91.

2. Define a group to RACF for Infoprint Server operators.

Infoprint Server operators can start and stop Infoprint Server daemons using the **aopstart** and **aopstop** commands or the AOPSTART and AOPSTOP procedures. The suggested group name for Infoprint Server operators is AOPOPER. However, you can use any name. The group profile must have an OMVS segment and a group identifier (GID).

Example: This ADDGROUP command creates group AOPOPER. For *group-identifier*, specify an integer that is different from other GIDs in your installation:

```
ADDGROUP (AOPOPER) OMVS(GID(group-identifier))
```

If your installation’s security policy does not require a distinction between Infoprint Server administrators and operators, you can skip this step and give the group for Infoprint Server administrators (AOPADMIN) access to the AOP.ADMINISTRATOR profile in a later step.

3. (Optional) Define a group to RACF for Infoprint Server configuration managers.

Infoprint Server configuration managers can view and update Infoprint Server configuration attributes in the system configuration definition if the dynamic configuration function is enabled. The suggested group name for Infoprint Server configuration managers is AOPCONF. However, you can use any name. The group profile does not need an OMVS segment and a group identifier (GID).

Example: This ADDGROUP command creates group AOPCONF:

```
ADDGROUP (AOPCONF)
```

Consider defining the AOPCONF group if you enable dynamic configuration and you define the AOP.CONFIGURATION resource profile to restrict access to the system configuration definition.

4. Activate the PRINTSRV class. In addition, to improve performance, copy profiles in the PRINTSRV class into virtual storage. If you copy profiles into virtual storage, you must use the SETROPTS command to refresh the PRINTSRV class after you define new profiles or permit new users to profiles in the class.

Example: This SETROPTS command activates the PRINTSRV class and copies profiles into virtual storage.

```
SETROPTS CLASSACT(PRINTSRV) RACLIST(PRINTSRV)
```

5. Define a resource profile to RACF named AOP.ADMINISTRATOR in the PRINTSRV class.

Example: This RDEFINE command defines resource profile AOP.ADMINISTRATOR with universal READ access:

```
RDEFINE PRINTSRV (AOP.ADMINISTRATOR) UACC(READ)  
SETROPTS RACLIST(PRINTSRV) REFRESH
```

Guideline: If you want RACF to notify the security administrator (by a message) when a user tries to read or update the Printer Inventory and is not authorized, specify the NOTIFY parameter on the RDEFINE command. Otherwise, Infoprint Server suppresses RACF messages when it checks the AOP.ADMINISTRATOR profile.

6. (Optional) Define a resource profile to RACF named AOP.CONFIGURATION in the PRINTSRV class.

If you want to restrict who can update the system configuration definition separately from other definitions in the Printer Inventory, define the AOP.CONFIGURATION profile. If you do not define the AOP.CONFIGURATION profile, users who have UPDATE access to the AOP.ADMINISTRATOR profile can update the system configuration definition as well as other objects in the Printer Inventory.

Users who have READ access to either the AOP.CONFIGURATION profile or the AOP.ADMINISTRATOR profile can view the system configuration definition.

Example: This RDEFINE command defines resource profile AOP.CONFIGURATION with universal access of NONE:

```
RDEFINE PRINTSRV (AOP.CONFIGURATION) UACC(NONE)  
SETROPTS RACLIST(PRINTSRV) REFRESH
```

-
7. Give non-administrators READ access to the AOP.ADMINISTRATOR profile. If you defined resource profile AOP.ADMINISTRATOR with universal access of NONE, give users who need to view the Printer Inventory using either Infoprint Server ISPF panels or Infoprint Central, READ access to the AOP.ADMINISTRATOR profile.

Example: This PERMIT command lets a user or group read the Printer Inventory:

```
PERMIT AOP.ADMINISTRATOR CLASS(PRINTSRV) ACCESS(READ) ID(userid or groupid)
SETROPTS RACLIST(PRINTSRV) REFRESH
```

Note: READ access to the AOP.ADMINISTRATOR profile also lets users use the ISPF panels to view the system configuration definition.

8. Give the Infoprint Server administrator group (AOPADMIN) UPDATE access to the AOP.ADMINISTRATOR resource profile.

You can permit group AOPADMIN to the AOP.ADMINISTRATOR resource profile or you can permit each user separately. Users must have UPDATE (or higher) access to the AOP.ADMINISTRATOR resource profile to view and update the Printer Inventory.

Example: This PERMIT command permits the AOPADMIN group to the AOP.ADMINISTRATOR profile in the PRINTSRV class:

```
PERMIT AOP.ADMINISTRATOR CLASS(PRINTSRV) ACCESS(UPDATE) ID(AOPADMIN)
SETROPTS RACLIST(PRINTSRV) REFRESH
```

9. (Optional) If you defined the AOP.CONFIGURATION resource profile in step 6, give Infoprint Server configuration managers UPDATE access to the profile.

You can permit group AOPCONF to the AOP.CONFIGURATION resource profile or you can permit each user separately.

Example: This PERMIT command permits the AOPCONF group to the AOP.CONFIGURATION profile in the PRINTSRV class:

```
PERMIT AOP.CONFIGURATION CLASS(PRINTSRV) ACCESS(UPDATE) ID(AOPCONF)
SETROPTS RACLIST(PRINTSRV) REFRESH
```

10. (Optional) If you defined resource profiles AOP.CONFIGURATION and AOP.ADMINISTRATOR with universal access of NONE, give users who need to view the system configuration definition, but not update it, READ access to either the AOP.CONFIGURATION profile or the AOP.ADMINISTRATOR profile.

Example: This PERMIT command lets a user view the system configuration definition:

```
PERMIT AOP.CONFIGURATION CLASS(PRINTSRV) ACCESS(READ) ID(userid)
SETROPTS RACLIST(PRINTSRV) REFRESH
```

11. Define Infoprint Server administrators, operators, and other users whom you want to read the Printer Inventory to RACF as z/OS UNIX users.

The steps shown here illustrate one way to define users to RACF. For complete information about how to define z/OS UNIX users to RACF, see *z/OS UNIX System Services Planning*.

- a. For each user, establish an OMVS segment and set the user's default group to a group that has an OMVS segment with a group identifier (GID). You can specify any home directory or use the default home directory. You can use either the ADDUSER command to define a new user ID or the ALTUSER command to modify an existing user ID.

Example: This ALTUSER command modifies an existing user ID. For *userid*, specify an existing user ID. For *user-identifier*, specify an integer that is different from other UIDs in your installation. For *group*, specify AOPOPER, AOPADMIN, or any another group that has an OMVS GID.

```
ALTUSER userid OMVS(UID(user-identifier) PROGRAM('/bin/sh'))  
      DFLTGRP(group)
```

Guidelines:

- 1) The user's default group must have a GID defined so that the user can use z/OS UNIX.
- 2) For instructions about how to define a user ID for use with the AOPSTART and AOPSTOP procedures, see “Creating a user ID to assign to the AOPSTART and AOPSTOP procedures” on page 60.

12. Connect Infoprint Server administrators, operators, and configuration managers to the appropriate group.

If you use the AOPSTART and AOPSTOP JCL procedures to start and stop Infoprint Server, connect the user ID assigned to these procedures to the AOPOPER group.

Example: These CONNECT commands connect a user to the AOPADMIN group for administrators, connect a user to the AOPOPER group for operators, and connect a user to the AOPCONF group for configuration managers. Specify an existing user ID for *userid*:

```
CONNECT (userid) GROUP(AOPADMIN)
```

```
CONNECT (userid) GROUP(AOPOPER)
```

```
CONNECT (userid) GROUP(AOPCONF)
```

Tip: If you did not define a separate group for Infoprint Server operators in step 2, connect the operators to the group for Infoprint Server administrators (AOPADMIN).

13. Give the AOPOPER group (or the user ID that is associated with the AOPSTART procedure) RACF authorization to these resources and profiles:

- If you use IP PrintWay extended mode, UPDATE access to profiles in the JESSPOOL class. See “Authorizing IP PrintWay to select data sets from the JES spool (extended mode)” on page 214.
- If you use IP PrintWay extended mode, READ access to the BPX.SMF profile in the FACILITY class. See “Authorizing IP PrintWay to write SMF type 6 records (extended mode)” on page 215.
- If you use Infoprint Central, access to profiles in the OPERCMDS and JESSPOOL classes. For information about the type of access required, see “Setting up security for Infoprint Central” on page 341.
- If you use IBM transforms from AFP to PCL, PDF, or PostScript, READ access to transform resources, such as AFP resource libraries, that reside on the z/OS system. For more information, see the documentation for the transform.

- If you use the MVS system logger for the common message log, UPDATE access to the profile that protects the log stream. For information, see “Setting up security for log streams” on page 91.

Tip: If you always use the AOPSTART and AOPSTART JCL procedures (and not the **aopstart** and **aopstop** commands) to start and stop Infoprint Server, instead of giving RACF access to the AOPOPER group, you can give RACF access to the user ID that is associated with the AOPSTART and AOPSTOP procedures.

-
14. Give universal RACF access to the Infoprint Server ISPF data sets. This lets all users view Infoprint Server ISPF panels.

Example: The RACF ADDGROUP command creates a group named AOP to own the data set resource profile. The RACF ADDSD command creates a generic data set resource profile that gives universal READ access to data sets that start with AOP.:

```
ADDGROUP (AOP) SUPGROUP(SYS1) OWNER(SYS1)
ADDSD 'AOP.*' GENERIC OWNER(AOP) UACC(READ)
```

Setting permissions for directories and executable files -- aopsetup

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 these directories and files:

- **/var/Printsrv** directory:

This directory contains the Printer Inventory files and other Infoprint Server files. 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 (suggested name is AOPADMIN) and users with an 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 by members of the RACF group for Infoprint Server operators (AOPOPER).
- Administrative commands, such as **pidu**, are executable by members of the RACF group for Infoprint Server administrators (AOPADMIN).
- The Printer Inventory Manager daemon, **aopd**, has the set-group-ID bit on.

Tip: **aopsetup** does not set permissions for all files in this directory. Therefore, you cannot copy this directory to another system and use **aopsetup** to restore all the original permissions.

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

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


operator-group

The name of the RACF group you created for Infoprint Server operators. The suggested RACF group name is AOOPER. However, you can use a different name for this group (see “AOPADMIN and AOOPER groups” on page 49). This argument is required.

administrator-group

The name of the RACF group you created for Infoprint Server administrators. The suggested RACF group name is AOPADMIN. However, you can use a different name for this group (see “AOPADMIN and AOOPER groups” on page 49). This argument is required.

Steps for running aopsetup

Before you begin:

- Define the RACF groups for Infoprint Server operators and administrators as described in “Setting up security for the Printer Inventory and operator commands” on page 47.
- Make sure that the file systems that contain the `/usr/lpp/Printsrv/bin` and `/var` or `/var/Printsrv` directories are mounted read/write.
- Make sure that the `/usr/lpp/Printsrv/bin` directory, which contains `aopsetup`, is in your PATH environment variable.
- If you do not want to use the default Infoprint Server base directory, `/var/Printsrv`, specify the `base-directory` attribute in the Infoprint Server configuration file, `aopd.conf`.
- You must have an effective UID of 0.

Steps for running aopsetup:

1. Switch to an effective UID of 0.

Example:

```
su
```

Tip: To use the z/OS UNIX `su` command, you must be permitted to the BPX.SUPERUSER profile in the FACILITY class in RACF.

2. Run `aopsetup`.

You can run `aopsetup` from a rlogin shell or from an OMVS session.

Tip: You should run `aopsetup` before you start Infoprint Server daemons. However, if you attempted to start Infoprint Server before running `aopsetup`, stop all Infoprint Server daemons before doing this step.

Examples:

- If you defined group AOOPER for operators and group AOPADMIN for administrators, enter:

```
/usr/lpp/Printsrv/bin/aopsetup AOOPER AOPADMIN
```
- If you defined group AOPADMIN for both Infoprint Server operators and administrators, enter:

```
/usr/lpp/Printsrv/bin/aopsetup AOPADMIN AOPADMIN
```
- If you want to run `aopsetup` while the `/usr/lpp/Printsrv` directory is mounted at a `/service` mount point, specify the service directory in the `INSTALL_DIR` environment variable. For example, enter:

```
INSTALL_DIR=/service/usr/lpp/Printsrv aopsetup AOOPER AOPADMIN
```

3. If you switched to an effective UID of 0 in step 1, switch back to your own UID:

```
exit
```

Results: If you specified the AOPOPER and AOPADMIN groups as arguments to **aopsetup**, output from the z/OS UNIX **ls** command should look similar to the following sample output. Also, the owner ID of the **/var/Printsrv** directory and the executable files, shown as ROOT in the sample output, should have a UID of 0.

Tip: To display the UID of the owner ID, you can use the **-n** option on the **ls** command.

```
• ls -d -E /var/Printsrv

drwxrwx--x      11 ROOT  AOPADMIN ... .. /var/Printsrv
• ls -E /usr/lpp/Printsrv/bin

...
-rwx--S---  a---  2 ROOT  AOPADMIN ... .. aopd
-rwxr-x---  --s-  2 ROOT  AOPADMIN ... .. aoplogu
-rwsr-xr--  --s-  2 ROOT  AOPOPER  ... .. aopstart
-rwsr-x---  --s-  2 ROOT  AOPOPER  ... .. aopstop
-rwxr-x---  a-s-  2 ROOT  AOPADMIN ... .. hinvu
-rwxr-x---  --s-  2 ROOT  AOPADMIN ... .. pidu
-rwxr-x---  a-s-  2 ROOT  AOPADMIN ... .. sdbu
```

Customizing 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.

The user ID assigned to the AOPSTART procedure or the user ID that runs the **aopstart** command must be connected to the AOPOPER group and have authorization to these RACF profiles:

- If you use the AFP to PCL, AFP to PostScript, or AFP to PDF transform, this user ID must have READ access to all system and user AFP resource libraries that the transform uses. See the documentation for the transform.
- If you use Infoprint Central, this user ID must have access to profiles in the OPERCMDS and JESSPOOL classes. See “Setting up security for Infoprint Central” on page 341.
- If you use IP PrintWay extended mode, this user ID must have UPDATE access to profiles in the JESSPOOL class and READ access to the BPX.SMF profile. See “Authorizing IP PrintWay to select data sets from the JES spool (extended mode)” on page 214 and “Authorizing IP PrintWay to write SMF type 6 records (extended mode)” on page 215.

Guideline: IBM strongly suggests that you use the AOPSTART procedure to run the **aopstart** command instead of running the **aopstart** command directly from the z/OS UNIX command line for these reasons:

- With the AOPSTART procedure, you can associate one user ID with the AOPSTART JCL procedure and that user can be the sole member of the AOPOPER group. You can give that user access to the required RACF profiles.
- If you run the **aopstart** command directly from the z/OS UNIX command line, Infoprint Server can inherit undesirable limits, such as the maximum CPU time limit, from the user ID that runs the command. For information about system limits, see *z/OS UNIX System Services Planning*.

Tip: The AOPSTART and AOPSTOP procedures, as well as the **aopstart** and **aopstop** commands, can end with abends EC6 (reason code FFFF) and 33E. These abends do not indicate an error, and they are typically suppressed in the IEASLP00 member of SYS1.PARMLIB. This statement in the IEASLP00 member suppresses EC6 abends:

```
SLIP SET,C=EC6,RE=0000FFXX,ID=XEC6,A=NODUMP,END
```

This table summarizes the customization tasks related to setting up your startup and shutdown procedures. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies to your installation.

Task	Condition	See page
Editing the aopstart EXEC	Optional: To specify environment variables	57
Creating a user ID to assign to the AOPSTART and AOPSTOP procedures	Optional: To use the AOPSTART and AOPSTOP JCL procedures	60
Defining the AOPSTART and AOPSTOP procedures to RACF	Optional: To use the AOPSTART and AOPSTOP JCL procedures	61
Editing the AOPSTART startup procedure	Optional: To use the AOPSTART JCL procedure and change default settings	61
Editing the AOPSTOP shutdown procedure	Optional: To use the AOPSTOP JCL procedure and change default settings	63
Starting Infoprint Server daemons automatically	Optional: To start Infoprint Server daemons automatically	65

Editing the aopstart EXEC

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**).

The **aopstart** EXEC provided with Infoprint Server is suitable for most installations. However, you must edit the **aopstart** EXEC if you need to specify certain environment variables. For security reasons, Infoprint Server uses some environment variables only if they are set in the **aopstart** EXEC. Infoprint Server does not use the same variables set in the **/etc/profile** file. See Appendix C, “Environment variables,” on page 409 for a summary of the environment variables that you must set in the **aopstart** EXEC.

In the **aopstart** EXEC, you can:

- Set a new environment variable using this format:

```
n=n+1;env.n='environment_variable=value'
```

Specify all environment variables before this statement in the EXEC:

```
env.0=n
```

Example:

```
n=n+1;env.n='AOPTRACEDIR=/mydirectory/trace'
env.0=n
```

- Change the value of any environment variable that is set in the EXEC.
- Change the name of the directory where Infoprint Server files are installed by editing this statement:

```
install_path='/usr/lpp/Printsrv'
```

Rule: The owner of the **aopstart** file must be UID 0. Also, only users with a UID of 0 should be permitted to write the file.

Steps for copying and editing the aopstart EXEC

Before you begin: You must have an effective UID of 0.

These steps show the z/OS UNIX commands to use to copy the **aopstart** EXEC from the **usr/lpp/Printsrv/bin** directory to a new directory and edit it. If you have already copied the **aopstart** EXEC to a new directory, follow the steps in “Steps for editing the aopstart EXEC” on page 59.

1. Switch to an effective UID of 0:

```
su
```

Tip: To use the z/OS UNIX **su** command, you must be permitted to the BPX.SUPERUSER profile in the FACILITY class in RACF.

2. Copy the default **aopstart** EXEC to another directory. You must copy the file because you should not modify the directory that contains Infoprint Server executable files. This example copies the **aopstart** EXEC to **/usr/local/bin**:

```
cp /usr/lpp/Printsrv/bin/aopstart /usr/local/bin
```

3. Change to the directory where **aopstart** exists:

```
cd /usr/local/bin
```

4. Change the owner of the **aopstart** file to UID 0:

```
chown 0 aopstart
```

5. Change the group owner of the **aopstart** file to AOPOPER. AOPOPER is the suggested RACF group name for Infoprint Server operators. However, you can use a different name for this group (see “AOPADMIN and AOPOPER groups” on page 49).

```
chgrp AOPOPER aopstart
```

6. Edit the new **aopstart** EXEC using your preferred editor and save your changes. For example:

```
oedit aopstart
```

Edit this file using the IBM-1047 code page. For additional instructions, see the comments in the **aopstart** EXEC.

Tip: You can select the ISPF “z/OS UNIX Shell” option (ISPF 13.11) to edit z/OS UNIX files.

7. Change the permissions for the file. Make the file writeable by only UID 0, readable by everyone, and executable only by UID 0 and members of the AOPOPER group. Also, set the **set-uid-flag** on. For example:

```
chmod 4754 aopstart
```

Tip: Do this step after you edit the file because the **set-uid-flag** is turned off when you edit a file.

8. Place the directory that contains the new **aopstart** EXEC in the PATH environment variable of each operator's **/u/userid/.profile** file *before* the directory that contains Infoprint Server files (default directory **/usr/lpp/Printsrv/bin**). For example, you can add this line to the end of the operators' profile files:

```
export PATH=/usr/local/bin:$PATH
```

9. If you switched to an effective UID of 0 in step 1, switch back to your own UID:

```
exit
```

10. Specify the full path name of the **aopstart** EXEC in the **/etc/rc** shell script. For details, see “Starting Infoprint Server daemons automatically” on page 65.
-

11. Specify the full path name of the **aopstart** EXEC on the PARM parameter in the AOPSTART procedure. For details, see “Editing the AOPSTART startup procedure” on page 61.
-

Steps for editing the aopstart EXEC

These steps show the z/OS UNIX commands to use to edit the **aopstart** EXEC. If the **aopstart** EXEC is in the **/usr/lpp/Printsrv/bin** directory, see “Steps for copying and editing the aopstart EXEC” on page 58.

Before you begin: You must have an effective UID of 0.

1. Switch to an effective UID of 0:

```
su
```

To use the z/OS UNIX **su** command, you must be permitted to the BPX.SUPERUSER profile in the FACILITY class in RACF.

2. Change to the directory where **aopstart** exists. For example:

```
cd /usr/local/bin
```

3. Edit the **aopstart** file using your preferred editor and save your changes. For example:

```
oedit aopstart
```

See comments in the EXEC. Edit this file using the IBM-1047 code page. For additional instructions, see the comments in the **aopstart** EXEC.

Tip: You can select the ISPF “z/OS UNIX Shell” option (ISPF 13.11) to edit z/OS UNIX files.

4. Set the **set-uid-flag** on. For example:

```
chmod 4754 aopstart
```

Tip: Do this step after you edit the file because the **set-uid-flag** is turned off when you edit a file.

5. If you switched to an effective UID of 0 in step 1, switch back to your own UID:

```
exit
```

6. If Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server for your changes to take effect. For example, enter these MVS commands to run the AOPSTOP and AOPSTART JCL procedures to stop and restart all Infoprint Server daemons:

```
START AOPSTOP  
START AOPSTART
```

Creating a user ID to assign to the AOPSTART and AOPSTOP procedures

Do these steps to create a user ID to assign to the AOPSTART and AOPSTOP procedures. This user ID must be defined to RACF as a z/OS UNIX user and be a member of the AOPOPER group.

Tip: The sample CLIST in SYS1.SAMPLIB(AOPRACF) contains the RACF commands that are described in this section. You must modify AOPRACF before you run it.

1. Define group AOPOPER to RACF as described in “Setting up security for the Printer Inventory and operator commands” on page 47. You can use another name for the AOPOPER group.
-

2. Create a z/OS user ID. The user ID must have an OMVS segment, and the default group must have an OMVS segment and a GID.

You can select any name for the user ID, or you can alter an existing user ID to be a z/OS UNIX ID. You can specify any home directory including the default directory.

Example: This RACF command defines user AOPSTC with default group AOPOPER and home directory **/tmp**.

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) PROGRAM('/bin/sh')) HOME(/tmp)
DFLTGRP(AOPOPER) NOPASSWORD
```

3. If the user's default group is not AOPOPER, connect the user ID to group AOPOPER.

Example: This CONNECT command connects user ID AOPSTC to group AOPOPER:

```
CONNECT (AOPSTC) GROUP(AOPOPER)
```

Note: If the user ID assigned to the AOPSTART and AOPSTOP procedures to RACF as a z/OS UNIX user is not a z/OS UNIX user, the Infoprint Server daemons do not start or stop, and you might not receive any error messages.

Defining the AOPSTART and AOPSTOP procedures to RACF

If you use the AOPSTART and AOPSTOP procedures, you must define profiles for them in the RACF STARTED class.

Example: These commands assume that the user ID you defined for use with these procedures is AOPSTC and that AOPSTC is connected to group AOPOPER.

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

Tips:

- The sample CLIST in SYS1.SAMPLIB(AOPRACF) contains the RACF commands described in this section. You must modify AOPRACF before you run it.
- If you want to use the started procedure table (ICHRIN03) instead, see *z/OS Security Server RACF Security Administrator's Guide* for more information.

Editing the AOPSTART startup procedure

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

AOPSTART is shipped in SYS1.IBM.PROCLIB. However, during installation AOPSTART might have been copied to another data set in the STC PROCLIB concatenation.

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

- The region size of 512 megabytes is not suitable. These minimum requirements apply:
 - If you start the IPP Server, specify at least 200 megabytes.
 - If you use transforms to convert data streams from one format to another, specify at least 256 megabytes.
- The **aopstart** command is not in the default directory **/usr/lpp/Printsrv/bin**. You specify the directory in the PARM parameter on the EXEC statement.
- You want to specify any of these environment variables:

- AOPVALIDATEDB
- LANG
- LC_ALL, LC_CTYPE, LC_TIME
- TZ
- For tracing: AOPTRACEON, AOPTRACEON_DATA, AOPTRACEON_FILTER, AOPTRACEON_MSGLOG, AOPTRACEBYTES, AOPTRACEMAXFILES
- AOPDUMPON

For information about these environment variables, see Appendix C, “Environment variables,” on page 409.

Tip: You can choose to specify these environment variables in the **aopstart** EXEC. If you specify them in the **aopstart** EXEC, do not specify them in the AOPSTART procedure.

Rules:

- For security purposes, you can specify only a limited set of environment variables in the AOPSTART procedure. You must specify all other environment variables in the **aopstart** EXEC, which only authorized users can edit. See “Editing the aopstart EXEC” on page 57.
- You can specify in-stream environment variables in the JCL, or you can specify the environment variables in a z/OS UNIX file or MVS data set that the STDENV data definition (DD) points to.
- If you specify the environment variables in an MVS data set, it must have these DCB attributes: RECFM=VB and LRECL=255.
- Specify each environment variable in the format *variable=value*. Specify one environment variable per line or record, starting in column 1. For example:
AOPTRACEON=1

Rules:

- If you specify a z/OS UNIX file or an MVS data set in a DD statement in the AOPSTART procedure, the user ID associated with the AOPSTART procedure must be authorized to:
 - Read the data set or file specified in the STDENV data definition (DD)
 - Read and write to the data set or file specified in the STDOUT and STDERR DD statements (not required if these DD statements point to SYSOUT)

To authorize the user ID:

- For an MVS data set, give the AOPOPER group RACF authorization to read or write the data set.
- For a z/OS UNIX file, make the file owned by group AOPOPER and set the UNIX permission bits to let the group read, or read and write, to the file.
- If you are using JES2 dynamic PROCLIBs, refresh the PROCLIBs before restarting Infoprint Server.

Example: To refresh PROCLIBs, enter this JES2 command:

```
$T PROCLIB(*)
```

- If Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server for your changes to take effect. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Example: To stop and restart all Infoprint Server daemons, enter these MVS commands:

```
START AOPSTOP
START AOPSTART
```

Figure 10 shows the AOPSTART procedure. If you need more information about the AOPBATCH program, see Appendix E, “AOPBATCH program,” on page 455.

```

/*-----
/*
/* 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 as required for the installation
/* directory.
/*-----
//AOPSTART EXEC PGM=AOPBATCH,PARM='/usr/lpp/Printsrv/bin/aopstart',
//           REGION=512M,
//           TIME=NOLIMIT
//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:
/*
/* AOPTRACEON=1
/*-----

```

Figure 10. AOPSTART startup procedure — SYS1.IBM.PROCLIB(AOPSTART)

Editing the AOPSTOP shutdown procedure

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

AOPSTOP is shipped in SYS1.IBM.PROCLIB. However, during installation AOPSTOP might have been copied to another data set in the STC PROCLIB concatenation.

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

- If 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. You can specify in-stream environment variables in the JCL, or you can specify environment variables in a z/OS UNIX file or MVS data set that the STDENV data definition (DD) points to. The AOPSTOP procedure does not use environment variables that are set elsewhere, for example in the **/etc/profile** file.

When the operator starts 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 Infoprint Server Transform Manager and the LPD daemons immediately, enter this MVS START command:

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

For the **aopstop** command options and more examples of how to specify options on the AOPSTOP procedure, see *z/OS Infoprint Server Operation and Administration*.

If you specify an MVS data set or z/OS UNIX file in a DD statement in the AOPSTOP procedure, the user ID associated with the AOPSTOP procedure must be authorized to:

- Read the data set or file specified in the STDENV data definition (DD)
- Read and write to the data set or file specified in the STDOUT and STDERR DD statements (not required if these DD statements point to SYSOUT)

To authorize the user ID:

- For an MVS data set, give the AOPOPER group RACF authorization to read or write the data set.
- For a z/OS UNIX file, make the file owned by group AOPOPER and set the UNIX permission bits to let the group read, or read and write, to the file.

Figure 11 on page 65 shows the AOPSTOP procedure. If you need more information about the AOPBATCH program, see Appendix E, "AOPBATCH program," on page 455.


```

/*-----
/*
/* AOPSTOP - This procedure stops the Infoprint Server daemons.
/*           The USERID assigned to this proc should be a member
/*           of the AOOPER 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 11. AOPSTOP shutdown procedure — SYS1.IBM.PROCLIB(AOPSTOP)

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 run 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 this command to `/etc/rc`:

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

Rules:

- Start the Printer Inventory Manager daemons *before* you start IP PrintWay basic mode FSAs or the NetSpool task.
- If PSF uses the Printer Inventory, start the Printer Inventory Manager daemons before you start any printers in the PSF FSS. For information about how to start PSF, see *PSF for z/OS: Customization*.
- Start TCP/IP, and make sure that TCP/IP has finished initialization, before you start these Infoprint Server daemons: LPD (`aoplpd`), the IPP Server daemon (`aopippd`), the IP PrintWay extended mode daemons (`aopoutd` and `aopwsmd`), and the SNMP subagent daemon (`aopsnmpd`).

If TCP/IP has not finished initialization, you might receive this error message:

```

BPXF024I (BPX0INIT aoplpd: socket(AF_INET,SOCK_STREAM,0)failed in
TCPSocket::TCPSocket (int&,int,int) at ./src/tcpsock.cpp 83:
EDC5112I Resource temporarily unavailable. errno=0x74b30296

```

Tips:

- If you use the AOPSTART procedure, you can specify the order in which TCP/IP and Infoprint Server start at system initialization in the COMMNDxx parmlib member. Put as many other start commands as possible in COMMNDxx between the TCP/IP startup procedure and the AOPSTART procedure to give TCP/IP as much time as possible to start before the AOPSTART procedure starts.
- If you add the **aopstart** command to **/etc/rc**, use the **setstack** command to make sure that TCP/IP has finished initialization. The **setstack** command is not formally supported by IBM. However, you can download **setstack** from the "Tools & Toys" link at website (<http://www.ibm.com/systems/z/os/zos/features/unix/>) to a directory identified in the **PATH** environment variable. For example, if the **aopstart** EXEC is in the default directory, you could add this line to **/etc/rc**:

```
setstack -w /usr/lpp/Printsrv/bin/aopstart &
```

Use the ampersand (&) on the **setstack** command so that **/etc/rc** can continue without blocking if **setstack** waits for TCP/IP to become available.

Enabling Infoprint Server ISPF panels

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

Infoprint Server ISPF panels are available in English and Japanese. The Japanese version of the panels has been translated to code page IBM-939.

Defining libraries in the TSO logon procedure

If you have followed the instructions in the ISPF Setup sections in the *z/OS Program Directory*, you can skip the instructions in this section. Otherwise, concatenate these libraries in the TSO logon procedure:

- Concatenate one of these panel libraries to DD statement ISPLLIB, depending on the language in which you want to view the panels:
English AOP.SAOPPENU
Japanese AOP.SAOPPJPN
- Concatenate one of these message libraries to DD statement ISPLMLIB, depending on the language in which you want to receive messages:
English AOP.SAOPMENU
Japanese AOP.SAOPMJPJN
- Concatenate library AOP.SAOPEXEC to DD statement SYSPROC or SYSEXEC.

Tip: If you did not add the Language Environment runtime libraries (*hlq.SCEERUN* and *hlq.CEERUN2*) and the C++ runtime library (*hlq.SCLBDLL*) to the system LNKLST, concatenate them to the ISPLLIB DD statement, or concatenate them to STEPLIB. You can use the TSOLIB function of TSO/E to do this.

Related information: For information about how to access the Language Environment runtime libraries and improve the performance of runtime libraries, see *z/OS UNIX System Services Planning*.

Displaying the Infoprint Server panel option

ISPF provides sample ISPF panels to enable panels for most z/OS elements. The system programmer panel, `ISR@390S`, includes an entry for Infoprint Server. If you are not using the default ISPF panels, customize ISPF to display the **Infoprint Server** option. For information about how to use these sample panels to customize ISPF, see *z/OS Program Directory*.

Modifying the AOPINIT EXEC

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

- Configuration file
The path name of the Infoprint Server configuration file. The default file name is `/etc/Printsrv/aopd.conf`.
- 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`.
- Language
The value of the `LANG` and `LC_ALL` environment variables. The default is `En_US` (equivalent to C).
- 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 these values:

English	<code>/usr/lpp/Printsrv/En_US/%N</code>
Japanese	<code>/usr/lpp/Printsrv/Ja_JP/%N</code>

The default is `/usr/lpp/Printsrv/%L/%N`.
- IP PrintWay basic mode message data set
The name of the data set where IP PrintWay basic mode writes messages. The default is `ANF.MESSAGE`.
- IP PrintWay basic mode ISPF trace data set
The name of the data set to contain the trace of IP PrintWay ISPF processing. The default is `userid.ANFISPF.TRACE`.
- Printer Inventory Manager ISPF trace file
The name of the directory to contain the trace of ISPF processing. The default directory name is `/var/Printsrv/trace`.
- Install directory
The name of the directory where you installed Infoprint Server. The default directory is `/usr/lpp/Printsrv`. If you installed Infoprint Server in a non-default directory, edit this line:

```
install_path='/usr/lpp/Printsrv'
```

To modify AOPINIT, copy AOPINIT to a local library and follow the instructions in the comments in AOPINIT to make your changes. Then add the local library that contains AOPINIT to the `SYSPROC` or `SYSEXEC` DD statement of each user's TSO logon procedure at the beginning of the search order.

Enabling dynamic configuration

The dynamic configuration function lets you change many Infoprint Server configuration attributes while Infoprint Server is running, without stopping and restarting Infoprint Server. These configuration attributes are called **dynamic attributes** and are stored in the system configuration definition in the Printer Inventory.

To use the dynamic configuration function, you must enable it. Some advantages of enabling dynamic configuration are:

- Authorized administrators can use the Infoprint Server ISPF panels or the Printer Inventory Definition Utility (PIDU) to view and change the dynamic attributes.
- If you change an attribute in the system configuration definition, with a few exceptions, you do not need to stop and restart Infoprint Server for the change to take effect.
- You can exploit new functions in Infoprint Server that require dynamic configuration. For example, you can exploit the MVS system logger function.

To change attributes in the system configuration definition, authorized administrators can use the Infoprint Server System Configuration ISPF panel or the Printer Inventory Definition Utility (PIDU).

The AOP.CONFIGURATION and AOP.ADMINISTRATOR resource profiles in the RACF PRINTSRV class let you authorize users to view and update the system configuration definition:

- READ access to the AOP.CONFIGURATION or AOP.ADMINISTRATOR resource profile lets users view the system configuration definition.
- UPDATE access to the AOP.CONFIGURATION profile lets users view and update the system configuration definition. If the AOP.CONFIGURATION profile does not exist, UPDATE access to the AOP.ADMINISTRATOR profile lets users view and update the system configuration definition.

If both the AOP.CONFIGURATION and AOP.ADMINISTRATOR profiles do not exist, READ access to the AOPADMIN profile in the RACF FACILITY class lets users view and update the system configuration definition.

To use PIDU, users must also be members of the RACF group for Infoprint Server administrators (default group name is AOPADMIN).

Limitation: The IP PrintWay attributes in the system configuration definition apply only to IP PrintWay extended mode. IP PrintWay basic mode does not use them. To configure IP PrintWay basic mode, you must specify the equivalent environment variables in the IP PrintWay basic mode started task.

Related information: For information about:

- Dynamic attributes, see “Dynamic attributes” on page 69
- How to set up security for the system configuration definition, see “Setting up security for the Printer Inventory and operator commands” on page 47.
- How to enable dynamic configuration, see “Steps for enabling dynamic configuration” on page 72
- How to view and edit the system configuration definition, see “Editing the system configuration definition for Printer Inventory Manager” on page 76.

Dynamic attributes

Dynamic attributes are configuration attributes that are stored in the system configuration definition in the Printer Inventory. When dynamic configuration is enabled, you can change the dynamic attributes while Infoprint Server is running. If you change an attribute, with a few exceptions, the new value takes effect when you save the change. A few attributes require that you stop and restart the daemon or daemons that use the attribute before the change can take effect.

Table 10 lists the dynamic attributes and the name of the field on the Infoprint Server System Configuration ISPF panel that you use to specify the attribute. The third column indicates whether you need to restart any Infoprint Server daemons before a change you make can take effect.

Table 10. Dynamic attributes and ISPF field names

Dynamic attribute	ISPF field	Restart these daemons
allow-all-characters-in-line-data	Allow all characters in line data	None
aophinvd-max-thread-tasks	Other	All daemons
aopipdd-max-thread-tasks	Other	IPP Server daemon (aopipddxp)
aoplpd-max-thread-tasks	Other	Infoprint Server LPD (aoplpd)
aopnetd-max-thread-tasks	Other	NetSpool daemon (aopnetd)
aopoutd-max-thread-tasks	Other	IP PrintWay extended mode daemons (aopoutd, aopwsmd)
aopsdbd-max-thread-tasks	Other	All daemons
aopssid-max-thread-tasks	Other	Infoprint Central daemon (aopssid)
aopsubd-max-thread-tasks	Other	Subsystem daemon (aopsubd)
aopwsmd-max-thread-tasks	Other	IP PrintWay extended mode daemons (aopoutd, aopwsmd)
applid	APPL ID	None
ascii-codepage	ASCII	None
blank-truncation-classes	Blank truncation classes	None
console-name	Extended MCS console name	Infoprint Central daemon (aopssid)
ebcdic-codepage	EBCDIC	None
hardcopy-messages	Send messages to hardcopy log	None
hardcopy-message-list	Message list for hardcopy log	None
ipp-port-number	IPP port number	None
ignore-dcf-routing-errors	Ignore DEST, CLASS, FORMS routing errors	None
job-prefix	Job ID prefix	None
lpd-port-number	LPD port number	None
log-retention	Log retention period (days)	None

Table 10. Dynamic attributes and ISPF field names (continued)

Dynamic attribute	ISPF field	Restart these daemons
logstream-name	Log stream name	None
mail-do-not-add-suffixes	Do not add suffixes	None
mail-preserve-suffixes	Preserve suffixes	None
mail-use-first-address	Send email to address of first document	None
mailer-path-name	Application path	None
mailer-options	Application options	None
max-historical-inventory-size	Maximum Historical Inventory size (MBs)	None
max-thread-tasks	Other	None
netspool-use-fixed-jobid	Other	NetSpool daemon (aopnetd)
netspool-use-unaltered-jobid	Other	NetSpool daemon (aopnetd)
select-work-from-hold-queue	Select work from hold queue	IP PrintWay extended mode daemons (aopoutd, aopwsmd)
smf-recording	Write SMF records	None
snmp-community	Community name	SNMP subagent daemon (aopsnmpd)
suppress-post-unix-filter-formatting	Suppress formatting after UNIX filter	None

Table 11 lists the attributes that you can specify in the system configuration definition, the equivalent attributes in the **aopd.conf** and **aopmsg.conf** files, and the equivalent environment variables in the **aopstart EXEC**. If dynamic configuration is enabled, Infoprint Server uses the dynamic attributes in the system configuration definition and ignores the equivalent attributes in the **aopd.conf** and **aopmsg.conf** files and the equivalent environment variables in the **aopstart EXEC**.

Table 11. Attributes in system configuration definition and the equivalent attributes in **aopd.conf** and **aopmsg.conf** and equivalent environment variables

Attribute in system configuration definition	Equivalent attribute in aopd.conf and aopmsg.conf files	Equivalent environment variable in aopstart EXEC
allow-all-characters-in-line-data	None	ALLOW_ALL_CHARACTERS_IN_LINE_DATA
aophinvd-max-thread-tasks	None	AOPHINVD_MAXTHREADTASKS
aopipppd-max-thread-tasks	None	AOPIPPD_MAXTHREADTASKS
aoplpd-max-thread-tasks	None	AOPLPD_MAXTHREADTASKS
aopnetd-max-thread-tasks	None	AOPNETD_MAXTHREADTASKS
aopoutd-max-thread-tasks	None	AOPOUTD_MAXTHREADTASKS
aopsdbd-max-thread-tasks	None	AOPSDBD_MAXTHREADTASKS
aopssid-max-thread-tasks	None	AOPSSID_MAXTHREADTASKS
aopsubd-max-thread-tasks	None	AOPSUBD_MAXTHREADTASKS
aopwsmd-max-thread-tasks	None	AOPWSMD_MAXTHREADTASKS
applid	None	AOP_APPLID
ascii-codepage	ascii-codepage (aopd.conf)	None

Table 11. Attributes in system configuration definition and the equivalent attributes in **aopd.conf** and **aopmsg.conf** and equivalent environment variables (continued)

Attribute in system configuration definition	Equivalent attribute in aopd.conf and aopmsg.conf files	Equivalent environment variable in aopstart EXEC
blank-truncation-classes	None	AOP_BLANK_TRUNCATION_CLASSES
console-name	console-name (aopd.conf)	None
ebcdic-codepage	ebcdic-codepage (aopd.conf)	None
hardcopy-messages	hardcopy-messages (aopmsg.conf)	None
hardcopy-message-list	hardcopy-message-list (aopmsg.conf)	None
ipp-port-number	ipp-port-number (aopd.conf)	None
ignore-dcf-routing-errors	None	AOP_IGNORE_DCF_ROUTING_ERRORS
job-prefix	job-prefix (aopd.conf)	None
lpd-port-number	lpd-port-number (aopd.conf)	None
log-retention	log-retention (aopd.conf)	None
logstream-name	None	None
mail-do-not-add-suffixes	None	AOP_MAIL_DO_NOT_ADD_SUFFIX
mail-preserve-suffixes	None	AOP_MAIL_PRESERVE_SUFFIXES
mail-use-first-address	None	AOP_MAIL_USE_FIRST_ADDRESS
mailer-path-name	None	AOPMAILER
mailer-options	None	AOPMAILER_OPTIONS
max-historical-inventory-size	max-historical-inventory-size (aopd.conf)	None
max-thread-tasks	None	AOP_MAXTHREADTASKS
netpool-use-fixed-jobid	None	AOPNETD_USE_FIXED_JOBID
netpool-use-unaltered-jobid	None	AOPNETD_USE_UNALTERED_JOBID
select-work-from-hold-queue	None	AOP_SELECT_WORK_FROM_HOLD_QUEUE
smf-recording	smf-recording (aopd.conf)	None
snmp-community	snmp-community (aopd.conf)	None
suppress-post-unix-filter-formatting	None	AOP_SUPPRESS_POST_UNIX_FILTER_FORMATTING
<p>Note: These attributes do not apply to IP PrintWay basic mode: aopoutd-max-thread-tasks, applid, ignore-dcf-routing-errors, mail-do-not-add-suffixes, mail-preserve-suffixes, mail-use-first-address, mailer-path-name, mailer-options, and select-work-from-hold-queue. To configure IP PrintWay basic mode, specify the equivalent environment variable (if one exists) in the IP PrintWay basic mode startup procedure, regardless of whether dynamic configuration is enabled.</p>		

Related information: For information about:

- How to view and edit the system configuration definition, see “Editing the system configuration definition for Printer Inventory Manager” on page 76.
- All attributes that you can specify in the system configuration definition, see Appendix D, “Configuration attributes,” on page 435.

Cross-system coupling facility (XCF)

When dynamic configuration is enabled, Infoprint Server uses the cross-system coupling facility (XCF) component of z/OS to communicate between daemons. Infoprint Server adds all daemons to an XCF group. The group name has this format: AOP

pppp

[q]:

AOP Infoprint Server identifier.

pppp Printer Inventory name. Any lowercase letters are converted to uppercase.

q XCF group qualifier that makes the group name unique when multiple instances of Infoprint Server are running in a sysplex with the same Printer Inventory name. This qualifier is optional.

Because the XCF group name must be unique in a sysplex, if multiple instances of Infoprint Server run in a sysplex with the same Printer Inventory name, you must specify a unique one-character group qualifier in the **xcf-group-qualifier** attribute in the **aopd.conf** file.

Related information: For information about the **xcf-group-qualifier** attribute, see “Creating the Infoprint Server configuration file (aopd.conf)” on page 34.

Steps for enabling dynamic configuration

These steps show how to enable dynamic configuration.

Before you begin:

- Make sure that you will not return to a release before z/OS V2R1. If you enable dynamic configuration and change attributes in the system configuration definition, the changes in the system configuration definition are lost when you return to a previous release.
- Make sure that all users who need to view and update the system configuration definition have access to the AOP.CONFIGURATION or AOP.ADMINISTRATOR resource profile in the RACF PRINTSRV class:
 - READ access to the AOP.CONFIGURATION or AOP.ADMINISTRATOR profile lets you view the system configuration definition.
 - UPDATE access to the AOP.CONFIGURATION profile lets you view and update the system configuration definition. If the AOP.CONFIGURATION profile does not exist, UPDATE access to the AOP.ADMINISTRATOR profile lets you view and update the system configuration definition.

If both the AOP.CONFIGURATION and AOP.ADMINISTRATOR profiles do not exist, READ access to the AOPADMIN profile in the RACF FACILITY class lets you view and update the system configuration definition.

- If the **log-retention** attribute is not specified in the **aopd.conf** file, make sure that the **/var/Printsrv** directory (or the directory specified in the **base-directory** attribute) has enough space for at least one day's worth of messages and historical print job information. This is because when you enable dynamic configuration, the default value for **log-retention** changes from 0 to 1 day.
- To edit the **aopd.conf** and **aopmsg.conf** files, you must be a member of the RACF group for Infoprint Server administrators (AOPADMIN) or have an effective UID of 0.
- To edit the **aopstart** EXEC, you must have an effective UID of 0.

1. Edit the Infoprint Server configuration file (**aopd.conf**) and save the file:
 - Specify this attribute:

dynamic-configuration = yes

- If more than one instance of Infoprint Server runs in a sysplex with the same Printer Inventory name, specify a unique one-character qualifier in the **xcf-group-qualifier** attribute. For example:

xcf-group-qualifier = 1

- (Optional) Comment out or remove the attributes in the **aopd.conf** and **aopmsg.conf** files that have equivalent attributes in the system configuration definition. This step is optional, but it prevents confusion about which attributes in the files Infoprint Server uses. When you start Infoprint Server after specifying **dynamic-configuration=yes**, Infoprint Server ignores the attributes in the **aopd.conf** and **aopmsg.conf** files that have equivalent attributes in the system configuration definition.

Tips:

- For a list of the dynamic attributes, see Table 11 on page 70.
- To comment out a dynamic attribute, type the # symbol in the first column of the line with the attribute.
- Consider making a backup copy of the **aopd.conf** and **aopmsg.conf** files before you comment out or remove attributes so that if you decide to disable dynamic configuration or return to a release before z/OS V2R1, you can restore the original files.

-
2. (Optional) Comment out or remove the environment variables in the **aopstart EXEC** that have equivalent attributes in the system configuration definition. This step is optional, but it prevents confusion about which environment variables Infoprint Server uses. When you start Infoprint Server after specifying **dynamic-configuration=yes**, Infoprint Server ignores the environment variables in the **aopstart EXEC** that have equivalent attributes in the system configuration definition.

Tip: Consider making a backup copy of the **aopstart EXEC** before you comment out or remove environment variables so that if you decide to disable dynamic configuration or return to a release before z/OS V2R1, you can restore the original **aopstart EXEC**.

-
3. If Infoprint Server is running, stop all daemons and wait for all daemons to stop.

Example: Enter this MVS command to run the AOPSTOP JCL procedure:
START AOPSTOP

-
4. Start Infoprint Server.

Example: Enter this MVS command to run the AOPSTART JCL procedure:
START AOPSTART

-
5. (Optional) To verify that dynamic configuration is enabled, use the Infoprint Server ISPF panels to view the system configuration definition in the Printer Inventory. If you can view the Infoprint Server System Configuration panel, dynamic configuration is enabled. You must have authority to view the system configuration definition.

-
6. If you use Infoprint Central, restart the z/OS HTTP Server.

Tip: You can restart the z/OS HTTP Server while Infoprint Server is stopped.

7. Use the Printer Inventory Definition Utility **pidu export** command to create a backup copy of the Printer Inventory so that if you need to restore the Printer Inventory, you can use a backup copy that you made after you enabled dynamic configuration. You must have authority to create a backup copy of the Printer Inventory.
-

Related information: For information about:

- How to set up security for the system configuration definition, see “Setting up security for the Printer Inventory and operator commands” on page 47.
- How to edit the **aopd.conf** file, see “Steps for editing the Infoprint Server configuration file (aopd.conf)” on page 42.
- How to edit the **aopmsg.conf** file, see “Editing the message configuration file (aopmsg.conf)” on page 87.
- Which attributes in the **aopd.conf** and **aopmsg.conf** files and which environment variables have equivalent attributes in the system configuration definition, see Table 11 on page 70.
- How to determine if dynamic configuration is enabled, see “Steps for determining whether dynamic configuration is enabled” on page 75.
- How to back up the Printer Inventory, see “Backing up and restoring the Printer Inventory” on page 105.

Steps for disabling dynamic configuration

These steps show how to disable dynamic configuration. When you disable dynamic configuration:

- If you made any changes to the system configuration definition while dynamic configuration was enabled, these changes are lost.
- When you restart Infoprint Server, Infoprint Server uses the configuration attributes specified in the **aopd.conf** and **aopmsg.conf** files and the environment variables set in the **aopstart** EXEC to create a new system configuration definition. You cannot modify this system configuration definition.
- Infoprint Server cannot write messages to an MVS system logger log stream. Instead, Infoprint Server writes messages to the z/OS UNIX file system specified in the **base-directory** attribute if the **log-retention** attribute in the **aopd.conf** file is set to at least 1 day or if you start the Infoprint Central daemon.

Before you begin:

- You must be a member of the RACF group for Infoprint Server administrators (AOPADMIN) or have an effective UID of 0.
- You must have READ access to the AOP.ADMINISTRATOR resource profile.

1. Run the **pidu** command to display all attributes in the system configuration definition.

Example: Enter this z/OS UNIX command to display the attributes:

```
pidu -c 'display configuration aopd.conf ; '
```

2. Edit the Infoprint Server configuration file (**aopd.conf**) and save the file:
 - Specify the **dynamic-configuration=no** attribute.

- Specify the configuration attributes that are currently specified in the system configuration definition. However, do not specify the **logstream-name** attribute because it is not allowed in the **aopd.conf** file.
 - Make sure that the **log-retention** attribute specifies at least 1 day. When dynamic configuration is disabled, the default for the **log-retention** attribute changes to 0 if the **aopssid** daemon is not specified in the **start-daemons** attribute.
-

3. If these attributes are specified in the system configuration definition, edit the **aopmsg.conf** file to specify them:

- **hardcopy-messages**
 - **hardcopy-message-list**
-

4. Edit the **aopstart** EXEC to specify any environment variables with equivalent attributes that are specified in the system configuration definition. Save the file.

5. If Infoprint Server is running, stop all daemons and wait for all daemons to stop.

Example: Enter this MVS command to run the AOPSTOP JCL procedure:

```
START AOPSTOP
```

6. Start Infoprint Server.

Example: Enter this MVS command to run the AOPSTART JCL procedure:

```
START AOPSTART
```

7. Use the Printer Inventory Definition Utility **pidu export** command to create a backup copy of the Printer Inventory so that if you need to restore the Printer Inventory, you can use a backup copy that you made after you disabled dynamic configuration.

Related information: For information about:

- How to run the **pidu** command, see *z/OS Infoprint Server Operation and Administration*.
- How to edit the **aopd.conf** file, see “Steps for editing the Infoprint Server configuration file (aopd.conf)” on page 42.
- How to edit the **aopmsg.conf** file, see “Editing the message configuration file (aopmsg.conf)” on page 87.
- Dynamic attributes, see “Dynamic attributes” on page 69.
- How to back up the Printer Inventory, see “Backing up and restoring the Printer Inventory” on page 105.

Steps for determining whether dynamic configuration is enabled

These steps show how to determine whether dynamic configuration is enabled.

1. If Infoprint Server is running, on the main Infoprint Server ISPF panel, select **8 System**.

```
Infoprint Server: Printer Inventory Manager
Option ==>

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

Infoprint Server Configuration
  7 ISPF         Manage ISPF panel configuration
  8 System       Manage system configuration
```

If dynamic configuration is enabled, you see the Infoprint Server System Configuration panel.

If dynamic configuration is not enabled, this error message is displayed on the panel:

```
AOPIM006 Option not enabled
```

2. If Infoprint Server is not running, view the **aopd.conf** file. If the **dynamic-configuraton=yes** attribute is set, dynamic configuration is enabled or will be enabled the next time you start Infoprint Server.

Related information: For information about:

- How to view and edit the **aopd.conf** file, see “Steps for editing the Infoprint Server configuration file (aopd.conf)” on page 42.

Editing the system configuration definition for Printer Inventory Manager

The system configuration definition in the Printer Inventory contains attributes that let you customize Infoprint Server while Infoprint Server is running. If dynamic configuration is enabled, you can specify these attributes in the system configuration definition to customize the Printer Inventory Manager:

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*.

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
- 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. For information about how to create conversion tables, see information about code set conversion utilities in *z/OS XL C/C++ Programming Guide*.

Infoprint Server uses the code page specified in this field in these ways:

- The Printer Inventory Manager displays this code page in the **Printer code page** field in ISPF panels for new IP PrintWay printer definitions that use the LPR, direct sockets, or IPP protocol to transmit data sets to the remote system.
- Print Interface uses this code page as the default source (document) code page when it converts data streams from ASCII to EBCDIC. A document code page specified in the printer definition or job attribute overrides this value.
- Print Interface uses this code page as the target (printer) code page when it converts PostScript documents that were created with a non-ASCII code page.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: `ascii-codepage = IS08859-1`

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*.

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
- 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. For information about how to create conversion tables, see information about code set conversion utilities in *z/OS XL C/C++ Programming Guide*.

Infoprint Server uses this code page in these ways:

- The Printer Inventory Manager displays this code page in the **Printer code page** field in ISPF panels for new PSF printer definitions, general printer definitions, and IP PrintWay printer definitions that use the VTAM or email protocol to transmit data sets to the remote system.
- NetSpool uses this code page as the default source (document) code page when it converts data streams from EBCDIC to ASCII. A document code page in the printer definition or job attribute overrides this value.
- IP PrintWay extended mode uses this code page as the default source (document) code page when it converts data streams from EBCDIC to ASCII. A document code page in the printer definition overrides this value.
- Print Interface uses this code page as the source (document) code page when Print Interface cannot determine the user locale for the **Ip** command.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server. Also restart the NetSpool started task.

Default: ebcddic-codepage = IBM-1047

log-retention = days

The number of days worth of messages that Infoprint Central displays for print jobs and printers. Also, the number of days for which Infoprint Central displays print jobs that have been removed from the JES spool.

If you do not use the MVS system logger for the common message log, this attribute also determines how many days worth of messages Infoprint Server retains in the z/OS UNIX file system. If you use the system logger for messages, the retention period specified in the system logger log stream definition is used instead.

You can specify a value from 0 – 59. A value of 0 means that Infoprint Central cannot display any messages and cannot display information about print jobs that have been removed from the JES spool. In addition, if you do not use the system logger for messages, Infoprint Server does not write any messages in the z/OS UNIX file system.

Guidelines:

- If you do not use the system logger for messages, specify a value greater than 0 or use the default of 1 day, so that Infoprint Server writes messages in a z/OS UNIX file system.
- If you use the system logger for messages and also use Infoprint Central, the value should be less than or equal to the retention period specified in the log stream definition. If you specify a higher value than the retention period for the log stream, you might not be able to see the messages for all the historical print jobs that Infoprint Central displays.
- If you use the system logger but do not use Infoprint Central, you can specify 0 to save space in the z/OS UNIX file system.
- If you specify a value greater than 0 or use the default of 1 day, you might need to increase the amount of space available to the file system mounted at the **/var** or **/var/Printsrv** mount point because the common message log and Historical Inventory can contain a large amount of data. See “Calculating DASD space requirements for the **/var/Printsrv** directory” on page 32.
- Because the common message log and Historical Inventory can contain a large amount of data, start with a value of 1 day. Increase the value by 1 day if there is enough space in the file system and if users request to see more messages or historical information.
- If the size of the common message log in the z/OS UNIX file system grows too large, you can decrease its size. For more information, see “Reducing the size of the common message log in the z/OS UNIX file system” on page 111.
- If you use Infoprint Central to view messages, set the TZ and either the LC_TIME or LC_ALL environment variables. These variables affect the

date and time displayed in messages. See “Setting environment variables for the Printer Inventory Manager” on page 43.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: log-retention = 1

Example: log-retention = 3

max-historical-inventory-size = megabytes

The maximum size in megabytes (MBs) of the Historical Inventory, **/var/Printsrv/hinv**. The Historical Inventory contains information about print jobs and documents (output data sets) 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. However, the number of days specified in the **log-retention** attribute limits the amount of information that Infoprint Server keeps in the Historical Inventory.

Table 12 shows the number of documents one MB in the Historical Inventory can contain.

Table 12. Number of documents in the Historical Inventory

If print jobs on the JES spool contain:	One MB in the Historical Inventory contains:
1 document in each print job	1350 documents
3 documents in each print job	1750 documents
5 documents in each print job	1850 documents
10 documents in each print job	1950 documents

Guideline: Use this formula to estimate the maximum size of the Historical Inventory:

$$\text{Maximum size of Historical Inventory} = \frac{\text{(Documents processed in one day X Days to retain information)}}{\text{Documents in one MB}}$$

For example:

- Documents processed in one day = 100,000
- Days to retain information = 3 (**log-retention** = 3)
- Documents in one MB = 1350 documents

The maximum size of the Historical Inventory = (100,000 X 3) / 1350 = 223 MB (rounded up)

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.

- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: max-historical-inventory-size = 10

Example: max-historical-inventory-size = 300

Example: This screen shows a portion of the System Configuration ISPF panel that you use to edit the system configuration definition for the Printer Inventory Manager:

```

                                Infoprint Server: System Configuration

Code Pages:
  ASCII . . IS08859-1
  EBCDIC . . IBM-1047

Messages:
  Log retention period (days) . . . . . 1
  Maximum Historical Inventory size (MBs) . 10
```

Related information: For information about:

- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 68.
- How to edit the system configuration definition, see “Steps for editing the system configuration definition.”

Steps for editing the system configuration definition

You can use the Infoprint Server ISPF panels or the Printer Inventory Definition Utility (PIDU) to edit the system configuration definition while Infoprint Server is running. These steps show how to use the ISPF panels to view and edit the system configuration definition.

You cannot create the system configuration definition. Infoprint Server creates it automatically when you enable dynamic configuration.

Before you begin:

- Dynamic configuration must be enabled.
- Infoprint Server must be started.
- You must have UPDATE access to the AOP.CONFIGURATION resource profile in the RACF PRINTSRV class. If the AOP.CONFIGURATION profile does not exist, you must have UPDATE access to the AOP.ADMINISTRATOR profile in the PRINTSRV class.

If both the AOP.CONFIGURATION and AOP.ADMINISTRATOR profiles do not exist, you must have READ or higher access to the AOPADMIN profile in the RACF FACILITY class.

1. On the main Infoprint Server ISPF panel, select **8 System** and press Enter. Option 8 is visible only if dynamic configuration is enabled.


```

                                Infoprint Server: Printer Inventory Manager
Option ==>

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

Infoprint Server Configuration
  7 ISPF              Manage ISPF panel configuration
  8 System            Manage system configuration

```

2. On the Infoprint Server System Configuration panel:

- a. View the non-editable field on the panel under the “Startup information” heading. To display help information for a field, place the cursor on the value area of the field and press the HELP function key.
- b. Edit any of the editable fields. In the **Other** field, you can specify other attributes that your IBM service representative tells you to specify. To display help information for a field, place the cursor on the value area of the field and press the HELP function key.
- c. (Optional) To validate the fields before you save the panel, press Enter. If you see a message that says you need to restart one or more Infoprint Server daemons before a change can take effect, note the daemon or daemons that you need to restart.
- d. Do one of these to save the system configuration definition or exit without saving the definition:
 - To save the panel but keep the panel on the screen, type SAVE on the command line and press Enter.
 - To save the panel and exit, type END on the command line and press Enter or press the END function key.
 - To exit without saving the panel, type CANCEL on the command line and press Enter.

If you need to restart an Infoprint Server daemon before a change can take effect, you see a message that tells you which daemons you need to restart. If you do not see a message, changes to fields on the panel take effect as soon as you save the system configuration definition.

3. Stop and restart these daemons if necessary:

- If you changed the **Extended MCS console name** field and the Infoprint Central daemon (**aopssid**) is running, stop the daemon and start Infoprint Server. When you start Infoprint Server, the Infoprint Central daemon is restarted; however, other Infoprint Server daemons continue to run uninterrupted.
- If you changed the **Select work from hold queue** field and the IP PrintWay is running, stop the IP PrintWay daemons (**aopoutd** and **aopwsmd**) and start Infoprint Server. When you start Infoprint Server, the IP PrintWay daemons are restarted; however, other Infoprint Server daemons continue to run uninterrupted.

- If you changed the **SNMP community name** field and the SNMP subagent daemon (**aopsnmpd**) is running, stop the daemon and start Infoprint Server. When you start Infoprint Server, the SNMP subagent daemon is restarted; however, other Infoprint Server daemons continue to run uninterrupted.
-

4. Use the Printer Inventory Definition Utility **pidu export** command to back up the Printer Inventory. The **export** command saves the system configuration definition in the backup copy of the Printer Inventory. To make a backup copy of the Printer Inventory, you must meet both of these requirements:
 - Be a member of the RACF group for Infoprint Server administrators (AOPADMIN) or have an effective UID of 0
 - Have READ access to the AOP.ADMINISTRATOR resource profile
-

This screen shows the Infoprint Server System Configuration ISPF panel:

Infoprint Server: System Configuration

```

Startup Information:
 / Dynamic configuration
 / Resolve IP PrintWay printers
Base directory . . . . . /var/Printsrv          (extend)
Printer Inventory name . . . . . AOP1
JES subsystem name . . . . .
XCF group name . . . . . AOPAOP1
Started daemons:
    ippd      / lpd      netd      outd
    snmpd     ssid      subd      xfd

General:
_ Allow all characters in line data
Job ID prefix . . PS

Code Pages:
ASCII . . ISO8859-1
EBCDIC . . IBM-1047

Messages:
Log retention period (days) . . . . . 1
Maximum Historical Inventory size (MBs) . 10
Log stream name . . . . .
Send messages to hardcopy log . . . . . 3 1. All 2. List 3. None
Message list for hardcopy log . . . . . (more)

Infoprint Central:
Extended MCS console name . . AOP1xxxx

IP PrintWay (extended mode):
_ Ignore DEST, CLASS, FORMS routing errors
/ Select work from hold queue
_ Suppress formatting after UNIX filter
/ Write SMF records
APPL ID . . . . .
Blank truncation classes . . -----
-----

Email:
_ Send email to address of first document
_ Do not add suffixes to files
Preserve suffixes . . . . . (more)
Application path . . . . /bin/sendmail (extend)
Application options . . . . (extend)

Print Interface:
IPP port number . . 631
LPD port number . . 515

SNMP Subagent:
Community name . . public

Other:
_____ (more)

```

Related information: For information about:

- How to set up security for the system configuration definition, see “Setting up security for the Printer Inventory and operator commands” on page 47.
- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 68.
- How to back up the Printer Inventory, see “Backing up and restoring the Printer Inventory” on page 105.
- How to use PIDU to edit the system configuration definition, see *z/OS Infoprint Server Operation and Administration*.
- How to stop daemons and start Infoprint Server, see *z/OS Infoprint Server Operation and Administration*.

- Attributes in the system configuration definition, see Appendix D, “Configuration attributes,” on page 435.

Sending non-console messages to the hardcopy log

Infoprint Server sends all console messages to the z/OS hardcopy log. (Console messages are messages that require the attention of an operator or administrator.) You can configure Infoprint Server to also send all or selected non-console messages to the hardcopy log. You might want to send non-console messages to the hardcopy log so that you can use a z/OS message automation facility to process them.

The hardcopy log can be the system log (SYSLOG) or the operations log (OPERLOG), or both, depending on how you configured the hardcopy medium in the HARDCOPY statement in the CONSOLxx Parmlib member.

Limitation: Infoprint Server cannot send non-console messages from IP PrintWay basic mode, the ISPF panels, or the Printer Inventory Definition Utility (PIDU) to the hardcopy log.

You can configure Infoprint Server to send messages to the hardcopy log in one of these ways:

- If dynamic configuration is enabled, specify the message attributes in the system configuration definition in the Printer Inventory.
- If dynamic configuration is not enabled, specify the message attributes in the Infoprint Server message configuration file (**aopmsg.conf**).

To send all or selected non-console messages to the hardcopy log, specify these attributes:

hardcopy-messages = all | list | none

Specifies which additional messages Infoprint Server sends to the hardcopy log:

- 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.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopmsg.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: hardcopy-messages = none

Example: hardcopy-messages = list

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.

Tips:

- You do not need to specify messages that Infoprint Server sends to the console, such as messages with the API prefix, because these messages are automatically sent to the hardcopy log.
- If your installation uses the MPFLSTxx Parmlib member for message automation, you might want to specify the same message IDs in MPFLSTxx.
- If your installation uses an automated operations tool, the tool can process these messages.

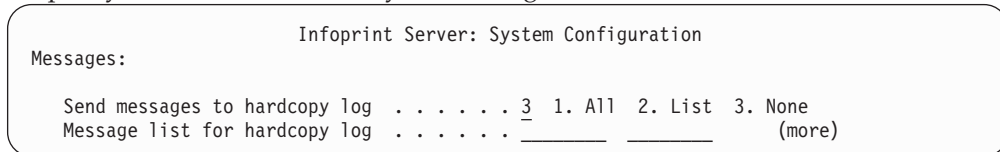
Rules:

- If you specify this attribute, also specify **hardcopy-messages = list**.
- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopmsg.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: None.

Example: `hardcopy-message-list={AOP3614I AOP3803E}`

This screen shows a portion of the System Configuration ISPF panel that you use to specify the attributes in the system configuration definition:



Messages sent to the hardcopy log have this format:

```
date time priority:level user:id [job:id] [filename:name] [dsn:name]
[output_device:name] [job_selection_rule:name] [fsa:name] [luname:name]
[jes_jobname:name] system:name inventory:name [program:name]
msg:message
```

For information about the fields after the message text, see the description of the output from the **aoplogu** command in *z/OS Infoprint Server Messages and Diagnosis*.

If necessary, Infoprint Server truncates messages to six lines with up to 70 characters in each line. Infoprint Server uses the system default routing and descriptor codes.

Related information: For information about:

- How to create the message configuration file (**aopmsg.conf**), see “Creating the message configuration file (aopmsg.conf)” on page 86.
- How to edit an existing message configuration file (**aopmsg.conf**), see “Editing the message configuration file (aopmsg.conf)” on page 87.
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 80.

- The CONSOLxx Parmlib member, see *z/OS MVS Initialization and Tuning Reference*.
- z/OS message automation, see *z/OS MVS Planning: Operations*.

Creating the message configuration file (aopmsg.conf)

These steps show the z/OS UNIX commands to use to create the message configuration file (**aopmsg.conf**) in the **/etc/Printsrv** directory, which is the default directory for Infoprint Server configuration files.

To edit an existing **aopmsg.conf** file, follow the steps in “Editing the message configuration file (aopmsg.conf)” on page 87.

Note: If dynamic configuration is enabled, do not create this file. Instead, specify the attributes in the system configuration definition in the Printer Inventory.

Before you begin: You must have permission to create files in the **/etc/Printsrv** directory. If you did not change the owner and permissions for **/etc/Printsrv** after installation, you must have an effective UID of 0 to create files in this directory.

1. On the z/OS UNIX command line, switch to an effective UID of 0:

```
su
```

To use the **su** command, you must be permitted to the BPX.SUPERUSER profile in the FACILITY class in RACF.

2. Copy the sample message configuration file, **/usr/lpp/Printsrv/samples/aopmsg.conf**, to **/etc/Printsrv/aopmsg.conf**:

```
cp /usr/lpp/Printsrv/samples/aopmsg.conf /etc/Printsrv/aopmsg.conf
```

Tips:

- If you copy the sample configuration file to a different location, specify the full path name of the configuration file in the AOPMSG_CONF environment variable in the **aopstart** EXEC and in the **/etc/profile** file. For information, see “Setting environment variables for the Printer Inventory Manager” on page 43.
- You can use the sample job that is provided in SYS1.SAMPLIB(AOPCPETC) to copy all Infoprint Server sample configuration files, including the message configuration file, from **/usr/lpp/Printsrv/samples** to the **/etc/Printsrv** directory. Review the comments in the sample job to determine which configuration files you require. A user with an effective UID of 0 should run this job.

3. Change the owner of the file to UID 0:

```
chown 0 /etc/Printsrv/aopmsg.conf
```

4. Change the group owner to AOPADMIN. AOPADMIN is the suggested RACF group name for Infoprint Server administrators. However, you can use a different name for this group (see “AOPADMIN and AOOPER groups” on page 49).

```
chgrp AOPADMIN /etc/Printsrv/aopmsg.conf
```

5. Change the permissions of the file so that it is readable by everyone and writeable only by UID 0 and members of the AOPADMIN group.

```
chmod 664 /etc/Printsrv/aopmsg.conf
```

6. Edit the message configuration file using your preferred editor and save your changes. For example, enter:

```
oedit /etc/Printsrv/aopmsg.conf
```

Edit this file using the IBM-1047 code page. For the syntax rules for editing the file, see “Syntax rules”.

7. If you switched to an effective UID of 0 in step 1, switch back to your own UID:

```
exit
```

8. If Infoprint Server is running, stop and restart all Infoprint Server daemons. Be sure to wait for all daemons to stop before restarting. For example, enter these MVS commands to run the AOPSTOP and AOPSTART JCL procedures:

```
START AOPSTOP  
START AOPSTART
```

Syntax rules: When you edit this file, follow these syntax rules:

- Use lowercase characters for the attribute names. Use uppercase or lowercase characters for the values.
- Use blank characters before or after the equal sign, if desired.
- Start comments with the # symbol.
- Include blank lines, if desired.

Related information: For information about:

- All the attributes that you can specify in the **aopmsg.conf** file, see “aopmsg.conf -- Infoprint Server message configuration file” on page 401.
- How to edit an existing **aopmsg.conf** file, see “Editing the message configuration file (aopmsg.conf).”
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 80.

Editing the message configuration file (aopmsg.conf)

These steps show the z/OS UNIX commands to use to edit an existing **aopmsg.conf** file.

Note: If dynamic configuration is enabled, do not edit this file. Instead, edit the attributes in the system configuration definition in the Printer Inventory.

Before you begin: You must be a member of the RACF group for Infoprint Server administrators (AOPADMIN) or have an effective UID of 0.

To edit the **aopmsg.conf** file:

1. Edit the message configuration file using your preferred editor and save your changes. For example, enter:

```
oedit /etc/Printsrv/aopmsg.conf
```

Edit this file using the IBM-1047 code page. For the syntax rules for editing the file, see “Syntax rules”.

Tip: You can select the ISPF “z/OS UNIX Shell” option (ISPF 13.11) to edit z/OS UNIX files.

-
2. If Infoprint Server is running, stop and restart all Infoprint Server daemons. For example, enter these MVS commands to run the AOPSTOP and AOPSTART JCL procedures:

```
START AOPSTOP
START AOPSTART
```

Syntax rules: When you edit this file, follow these syntax rules:

- Use lowercase characters for the attribute names. Use uppercase or lowercase characters for the values.
- Use blank characters before or after the equal sign, if desired.
- Start comments with the # symbol.
- Include blank lines, if desired.

Related information: For information about:

- All the attributes that you can specify in the **aopmsg.conf** file, see “aopmsg.conf -- Infoprint Server message configuration file” on page 401.
- How to create the **aopmsg.conf** file, see “Creating the message configuration file (aopmsg.conf)” on page 86.
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 80.

Setting up the MVS system logger for the common message log

You can set up the MVS system logger for the Infoprint Server common message log. All components of Infoprint Server except for IP PrintWay basic mode write messages to the common message log. In addition, IBM transform products and PSF V4R4 (and later) can write messages to the common message log.

Table 13 summarizes the customization tasks for setting up the MVS system logger environment for the common message log. Required tasks are required by all installations that want to use the system logger. Optional tasks are required only if the listed condition applies to your installation.

Table 13. Customization tasks for system logger

Task	Condition	See page
Deciding whether to use the MVS system logger	Required	89
Deciding the type of log stream	Required	89
Deciding on a naming convention for log streams	Required	90
Determining how long to retain messages	Required	90
Determining the size of log-stream data sets and interim storage	Required	91
Setting up security for log streams	Required	91
Defining log streams to the MVS system logger	Required	94

Table 13. Customization tasks for system logger (continued)

Task	Condition	See page
Activating a log stream in Infoprint Server	Required	99
Switching from one log stream to another	Optional: To switch to a new log stream	101
Deactivating a log stream in Infoprint Server	Optional: To deactivate a log stream	102
Handling system logger errors	Optional: To handle any logger errors	103
Viewing messages in the log stream	Optional: To view messages	104

Deciding whether to use the MVS system logger

The Infoprint Server common message log can reside in one of these locations:

- Log files in the z/OS UNIX file system
- An MVS system logger log stream

If you don't set up the MVS system logger for the common message log, the log, by default, resides in the z/OS UNIX file system specified in the **base-directory** attribute in the **aopd.conf** configuration file provided that the value of the **log-retention** attribute is at least 1 day.

Advantages of setting up the MVS system logger for the common message log are:

- You can retain more messages in a system logger log stream because the system logger can automatically offload log records to auxiliary storage when the log stream is nearing capacity. If you retain a lot of messages in the z/OS UNIX file system, you can run out of space in the file system.
- If Infoprint Server runs on more than one system in a sysplex, all instances of Infoprint Server can write messages to the same log stream (a coupling-facility log stream is required). You can use the Infoprint Server **aoplogu** command to view merged messages from all systems.

One disadvantage of using the system logger is that it might take more time for Infoprint Central and the **aoplogu** command to retrieve and display messages from a log stream.

To use the MVS system logger, you must meet the MVS system logger requirements. In addition, the Infoprint Server dynamic configuration function must be enabled.

Related information: For information about

- MVS system logger, see *z/OS MVS Setting Up a Sysplex*
- Enabling dynamic configuration, see "Enabling dynamic configuration" on page 68

Deciding the type of log stream

You can define either of these types of log streams:

- DASD-only log stream
- Coupling-facility log stream

If only one instance of Infoprint Server runs in a sysplex, for simpler administration, you might want to define a DASD-only log stream. However, you can also define a coupling-facility log stream.

If more than one instance of Infoprint Server runs in a sysplex, you can define:

- A separate DASD-only log stream, or a separate coupling-facility log stream, for each instance of Infoprint Server
- A coupling-facility log stream that all (or some) instances of Infoprint Server share

If one or more instances of Infoprint Server run in a sysplex, consider defining a coupling-facility log stream and share the same log stream among all instances of Infoprint Server. This lets you use the Infoprint Server **aoplogu** command to view merged messages from all the systems. For example, if Infoprint Server transforms a document on one system and prints the document on another system, the **aoplogu** command lets you view all messages for that document regardless of which system wrote the message. However, even if several instances of Infoprint Server share the same log stream, Infoprint Central can only display messages from the system on which Infoprint Central is running.

Related information: For information about the types of log streams, see *z/OS MVS Setting Up a Sysplex*.

Deciding on a naming convention for log streams

You can use any log stream name that follows the MVS system logger guidelines because Infoprint Server does not use a fixed name for the log stream. The system logger requires that each log stream name be unique in a sysplex.

Tip: To make sure that a log stream name is unique, use "AOP" as the first qualifier for Infoprint Server log stream names.

Examples:

- If you plan to define only one DASD-only or one coupling-facility log stream, you could use this naming convention:

AOP.MSG

- If you plan to define a separate DASD-only log stream, or a separate coupling-facility log stream, for each instance of Infoprint Server in a sysplex, you could use this naming convention:

AOP.MSG.SYSA

AOP.MSG.SYSB

The third qualifier of the log-stream name (SYSA, SYSB) identifies the system.

Related information: For information about system logger naming requirements, see *z/OS MVS Setting Up a Sysplex*.

Determining how long to retain messages

Infoprint Server does not delete any messages from the MVS system logger log stream. Therefore, when you define a log stream to the system logger, you must select the system logger automatic delete function and specify how long you want the system logger to retain the log-stream data sets that contain messages.

Define the log stream to the system logger with these parameters when you run the system logger IXCMIAPU utility:

- AUTODELETE(YES): Specifies that the system logger automatically deletes messages when the retention period specified in the RETPD parameter expires.
- RETPD(*dddd*): Specifies that the system logger retains messages in the log stream for *dddd* days. The system logger deletes an entire log data set when all data is older than the retention period (RETPD) specified for the log stream.

Guideline: Specify a RETPD value that is equal to or greater than the value specified in the **log-retention** attribute in the Infoprint Server system configuration definition. For example, if **log-retention=2** , specify a RETPD value of at least 2 days.

Determining the size of log-stream data sets and interim storage

When you define a log stream, you can specify the size of the log-stream interim storage, staging data set, and offload data sets. To determine the optimum sizes, follow the guidelines in these publications:

- *z/OS MVS Setting Up a Sysplex*
- IBM redbooks *System Programmer's Guide to: System Logger*, SG24-6898, at <http://www.ibm.com/redbooks>

You can use the Coupling Facility Structure Sizer Tool (CFSizer) to determine the values for some of the size parameters. You can find the CFSizer tool at <http://www.ibm.com/servers/eserver/zseries/cfsizer>.

Tip: If you need to determine the number of bytes that Infoprint Server writes to the common message log in a specified time period, you can use the Infoprint Server **aoplogu** command to retrieve all messages written in the time period and pipe the results to the **wc -c** command to count the number of bytes.

Example: To determine the number of bytes written to the common message log in one day, enter on the z/OS UNIX command line:

```
aoplogu -l 1:00:00 | wc -c
```

Related information: For information about:

- The **aoplogu** command, see *z/OS Infoprint Server Operation and Administration*
- The **wc** command, see *z/OS UNIX System Services Command Reference*

Setting up security for log streams

Before you can define a log stream to the MVS system logger, you must define one or more profiles to RACF to protect the log stream. You must also authorize system logger administrators and Infoprint Server users to the RACF profiles.

Note: System logger administrators might need access to other system logger profiles that are not described in this section. For information, see *z/OS MVS Setting Up a Sysplex*.

Setting up security for a DASD-only log stream

These steps show how to define a log-stream resource profile to RACF for a DASD-only log stream and give users access to the profile. You need to define a separate log-stream resource profile for each DASD-only log stream that you plan to define.

1. Define a resource profile for the log stream to RACF in the LOGSTRM class.

Example: This RACF command defines a resource profile named AOP.MSG:
RDEFINE LOGSTRM (AOP.MSG) UACC(NONE)

The name of the resource profile (AOP.MSG in the example) must match the log stream name specified in the NAME parameter of the DEFINE LOGSTREAM statement in the LOGR policy. If you plan to define more than one log stream starting with AOP, you might want to define a resource profile with the generic name AOP*.

-
2. Give system logger administrators who run the IXCMIAPU utility ALTER access to the log-stream resource profile. IXCMIAPU is used to define, update, and delete log streams.

Example: This RACF command gives a user ALTER access to profile AOP.MSG:

```
PERMIT AOP.MSG CLASS(LOGSTRM) ACCESS(ALTER) ID(userID)
```

-
3. Give Infoprint Server operators who start Infoprint Server UPDATE access to the log-stream resource profile. This allows Infoprint Server to read and write messages. Because users who start Infoprint Server must be members of the RACF group for Infoprint Server operators (AOPOPER), you can give the AOPOPER group UPDATE access to the log-stream resource profile.

Example: This RACF command gives the AOPOPER group UPDATE access to profile AOP.MSG:

```
PERMIT AOP.MSG CLASS(LOGSTRM) ACCESS(UPDATE) ID(AOPOPER)
```

Tip: If you always use the AOPSTART procedure and not the **aopstart** command to start Infoprint Server, instead of giving the AOPOPER group access to the profiles, you can give the user ID that is associated with the AOPSTART procedure access to the profile.

-
4. Give Infoprint Server administrators who use the **aoplogu** command READ access to the log-stream resource profile. The **aoplogu** command lets you view messages in the common message log. Because users of the **aoplogu** command must be members of the RACF group for Infoprint Server administrators (AOPADMIN), you can give the AOPADMIN group READ access.

Example: This RACF command gives the AOPADMIN group READ access to profile AOP.MSG:

```
PERMIT AOP.MSG CLASS(LOGSTRM) ACCESS(READ) ID(AOPADMIN)
```

Tip: Infoprint Central displays messages for a selected print job or printer. However, Infoprint Central users do not need access to the log-stream profile because Infoprint Central checks that users are authorized to view the print job or the printer before displaying messages for the print job or printer.

-
5. (Optional) If you copy profiles in the LOGSTRM class into virtual storage, refresh the class after you define new profiles or permit new users to profiles in the class.

Example: This RACF command refreshes the LOGSTRM class:

```
SETROPTS RACLIST(LOGSTRM) REFRESH
```

Related information: For information about:

- System logger security, see *z/OS MVS Setting Up a Sysplex*.
- Infoprint Central security, see “Setting up security for Infoprint Central” on page 341.

Setting up security for a coupling-facility log stream

These steps show how to define a log-stream resource profile and a structure resource profile to RACF for a coupling-facility log stream and give users access to the profiles.

1. Define a resource profile to RACF in the LOGSTRM class for the coupling-facility log stream.

Example: This RACF command defines a resource profile named AOP.MSG:

```
RDEFINE LOGSTRM (AOP.MSG) UACC(NONE)
```

The name of the resource profile (AOP.MSG in the example) must match the log stream name specified in the NAME parameter of the DEFINE LOGSTREAM statement in the LOGR policy. If you plan to define more than one log stream starting with AOP, you might want to define a resource profile with the generic name AOP.*.

-
2. Define a resource profile to RACF in the FACILITY class to protect the structure for the log stream.

Example: This RACF command defines a structure resource profile named IXLSTR.AOP_MSG_STR:

```
RDEFINE FACILITY (IXLSTR.AOP_MSG_STR) UACC(NONE)
```

The profile name (IXLSTR.AOP_MSG_STR in the example) is “IXLSTR.” followed by the structure name. You specify the structure name (AOP_MSG_STR) in the NAME parameter of the DEFINE STRUCTURE statement in the LOGR policy.

-
3. Give the system logger administrator who runs the IXCMIAPU utility to define, update, and delete log streams ALTER access to the log-stream resource profile and UPDATE access to the structure resource profile.

Example: These RACF commands give a user ALTER access to profile AOP.MSG and UPDATE access to profile IXLSTR.AOP_MSG_STR:

```
PERMIT AOP.MSG CLASS(LOGSTRM) ACCESS(ALTER) ID(userID)
PERMIT IXLSTR.AOP_MSG_STR CLASS(FACILITY) ACCESS(UPDATE) ID(userID)
```

-
4. Give the system logger administrator who runs the IXCMIAPU utility to define and delete structures ALTER access to the structure resource profile.

Example: This RACF command gives a user ALTER access to profile IXLSTR.AOP_MSG_STR:

```
PERMIT IXLSTR.AOP_MSG_STR CLASS(FACILITY) ACCESS(ALTER) ID(userID)
```

-
5. Give Infoprint Server operators who can start Infoprint Server UPDATE access to the log-stream resource profile. This allows Infoprint Server to read and write messages. Because users who start Infoprint Server must be members of the RACF group for Infoprint Server operators (AOPOPER), you can give the AOPOPER group UPDATE access to the log-stream resource profile.

Example: This RACF command gives the AOPOPER group UPDATE access to profile AOP.MSG:

```
PERMIT AOP.MSG CLASS(LOGSTRM) ACCESS(UPDATE) ID(AOPOPER)
```

Tip: If you always use the AOPSTART procedure and not the **aopstart** command to start Infoprint Server, instead of giving the AOPOPER group access to the profiles, you can give the user ID that is associated with the AOPSTART procedure access to the profile.

6. Give Infoprint Server administrators who use the **aoplogu** command READ access to the log-stream resource profile. The **aoplogu** command lets you view messages in the common message log. Because users of the **aoplogu** command must be members of the RACF group for Infoprint Server administrators (AOPADMIN), you can give the AOPADMIN group READ access.

Example: This RACF command gives the AOPADMIN group READ access to profile AOP.MSG:

```
PERMIT AOP.MSG CLASS(LOGSTRM) ACCESS(READ) ID(AOPADMIN)
```

Tip: Infoprint Central displays messages for a selected print job or printer. However, Infoprint Central users do not need access to the log-stream profile because Infoprint Central checks that users are authorized to view the print job or the printer before displaying messages for the print job or printer.

7. (Optional) If you copy profiles in the LOGSTRM and FACILITY classes into virtual storage, refresh the classes after you define new profiles or permit new users to profiles in the class.

Example:

```
SETROPTS RACLIST(LOGSTRM) REFRESH  
SETROPTS RACLIST(FACILITY) REFRESH
```

Related information: For information about:

- MVS system logger security, see *z/OS MVS Setting Up a Sysplex*.
- Infoprint Central security, see “Setting up security for Infoprint Central” on page 341.

Defining log streams to the MVS system logger

These topics describe how to define DASD-only and coupling-facility log streams for the common message log to the MVS system logger. These topics do not include all the information you need to define log stream. For complete information, see *z/OS MVS Setting Up a Sysplex*.

Defining a DASD-only log stream to the MVS system logger

This topic describes, in general, how to define a DASD-only log stream for the common message log to the MVS system logger.

Before you begin:

- Define the system logger LOGR policy.
 - Set up security for the log stream.
1. Define the log stream in the LOGR policy. Table 14 on page 95 shows some of the parameters that you specify to define a log stream. Figure 12 on page 96 shows sample JCL for running the logger data administrative utility, IXCMIAFU, to define a DASD-only log stream.
-

2. (Optional) Verify that the log stream has been defined to the MVS system logger.

Example: This MVS command displays the status of log stream AOP.MSG:
 DISPLAY LOGGER,L,LSN=AOP.MSG

Table 14 lists the log-stream parameters for DASD-only log streams that have special considerations for Infoprint Server. For parameters that are not shown in Table 14, you can specify any values that are suitable for your installation.

Table 14. DASD-only log-stream parameters for Infoprint Server

Parameter	Meaning	Suggested value
NAME	Name of the log stream	See "Deciding on a naming convention for log streams" on page 90.
DASDONLY	Type of log stream	DASDONLY(YES)
RETPD	Number of days that the system logger retains messages before deleting them	See "Determining how long to retain messages" on page 90.
AUTODELETE	Automatic delete option	AUTODELETE(YES)
MAXBUFSIZE	Size, in bytes, of the largest log block that can be written to the log stream	MAXBUFSIZE(8192)
LS_SIZE	Size, in 4K blocks, of the offload data sets for the log stream	See "Determining the size of log-stream data sets and interim storage" on page 91.
STG_SIZE	Size, in 4K blocks, of the staging data sets for the log stream	See "Determining the size of log-stream data sets and interim storage" on page 91.
HIGHOFFLOAD	High offload threshold	HIGHOFFLOAD(80)
LOWOFFLOAD	Low offload threshold	LOWOFFLOAD(0)
DIAG	Allows collection of diagnostic information	YES

Figure 12 on page 96 shows sample JCL to define a DASD-only log stream named AOP.MSG to the system logger.


```

//STEP1 EXEC PGM=IXCMIAPU
//SYSPRINT DD SYSOUT=A
//*****
/* Define DASD-only log streams to the system logger. The      *
/* parameter values are for illustration only. Substitute values *
/* appropriate to your installation.                            *
//*****
//SYSIN DD *
        DATA TYPE(LOGR)
        DEFINE LOGSTREAM
        NAME(AOP.MSG)
        DESCRIPTION(Infoprint_Server)
        DASDONLY(YES)
        LS_SIZE(size)
        STG_SIZE(size)
        HIGHOFFLOAD(80)
        LOWOFFLOAD(0)
        MAXBUFSIZE(8192)
        RETPD(25)
        AUTODELETE(YES)
        DIAG(YES)
        HLQ(IXGLOGR)

/*
//

```

Note: The HLQ and EHLQ parameters are mutually exclusive. The HLQ parameter specifies the high-level qualifier for log-stream data set names (IXGLOGR is the default).

Figure 12. Sample JCL to define a DASD-only log stream to the system logger

Related information: For information about:

- How to define the LOGR policy and run the IXCMIAPU utility, see *z/OS MVS Setting Up a Sysplex*.
- How to set up security for a log stream, see “Setting up security for a DASD-only log stream” on page 91.
- The DISPLAY LOGGER command, see *z/OS MVS System Commands*.

Defining a coupling-facility log stream to the MVS system logger

This topic describes, in general how to define a coupling-facility log stream and structure for the common message log to the MVS system logger.

Before you begin:

- Define the system logger LOGR policy.
 - Define the system logger CFRM policy.
 - Set up security for the log stream.
1. Define the structure in the LOGR policy. Table 15 on page 97 shows some of the parameters that you specify to define a structure in the LOGR policy. Figure 13 on page 98 shows sample JCL to run the administrative data utility, IXCMIAPU, to define a structure in the LOGR policy.
-
2. Define the structure in the CFRM policy. Figure 14 on page 99 shows sample JCL to run the IXCMIAPU utility to define a structure in the CFRM policy.
-

3. Define the log stream in the LOGR policy. Table 16 shows some of the parameters that you specify to define a log stream in the LOGR policy. Figure 15 on page 99 shows sample JCL to run the IXCMIAPU utility to define the log stream.

4. (Optional) Verify that the log stream has been defined to the MVS system logger.

Example: This MVS command displays the status of log stream AOP.MSG:

```
DISPLAY LOGGER,L,LSN=AOP.MSG
```

Table 16 lists the coupling-facility structure parameters and the suggested values for Infoprint Server.

Table 15. Coupling-facility structure parameters for Infoprint Server

Parameter	Meaning	Suggested value
NAME	Structure name	Any name that follows the system logger guidelines
LOGSNUM	Number of log streams that can be allocated in the coupling facility list structure	2 or more
AVGBUFSIZE	Average size, in bytes, of log blocks written to all the log streams using this coupling facility structure	AVGBUFSIZE(512)
MAXBUFSIZE	Size, in bytes, of the largest log block that can be written to log streams allocated to the coupling facility	MAXBUFSIZE(8192)

Table 16 lists the coupling-facility log-stream parameters that have special considerations for the Infoprint Server. For parameters that are not shown in Table 16, you can specify any values that are suitable for your installation.

Table 16. Coupling-facility log-stream parameters for Infoprint Server

Parameter	Meaning	Suggested value
NAME	Name of the log stream	See "Deciding on a naming convention for log streams" on page 90.
STRUCTURE	Name of the structure	Name of a structure defined to the system logger
DASDONLY	Type of log stream	DASDONLY(NO)
RETPD	Number of days that the system logger messages before deleting them	See "Determining how long to retain messages" on page 90.
AUTODELETE	Automatic delete option	AUTODELETE(YES)
LS_SIZE	Size, in 4K blocks, of the offload data sets for the log stream	See "Determining the size of log-stream data sets and interim storage" on page 91.

Table 16. Coupling-facility log-stream parameters for Infoprint Server (continued)

Parameter	Meaning	Suggested value
STG_SIZE	Size, in 4K blocks, of the staging data sets for the log stream	See "Determining the size of log-stream data sets and interim storage" on page 91.
HIGHOFFLOAD	High offload threshold	HIGHOFFLOAD(80) You can change this value to suit your installation.
LOWOFFLOAD	Low offload threshold	LOWOFFLOAD(0) You can change this value to suit your installation.
DIAG	Allows collection of diagnostic information	DIAG(YES)
STG_DUPLEX	Duplex option	Because Infoprint Server can tolerate a loss of messages, you can specify STG_DUPLEX(NO) or STG_DUPLEX(YES).

Figure 13 shows sample JCL to define a coupling-facility structure named AOP_MSG_STR to the system logger.

```

//STEP1 EXEC PGM=IXCMIAPU
//SYSPRINT DD SYSOUT=A
//*****
//* Define a coupling-facility structure in the LOGR policy. The *
//* parameter values are for illustration only. Substitute values *
//* appropriate to your installation. *
//*****
//SYSIN DD *
        DATA TYPE(LOGR)
        DEFINE STRUCTURE
        NAME(AOP_MSG_STR)
        LOGSNUM(2)
        AVGBUFSIZE(512)
        MAXBUFSIZE(8192)

//*
```

Figure 13. Sample JCL to define a coupling-facility structure to the system logger

Figure 14 on page 99 shows sample JCL to define a coupling-facility structure named AOP_MSG_STR in the system logger CFRM policy.

```

//STEP1 EXEC PGM=IXCMIAPU
//SYSPRINT DD SYSOUT=A
//*****
//* Define a coupling-facility structure in the CFRM policy. The      *
//* parameter values are for illustration only. Substitute values   *
//* appropriate to your installation.                                *
//*****
//SYSIN DD *
        DATA TYPE(CFRM)
        STRUCTURE NAME(AOP_MSG_STR)
        SIZE(size)
        INITSIZE(size)
        PREFLIST(FACIL01,FACIL02)
/*

```

Figure 14. Sample JCL to define a coupling-facility structure in the CFRM policy

Figure 15 shows sample JCL to define a coupling-facility log stream named AOP.MSG in the LOGR policy.

```

//STEP1 EXEC PGM=IXCMIAPU
//SYSPRINT DD SYSOUT=A
//*****
//* Define a coupling-facility log stream to the system logger. The *
//* parameter values are for illustration only. Substitute values   *
//* appropriate to your installation.                                *
//*****
//SYSIN DD *
        DATA TYPE(LOGR)
        DEFINE LOGSTREAM
        NAME(AOP.MSG)
        DESCRIPTION(Infoprint_Server)
        DASDONLY(NO)
        STRUCTURE(AOP_MSG_STR)
        STG_SIZE(size)
        LS_SIZE(size)
        HIGHOFFLOAD(80)
        LOWOFFLOAD(0)
        RETPD(25)
        AUTODELETE(YES)
        DIAG(YES)
        HLQ(IXGLOGR)
/*

```

Note: The HLQ and EHLQ parameters are mutually exclusive. HLQ specifies the high-level qualifier for log-stream data set names (IXGLOGR is the default).

Figure 15. Sample JCL to define a coupling-facility log stream to the system logger

Related information: For information about:

- How to run the IXCMIAPU utility, see *z/OS MVS Setting Up a Sysplex*.
- How to set up security for a log stream, see “Setting up security for a coupling-facility log stream” on page 93.
- The DISPLAY LOGGER command, see *z/OS MVS System Commands*.

Activating a log stream in Infoprint Server

After you have defined the log stream for messages to the MVS system logger, you must specify the log stream name in the Infoprint Server system configuration definition. When you save the system configuration definition, Infoprint Server immediately starts writing new messages to the specified log stream. If the log

stream is a shared coupling-facility log stream, you must specify the name of the log stream in the system configuration definition on each system in the sysplex that shares the log stream.

Before you begin:

- Define the log stream to the MVS system logger and set up security for the log stream.
- Infoprint Server must be started. The user who started Infoprint Server must have UPDATE access to the RACF resource profile for the log stream.
- You must have UPDATE access to the AOP.CONFIGURATION resource profile in the RACF PRINTSRV class. If the AOP.CONFIGURATION profile does not exist, you must have UPDATE access to the AOP.ADMINISTRATOR profile in the PRINTSRV class.

If both the AOP.CONFIGURATION and AOP.ADMINISTRATOR profiles do not exist, you must have READ or higher access to the AOPADMIN profile in the RACF FACILITY class.

1. Use the Infoprint Server ISPF panels to edit the system configuration definition in the Printer Inventory.
 - a. On the main Infoprint Server ISPF panel, select **8 System** and press Enter.
 - b. On the Infoprint Server System Configuration panel, specify the name of the log stream in the **Log stream name** field.

Example: This example shows a portion of the ISPF panel that you use to specify the log stream name:

```

Edit                               Infoprint Server: System Configuration
Command ==>>

Messages:

Log stream name . . . . . AOP.MSG
  
```

- c. Save the panel. When Infoprint Server writes the first message to the common message log, it connects to the log stream.

2. (Optional) Verify that Infoprint Server is connected to the new log stream:
 - a. Print a document. Infoprint Server connects to the log stream when the first message is written to the common message log.
 - b. Display the connection status of the log stream. If the log stream is not connected, check the system console for error messages.

Example: This MVS command displays the connection status of log stream AOP.MSG:

```
DISPLAY LOGGER,CONN,LSN=AOP.MSG
```

```

CONNECTION INFORMATION BY LOGSTREAM FOR SYSTEM xxxx
LOGSTREAM          STRUCTURE          #CONN STATUS
-----
AOP.MSG            *DASDONLY*          000001 IN USE
  
```

Tip: IP PrintWay basic mode does not write any messages to the log stream. Therefore, if you use IP PrintWay basic mode, to connect to the log stream, you must print a document with the z/OS UNIX **lp** command, with the TCP/IP **lpr** command, from the Windows operating system, or from NetSpool. If you use one of these methods to submit a print job, Infoprint Server connects to the log stream and writes at least one message to it.

Related information: For information about:

- How to set up security for a log stream, see “Setting up security for log streams” on page 91.
- How to define a log stream to the system logger, see “Defining log streams to the MVS system logger” on page 94.
- How to edit the system configuration definition, see “Editing the system configuration definition for Printer Inventory Manager” on page 76.
- How to handle system logger errors, see “Handling system logger errors” on page 103.
- DISPLAY LOGGER command, see *z/OS MVS System Commands*.

Switching from one log stream to another

You can switch from one system logger log stream to another one. You might want to switch log streams if the current log stream fails.

Before you begin:

- Define the new log stream to the MVS system logger and set up security for the log stream.
- Infoprint Server must be started. The user who started Infoprint Server must have UPDATE access to the RACF resource profile for the new log stream.
- You must have UPDATE access to the AOP.CONFIGURATION resource profile in the RACF PRINTSRV class. If the AOP.CONFIGURATION profile does not exist, you must have UPDATE access to the AOP.ADMINISTRATOR profile in the PRINTSRV class.

If both the AOP.CONFIGURATION and AOP.ADMINISTRATOR profiles do not exist, you must have READ or higher access to the AOPADMIN profile in the RACF FACILITY class.

1. Use the Infoprint Server ISPF panels to edit the system configuration definition in the Printer Inventory.
 - a. On the Infoprint Server System Configuration panel, change the name of the log stream in the **Log stream name** panel.

Example: This example shows a portion of the panel:

```
Edit                               Infoprint Server: System Configuration
Command ==>>

Messages:

Log stream name . . . . . AOP.MSG.NEW
```

Save the panel. When Infoprint Server writes the first message to the common message log, it disconnects from the old log stream and connects to the new log stream.

2. (Optional) Verify that the Infoprint Server is connected to the new log stream:
 - a. Print a document. Infoprint Server connects to the new log stream when the first message is written to the common message log.
 - b. Display the connection status of the log stream. If the log stream is not connected, check the system console for error messages.

Example: This MVS command displays the connection status of log stream AOP.MSG.NEW:

```
DISPLAY LOGGER,CONN,LSN=AOP.MSG.NEW
```

```
CONNECTION INFORMATION BY LOGSTREAM FOR SYSTEM xxxx
LOGSTREAM          STRUCTURE          #CONN  STATUS
-----
AOP.MSG.NEW        *DASDONLY*          000001  IN USE
```

Tip: IP PrintWay basic mode does not write any messages to the common message log. If you use IP PrintWay basic mode and the DISPLAY LOGGER command shows that the log stream is not connected, try printing a document with the z/OS UNIX **lp** command, with the TCP/IP **lpr** command, from the Windows operating system, or from NetSpool.

Related information: For information about:

- How to set up security for a log stream, see “Setting up security for log streams” on page 91.
- How to define a log stream to the system logger, see “Defining log streams to the MVS system logger” on page 94.
- How to edit the Infoprint Server system configuration definition, see “Editing the system configuration definition for Printer Inventory Manager” on page 76.
- How to handle system logger errors, see “Handling system logger errors” on page 103.
- DISPLAY LOGGER command, see *z/OS MVS System Commands*.

Deactivating a log stream in Infoprint Server

You might want to deactivate a log stream if an error occurs that prevents Infoprint Server from connecting to the log stream or writing messages to the log stream. When you deactivate a log stream, Infoprint Server automatically starts writing messages to the z/OS UNIX file system instead of to the log stream. Infoprint Server writes messages to the z/OS UNIX file system only if the **Log retention period (days)** field is set to 1 day or higher (1 day is the default).

Before you begin:

- Infoprint Server must be started.
- You must have UPDATE access to the AOP.CONFIGURATION resource profile in the RACF PRINTSRV class. If the AOP.CONFIGURATION profile does not exist, you must have UPDATE access to the AOP.ADMINISTRATOR profile in the PRINTSRV class.

If both the AOP.CONFIGURATION and AOP.ADMINISTRATOR profiles do not exist, you must have READ or higher access to the AOPADMIN profile in the RACF FACILITY class.

1. Use the Infoprint Server ISPF panels to edit the system configuration definition in the Printer Inventory.
 - a. On the Infoprint Server System Configuration panel, remove the name of the log stream in the **Log stream name** field.
 - b. On the Infoprint Server System Configuration panel, specify the number of days worth of messages you want to retain in the z/OS UNIX file system in the **Log retention period (days)** field. The default value is 1 day.
 - c. Save the panel. When Infoprint Server writes the first message to the common message log, it disconnects from the log stream and starts writing messages to the z/OS UNIX file system.

Example: This example shows a portion of the panel:

```

Edit                               Infoprint Server: System Configuration
Command ==>

Messages:
  Log retention period (days) . . . . . 1
  Log stream name . . . . . _____

```

After you deactivate a log stream, if you use the **aoplogu** command or Infoprint Central to view messages that Infoprint Server wrote to the log stream, Infoprint Server tries to read messages from the log stream and from the z/OS UNIX file system so that you can view messages that were written to the log stream before you deactivated it.

Related information: For information about:

- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 80.
- The **Log retention period (days)** field, see “Editing the system configuration definition for Printer Inventory Manager” on page 76.

Handling system logger errors

If an MVS system logger error occurs that prevents Infoprint Server from connecting to the log stream, writing messages to the log stream, or reading messages from the log stream, Infoprint Server writes one of these error messages to the console:

- AOP186E *user* is not authorized to *access* log stream *log-stream*.
- AOP187E *log-stream* is not a valid log stream name.
- AOP188E Log stream *log-stream* is in use on another system.
- AOP189E System logger *macro* request for log stream *log-stream* failed with return code *return-code*, reason code *reason-code*.
- AOP190W System logger *macro* request for log stream *log-stream* was successful, but with return code *return-code*, reason code *reason-code*.
- AOP193E The system logger address space is not available.

After writing the error message or messages to the console, if the same system logger error occurs again, Infoprint Server does not write any more error messages to the console. However, Infoprint Server continues to try to connect to the log stream and write or read messages until the problem is corrected or you deactivate the log stream in Infoprint Server.

To prevent lost messages while you fix an MVS system logger problem, you can deactivate the log stream in Infoprint Server:

- If you deactivate the log stream, Infoprint Server starts writing messages to the z/OS UNIX file system instead of to the log stream. When the system logger error is fixed, activate the log stream in Infoprint Server again.
- If you do not deactivate the log stream in Infoprint Server, when the system logger problem is fixed, Infoprint Server automatically connects to the log stream and starts writing messages to it.

Related information: For information about:

- How to deactivate a log stream, see “Deactivating a log stream in Infoprint Server” on page 102.
- How to activate a log stream, see “Activating a log stream in Infoprint Server” on page 99.
- Infoprint Server messages, see *z/OS Infoprint Server Messages and Diagnosis*.

Viewing messages in the log stream

You can use the same methods to view messages in the common message log regardless of whether you use the system logger:

- You can use the Infoprint Server **aoplogu** command to view messages that were written in the time period specified on the command. If several instances of Infoprint Server in a sysplex use the same coupling-facility log stream, the messages from all systems that share the log stream are merged.

To use the **aoplogu** command, you must be a member of the RACF group for Infoprint Server administrators (AOPADMIN) and have READ access to the RACF resource profile for the log stream.

- You can use Infoprint Central to view messages for a selected printer, print job, or NetSpool LU.

Infoprint Central displays only the messages that were written during the period specified in the **log-retention** attribute (**Log retention period** field) of the system configuration definition. For example, if the value is 1 day, Infoprint Central only displays messages written in the last day.

Infoprint Central displays only the messages that were written from the system on which Infoprint Central is running. For example, Infoprint Central on system A displays only the messages that were written on system A even if Infoprint Server on system A and system B use the same coupling-facility log stream.

To view messages in Infoprint Central, you do not need to have access to the RACF resource profile for the log stream. However, you must be authorized to view information for the selected printer, print job, or NetSpool LU.

Both the **aoplogu** command and Infoprint Central can automatically determine which log streams were used in the specified time period. If more than one log stream was used in the time period, the **aoplogu** command and Infoprint Central read messages from all log streams. In addition, the **aoplogu** command reads any messages that were written in the z/OS UNIX file system during the time period.

You can run the **aoplogu** command while Infoprint Server is not running. However, in this case, the **aoplogu** command cannot determine the names of the log streams that were used. Therefore, you must specify the **-n** option on the **aoplogu** command.

Example: To view all messages in log stream AOP.MSG when Infoprint Server is not running, enter on the z/OS UNIX command line:

```
aoplogu -n AOP.MSG
```

The **aoplogu** output has this format:

```
date time priority:level user:id [job:id] [system:name] [inventory:name]
[filename:name] [dsn:name] [output_device:name] [job_selection_rule:name]
[fsa:name] [luname:name] [jes_jobname:name] [program:name] msg:message
```

Tip: The **system** keyword contains the MVS system name. You can use the **system** keyword to determine which system wrote the message.

Related information: For information about the **aoplogu** command, see *z/OS Infoprint Server Operation and Administration*.

Backing up and restoring the Printer Inventory

It is extremely important that you back up the Printer Inventory regularly. You should also back up the Printer Inventory after creating or modifying definitions (such as, printer definitions) in the Printer Inventory.

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.

You can use the Infoprint Server Printer Inventory Definition Utility **pidu export** command to back up and restore the Printer Inventory. The next sections summarize the steps to back up (and restore) the Printer Inventory to one of these types of files:

- A file in a Hierarchical File System (HFS) or a z/OS File System (zFS)
- A data set in a generation data group (GDG)

Rule: 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.

Steps for backing up the Printer Inventory to an HFS or zFS file

Do these steps to back up the Printer Inventory to an HFS or zFS file at the same time each day. These steps show how to back up the Printer Inventory at 3 AM each day to file `/var/Printsrv.backups/inventory.backup.yymmdd`. The date of the backup is `yymmdd`.

Guideline: Do not back up the Printer Inventory to files in the `/var/Printsrv` directory because the backup files can be large.

Before you begin:

- Create a separate file system mounted at `/var/Printsrv.backups`.
- You must be a member of the RACF group for Infoprint Server administrators (AOPADMIN) or have an effective UID of 0.
- You must have READ access to the AOP.ADMINISTRATOR profile in the PRINTSRV class.

To back up the Printer Inventory to an HFS or zFS file:

1. Create a file to contain a short shell script, for example `/u/userid/do.backup`. Your login user ID is `userid`.

If you set the PATH, NLSPATH, and optional AOPCONF environment variables with the values required by Infoprint Server in `/etc/profile` or `/u/userid.profile`, put these lines in this file:

```
. /etc/profile
. /u/userid.profile
export extension=$(date "+%y%m%d")
pidu -c "export /var/Printsrv.backups/inventory.backup.$extension;"
```

Otherwise, put these lines in this file to export the environment variables. You can omit the AOPCONF environment variable if the Infoprint Server configuration file is `/etc/Printsrv/aopd.conf`.

```
export PATH=/usr/lpp/Printsrv/bin:/bin:$PATH
export NLSPATH=/usr/lpp/Printsrv/%L/%N:/usr/lpp/Printsrv/En_US/%N:$NLSPATH
export AOPCONF=/etc/Printsrv/aopd.conf
export extension=$(date +%y%m%d)
pidu -c "export /var/Printsrv.backups/inventory.backup.$extension;"
```

Tip: To automatically delete all files in the `/var/Printsrv.backups` directory that were created more than five days ago, add this command to the shell script:

```
rm $(find /var/Printsrv.backups -ctime "+5")
```

-
2. Set the execute permission bit for the `/u/userid/do.backup` file. For example:

```
chmod +x /u/userid/do.backup
```

-
3. Use the `crontab` command with the `-e` option to insert this line into the `crontab` entry:

```
00 03 * * * /u/userid/do.backup
```

When the `cron` daemon runs the commands in your `crontab` entry, it does not first run your `.profile` to establish environment variables. For more information about the `crontab` command and the `cron` daemon, see *z/OS UNIX System Services Command Reference*.

Steps for restoring the Printer Inventory from an HFS or zFS file

Before you begin:

- You must be a member of the RACF group for Infoprint Server administrators (AOPADMIN) or have an effective UID of 0.
- You must have UPDATE access to the AOP.ADMINISTRATOR profile in the PRINTSRV class.
- If dynamic configuration is enabled, you must also have UPDATE access to the AOP.CONFIGURATION profile in the PRINTSRV class if the profile is defined.

To restore the Printer Inventory from an HFS or zFS file:

1. Stop the NetSpool and IP PrintWay basic mode tasks. Stop any PSF FSAs that use the Printer Inventory.

-
2. Stop all Infoprint Server daemons. Be sure to wait for all daemons to stop.

Example: To stop all Infoprint Server daemons, enter this MVS command to run the AOPSTOP JCL procedure:

```
START AOPSTOP
```

-
3. Move the `/var/Printsrv/master.v2db`, `/var/Printsrv/jestoken.v2db`, `pwjestoken.v2db`, `/var/Printsrv/logdb/log.v2db`, `/var/Printsrv/hinv/hinv.v2db`, and `/var/Printsrv/searchdb/*` 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.
-

4. Start Infoprint Server daemons. Do step 6 immediately after you start Infoprint Server daemons.

Example: To start Infoprint Server daemons, enter this MVS command to run the AOPSTART JCL procedure:

```
START AOPSTART
```

5. Run the **pidu** command to restore the Printer Inventory.

Example: This example shows how to restore the Printer Inventory from file `/var/Printsrv.backups/inventory.backup.yymmdd`. This command uses the UNIX redirection symbols (`>` for **stdout** and `2>` for **stderr**) to redirect command output to file **pidu.output**, and command errors to file **pidu.errors**:

```
pidu /var/Printsrv.backups/inventory.backup.yymmdd >pidu.output 2>pidu.errors
```

6. If you use Infoprint Central, update the Infoprint Central search database to reflect the changed contents of the Printer Inventory.

Example: Run the **sdbu** command with the **-z** option to update this database:

```
sdbu -z
```

7. Restart the NetSpool and IP PrintWay basic mode tasks.
-

8. Restart any PSF FSAs that use the Printer Inventory.
-

Steps for backing up the Printer Inventory to a generation data group (GDG)

Before you begin:

- You must be a member of the RACF group for Infoprint Server administrators (AOPADMIN) or have an effective UID of 0.
- You must have READ access to the AOP.ADMINISTRATOR profile in the PRINTSRV class.

To back up the Printer Inventory to a data set in a GDG:

1. Create a GDG. This sample JCL creates a GDG for which five generation data sets are maintained:

```
//AOPGDG JOB MSGLEVEL=(1,1)
//STEP1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DEFINE GDG (NAME(hlq.INV.BACKUPS) NOEMPTY SCRATCH LIMIT(5))
/*
```

2. Use one of these methods to run the **pidu export** command to back up the Printer Inventory to a new data set in GDG `hlq.INV.BACKUPS`:

- Run the **pidu export** command as a batch job:

```
//AOPBINV JOB MSGLEVEL=(1,1)
//STEP1 EXEC PGM=AOPBATCH,PARM='/pidu -c "export //DD:INVENT;"'
//INVENT DD DISP=(NEW,CATLG),DSN=hlq.INV.BACKUPS(+1),
// DCB=(RECFM=VB,LRECL=32756,BLKSIZE=32760),
// UNIT=SYSDA,SPACE=(TRK,(50,10),RLSE)
```

```
//STDOUT DD SYSOUT=*
//STDERR DD SYSOUT=*
//STDENV DD *
NLSPATH=/usr/lpp/Printsrv/En_US/%N
AOPCONF=/etc/Printsrv/aopd.conf
/*
```

If Infoprint Server files are located in the default locations, you can omit the STDENV data definition (DD).

- Run the **pidu export** command as a **cron** job:

- a. Create a file to contain a short shell script, for example */u/userid/do.backup*. Your login user ID is *userid*.

If you set the PATH, NLSPATH, and optional AOPCONF environment variables with the values required by Infoprint Server in */etc/profile* or */u/userid.profile*, put these lines in this file:

```
. /etc/profile
. /u/userid.profile
pidu -c "export /\`h1q.INV.BACKUPS\(\+1\)\`";"
```

Otherwise, put these lines in this file to export the variables. You can omit the AOPCONF variable if the Infoprint Server configuration file is */etc/Printsrv/aopd.conf*.

```
export PATH=/usr/lpp/Printsrv/bin:/bin:$PATH
export NLSPATH=/usr/lpp/Printsrv/%L/%N:/usr/lpp/Printsrv/En_US/%N:$NLSPATH
export AOPCONF=/etc/Printsrv/aopd.conf
pidu -c "export /\`h1q.INV.BACKUPS\(\+1\)\`";"
```

- b. Set the execute permission bit for the */u/userid/do.backup* file. For example:

```
chmod +x /u/userid/do.backup
```

- c. Use the **crontab** command with the **-e** option to insert this line into the **crontab** entry to back up the Printer Inventory at 3 AM each day.

```
00 03 * * * /u/userid/do.backup
```

When the **cron** daemon runs the commands in your **crontab** entry, it does not first run your **.profile** to establish environment variables. For more information about the **crontab** command and the **cron** daemon, see *z/OS UNIX System Services Command Reference*.

Steps for restoring the Printer Inventory from a GDG

Before you begin:

- You must be a member of the RACF group for Infoprint Server administrators (AOPADMIN) or have an effective UID of 0.
- You must have UPDATE access to the AOP.ADMINISTRATOR profile in the PRINTSRV class.
- If dynamic configuration is enabled, you must also have UPDATE access to the AOP.CONFIGURATION profile in the PRINTSRV class if the profile is defined.

To restore the Printer Inventory from the latest data set in the GDG:

1. Stop the NetSpool and IP PrintWay basic mode tasks. Stop any PSF FSAs that use the Printer Inventory.

2. Stop all Infoprint Server daemons. Be sure to wait for all daemons to stop.

Example: To stop all Infoprint Server daemons, enter this MVS command to run the AOPSTOP JCL procedure:

```
START AOPSTOP
```

-
3. Move the `/var/Printsrv/master.v2db`, `/var/Printsrv/jestoken.v2db`, `pwjestoken.v2db`, `/var/Printsrv/logdb/log.v2db`, `/var/Printsrv/hinv/hinv.v2db`, and `/var/Printsrv/searchdb/*` 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.

-
4. Start Infoprint Server daemons. Do step 6 immediately after you start Infoprint Server daemons.

Example: To start Infoprint Server daemons, enter this MVS command to run the AOPSTART JCL procedure:

```
START AOPSTART
```

-
5. Use one of these methods to restore the Printer Inventory from the most recent generation data set of a GDG named `hlq.INV.BACKUPS`:

- Run the **pidu** command as a batch job:

```
//AOPRINV JOB MSGLEVEL=(1,1)
//STEP1 EXEC PGM=AOPBATCH,PARM='/pidu //DD:INVENT'
//INVENT DD DISP=SHR,DSN=hlq.INV.BACKUPS(0)
//STDOUT DD SYSOUT=*
//STDERR DD SYSOUT=*
//STDENV DD *
NLSPATH=/usr/lpp/Printsrv/En_US/%N
AOPCONF=/etc/Printsrv/aopd.conf
/*
```

If Infoprint Server files are located in the default locations, you can omit the STDENV data definition (DD).

- Run the **pidu** command from the **sh** command:

- a. Create a file to contain a short shell script, for example `/u/userid/do.restore`. Your login user ID is `userid`. This example shows how to restore the Printer Inventory from the most recent generation data set of a GDG named `hlq.INV.BACKUPS`. This example uses the UNIX redirection symbols (`>` for **stdout** and `2>` for **stderr**) to redirect command output to file `pidu.output`, and command errors to file `pidu.errors`:

```
pidu /\ 'hlq.INV.BACKUPS\(\0\)\' >pidu.output 2>pidu.errors
```

- b. Run the **sh** command to restore the Printer Inventory by running the **pidu** command in the `/u/userid/do.restore` file:

```
sh /u/userid/do.restore
```

Tip: Use the **sh** command to run the **pidu** command in the `/u/userid/do.restore` file instead of running the **pidu** command directly. GDG relative generations are maintained for the duration of the UNIX shell created when you log in to z/OS UNIX System Services. Using the **sh** command to create a new shell ensures that the current generation is always 0. For example, these commands create a new generation data set and then display the generation data set prior to the newly created data set:

```
pidu -c "export //\'hlg.INV.BACKUPS\(\+1\)\'";  
cat //\'hlg.INV.BACKUPS\(\0\)\'
```

To create a new generation data set and then display that data set, you would need to run this command:

```
pidu -c "export //\'hlg.INV.BACKUPS\(\+1\)\'";  
cat //\'hlg.INV.BACKUPS\(\+1\)\'
```

-
6. If you use Infoprint Central, update the Infoprint Central search database to reflect the changed contents of the Printer Inventory.

Example: Run the **sdbu** command with the **-z** option to update this database:

```
sdbu -z
```

-
7. Restart the NetSpool and IP PrintWay basic mode tasks.

-
8. Restart any PSF FSAs that use the Printer Inventory.
-

Saving messages in the common message log in the z/OS UNIX file system

This topic describes how to save messages in the common message log regularly. You might want to save messages if you specify a low retention period for messages. If you use the z/OS UNIX file system for messages, Infoprint Server automatically deletes messages after the retention time specified in the **log-retention** attribute expires. If you use the MVS system logger for messages, the system logger automatically deletes messages after the retention time for the log stream (specified in the RETPD parameter) expires.

You can use the Infoprint Server **aoplogu** command to save messages in the common message log to an HFS or zFS file or to an MVS data set. To use the **aoplogu** command, you must be a member of the AOPADMIN group. For more information, see “Setting up security for the Printer Inventory and operator commands” on page 47.

Before you begin: Create a separate file system mounted at **/var/Printsrv.backups** for the saved message log files. Do not save messages in the **/var/Printsrv** directory because the saved message log files can be large.

Steps for saving messages: These steps show how to save one day's worth of messages at 3 AM each day to file **/var/Printsrv.backups/message.logs.yymmdd**. The date of the backup is **yymmdd**.

1. Create a file to contain a short shell script, for example **/u/userid/do.savemsg**, to save one day's worth of messages. Your login user ID is **userid**.

If you set the PATH, NLSPATH, and optional AOPCONF environment variables with the values required by Infoprint Server in **/etc/profile** or **/u/userid/profile**, put these lines in this file:

```
. /etc/profile  
. /u/userid/profile  
export extension=$(date "+%y%m%d")  
aoplogu -l 1:0:0 > /var/Printsrv.backups/message.logs.$extension
```

Otherwise, put these lines in this file to export the environment variables. You can omit the AOPCONF variable if the Infoprint Server configuration file is `/etc/Printsrv/aopd.conf`.

```
export PATH=/usr/lpp/Printsrv/bin:/bin:$PATH
export NLSPATH=/usr/lpp/Printsrv/%L/%N:/usr/lpp/Printsrv/En_US/%N:$NLSPATH
export AOPCONF=/etc/Printsrv/aopd.conf
export extension=$(date "+%y%m%d")
aoplogu -l 1:0:0 > /var/Printsrv.backups/message.logs.$extension
```

Tip: To automatically delete all files in the `/var/Printsrv.backups` directory that were created more than five days ago, add this command to the shell script:

```
rm $(find /var/Printsrv.backups -ctime "+5")
```

-
2. Set the execute permission bit for the `/u/userid/do.savemsg` file. For example:

```
chmod +x /u/userid/do.savemsg
```

-
3. Use the `crontab` command with the `-e` option to insert this line into the `crontab` entry:

```
00 03 * * * /u/userid/do.savemsg
```

When the `cron` daemon runs the commands in your `crontab` entry, it does not first run your `.profile` to establish environment variables. For more information about the `crontab` command and the `cron` daemon, see *z/OS UNIX System Services Command Reference*.

Reducing the size of the common message log in the z/OS UNIX file system

This topic describes how to reduce the size of the common message log database in the z/OS UNIX file system. You might need to do this task if the database becomes larger than your file system can accommodate. If you reduce the size of the database, Infoprint Central cannot display messages that were written to the log before you decreased its size.

Steps for reducing the size of the common message log:

1. Switch to an effective UID of 0:

```
su
```

To use the z/OS UNIX `su` command, you must be permitted to the `BPX.SUPERUSER` profile in the `FACILITY` class in RACF.

-
2. Specify fewer days or 0 in the `log-retention` attribute in the Infoprint Server configuration file (`aopd.conf`).

Tip: Because the common message log and Historical Inventory can contain a large amount of data, start with a value of 1 day. Increase the value by 1 if there is enough space in the file system and if users request to see more messages or historical information. (For other tips related to this attribute, see “Creating the Infoprint Server configuration file (`aopd.conf`)” on page 34.)

-
3. Stop all Infoprint Server daemons. For example, enter this MVS command to run the AOPSTOP JCL procedure:


```
START AOPSTOP
```

4. Change to the Infoprint Server base directory. The default base directory is `/var/Printsrv`. If you specify a different base directory in the **base-directory** attribute in the **aopd.conf** file, change to that directory.

```
cd /var/Printsrv
```

5. Delete the **log.v2db** file, which contains the common message log database:

```
rm logdb/log.v2db
```

6. If you switched to an effective UID of 0 in step 1, switch back to your own UID:

```
exit
```

7. Restart Infoprint Server daemons. For example, enter this MVS command to run the AOPSTART JCL procedure:

```
START AOPSTART
```

Results:

- Infoprint Central can no longer display messages that were previously written in the common message log. However, you can still use the **aoplogu** command to retrieve these messages.
- Infoprint Server creates a new **log.v2db** file with no data. If you specified a value greater than 0 in the **log-retention** attribute, as Infoprint Server processes print jobs, the size of this file grows.
- As the retention time expires for messages, Infoprint Server deletes message files (`/var/Printsrv/logs/timestamp`).

Setting Workload Manager goals

This section lists 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.

- Put the NetSpool and IP PrintWay basic mode started tasks in a high-priority started task (subsystem type STC) service class.
- Assign the Infoprint Server daemons in a Classification Group, using Transaction Class Groups, to a Transaction Name Group (TNG). If you want to list each Infoprint Server daemon individually, see “Customizing startup and shutdown commands and procedures” on page 56 for a list of daemon names.

The transform daemons, by default, inherit the name of the Infoprint Server Transform Manager daemon, AOPXFD. However, you can assign a unique name to each transform daemon in the `_BPX_JOBNAME` environment variable in the transform configuration file, **aopxfd.conf**. If the transform daemons have unique names, you can put transform daemons in separate TNGs and manage them separately. (For information about how to specify the `_BPX_JOBNAME` environment variable, see “Format of an entry in the transform configuration file” on page 160.)

Example: This example uses the group name AOP and a wildcard for all daemons that start with AOP. Also, it assumes that the transform daemons have unique names in the `_BPX_JOBNAME` environment variable. For example, it assumes that this environment variable is set for the PostScript/PDF to AFP transform daemon: `_BPX_JOBNAME=PS2AFPD`.

```
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
AFP*	From AFP transforms
PS2AFPD	PostScript/PDF to AFP transform
PCL2AFPD	PCL to AFP 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.
- 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.

Example: This example shows a definition for service class OMVSDMN, which is used for Infoprint Server daemons:

#	Duration	Imp	Goal description
1	200	3	80% complete within 00:00:00.500
2	600	4	60% complete within 00:00:01.000
3		5	Execution velocity of 40

Tips:

1. # = Period number
 2. Imp = Importance
- Assign the TNG name that you defined for Infoprint Server to the service class in Classification Rules, under the OMVS subsystem type.

Example: This example shows the TNG named AOP, assigned to service class OMVSDMN, using the report class RAOP. The report class is optional.

```
Subsystem Type . . : OMVS
Description . . . z/OS UNIX System Services
```

#	Qualifier type	Qualifier name	Starting position	Service Class	Report Class
1	TNG	AOP	___	OMVSDMN	RAOP

For more information about Workload Manager, see *z/OS MVS Planning: Workload Management*.

Chapter 4. Customizing Print Interface

This information describes how to customize the Print Interface component of Infoprint Server. You must customize Print Interface to submit print requests from these sources:

- z/OS UNIX shell commands (**lp**, **lpstat**, or **cancel**) or the 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 (API)
- SAP R/3 Application Server for z/OS
- Batch jobs that specify the Print Interface subsystem in the SUBSYS parameter on the DD JCL statement

Print Interface processes print requests from local users and remote clients and allocates output data sets on the JES spool. For an overview of Print Interface and how it fits into your system, see “Print Interface” on page 6.

Table 17 summarizes the tasks to customize Print Interface. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies.

Table 17. Customization tasks for Print Interface

Task	Condition	See page
Identifying software requirements for Print Interface	Required	116
Common customization for Print Interface	Required	116
Customizing the z/OS UNIX printing commands	Optional: To use lp command	130
Customizing the AOPPRINT JCL procedure	Optional: To use AOPPRINT procedure	130
Customizing the Print Interface LPD	Optional: To print from an LPR, the Infoprint Port Monitor for Windows, or a remote SAP R/3 application server; also required to use the IP PrintWay basic mode resubmit for filtering function	132
Customizing the z/OS SMB server to use Print Interface	Optional: To print from a Windows client using SMB protocol	138
Customizing the Print Interface IPP Server	Optional: To print from an IPP client	140
Customizing SAP R/3	Optional: To print from SAP R/3	144
Customizing the Print Interface subsystem	Optional: To use the Print Interface subsystem	153

This table describes other components of Infoprint Server that you might need to customize:

Task	Condition	See page
Customizing the Printer Inventory Manager	Required	27
Customizing Transform Interface	Optional: To transform data	157
Customizing IP PrintWay	Optional: To print on printers in a TCP/IP or SNA network	199
Customizing the Windows client	Optional: To submit print requests from the Infoprint Port Monitor for Windows	369

Related tasks: After you customize Print Interface, do these tasks. For information about these tasks, see *z/OS Infoprint Server Operation and Administration*:

- Create a printer definition in the Printer Inventory for each printer and specify the fields in the printer definition that Print Interface requires.
- To use the LPD, IPP Server, or subsystem, start the daemons by running either the **aopstart** command or the AOPSTART procedure.

Identifying software requirements for Print Interface

This additional IBM software beyond the z/OS base elements is required:

- To submit print jobs using the SMB printing protocol, the z/OS SMB print server, which is part of the z/OS Distributed File Service element, is required.
- If you start the Print Interface IPP Server, one of these Java™ products is required:
 - IBM 31-bit SDK for z/OS, Java Technology Edition, V7 (5655-W43)
 - IBM 31-bit SDK for z/OS, Java Technology Edition, V6.0.1 (5655-R31)
 - IBM 31-bit SDK for z/OS, Java Technology Edition, V6 (5655-R31)

For information about Java products for z/OS, see <http://www-03.ibm.com/systems/z/os/zos/tools/java/>

The IPP Server cannot use the 64-bit SDK. However, you can install it on the z/OS system.

- To submit print jobs from SAP R/3, one of these SAP R/3 application servers, which you can order 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 AFP printers on the z/OS system, Infoprint Transforms to AFP for z/OS (5655-N60) is required.

Common customization for Print Interface

This section describes these customization tasks, which you must do before you submit print requests to Print Interface. Required tasks are required by all installations that want to use Print Interface. Optional tasks are required only if the listed condition applies to your installation.

Table 18. Common customization tasks for Print Interface

Task	Condition	See page
Customizing z/OS Communications Server	Required	117
Updating the BPXPRMxx member of SYS1.PARMLIB	Required	119
Working with JES spool limits	Required	119
Editing the aopd.conf file for Print Interface	Optional: To change default configuration	120
Writing a filter program	Optional: To inspect or modify input data streams	121
Customizing JES to specify no blank truncation	Optional: To print XML data using PSF	127
Setting environment variables for Print Interface	Optional: To print line data that has unprintable characters	127
Editing the system configuration definition for Print Interface	Optional: To specify dynamic attributes	128

Customizing z/OS Communications Server

You must customize and start the IP component (TCP/IP) of z/OS Communications Server if you plan to (1) submit print jobs to Print Interface from remote systems, (2) use IP PrintWay to transmit print jobs to remote printers or email destinations, or (3) use the SNMP subagent. This section describes considerations related to customizing TCP/IP.

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**).

For complete information about how to customize TCP/IP, see:

- *z/OS Communications Server: IP Configuration Guide*
- *z/OS Communications Server: IP Configuration Reference*

Editing the hlq.PROFILE.TCPIP data set

You can customize the *hlq.PROFILE.TCPIP* 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.

Tips:

- If you change any of the values in the *hlq.PROFILE.TCPIP* data set, restart the Communications Server so that your changes take effect.
- For considerations that apply if you also run IP PrintWay, see “Customizing z/OS Communications Server” on page 232.

TCPCONFIG statement: The TCPCONFIG statement in the *hlq.PROFILE.TCPIP* data set updates the TCP layer of TCP/IP. This is a sample TCPCONFIG statement that you might want to include in the *hlq.PROFILE.TCPIP* data set, followed by an

explanation of the parameters and considerations for both Print Interface and IP PrintWay:TCPCONFIG INTERVAL 10 SENDGARBAGE TRUE TCPCVBFRSIZE 32768 TCPSENDBFRSIZE 32768

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.

TCPCVBFRSIZE *buffer size*

The TCP receive buffer size. IP PrintWay requires a buffer size of at least 32768.

TCPSENDBFRSIZE *buffer size*

The TCP send buffer size. IP PrintWay requires a buffer size of at least 32768.

Tip: You can also specify some of these parameters on other Communications Server statements. The settings from the last statement processed are used.

PORT and PORTRANGE statements: The PORT and PORTRANGE statements tell TCP/IP to reserve ports for specific processes. Make sure that the port numbers Infoprint Server uses are not reserved for other processes. If you use the Infoprint Server default ports, do not reserve these ports:

- Print Interface LPD: port 515
- Print Interface IPP Server: port 631
- z/OS SNMP agent: ports 161 and 162

Example: This example comments out a line that reserves port 515 for the TCP/IP LPD. A semicolon in the first column indicates the line is a comment.

```
PORT
:
; 515 TCP LPSERVE ; LP Server
```

Tip: Both Print Interface and IP PrintWay must be able to obtain an available port in the 0 through 1023 range. However, you do not need to explicitly reserve any ports in this range for use by Print Interface or IP PrintWay even if the **Restrict ports** field is selected in a printer definition.

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
```

For information about whether you want to run the Communications Server LPD in addition to the Print Interface LPD, see “Customizing z/OS Communications Server (TCP/IP)” on page 133.

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:

Parameter	Meaning
MAXPROCUSER	The maximum number of processes that a single z/OS UNIX user ID can have concurrently active. Guideline: Add at least 70 for Infoprint Server processes.
MAXPROCSYS	The maximum number of processes that the z/OS system allows. Guideline: 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).

Example: Specify these statements in the BPXPRMxx member:

```
MAXPROCUSER(70) /*70(Infoprint Server) */
MAXPROCSYS(450) /*200(PrintInterface/NetSpool)+50(PrintWay extended)+200(initial)*/
```

Tip: You can use the SETOMVS or SET OMVS command to dynamically increase the MAXPROCUSER and MAXPROCSYS values. However, to make a permanent change, edit the BPXPRMxx member that is used for IPLs. For more information about the BPXPRMxx member, see *z/OS MVS Initialization and Tuning Reference*.

Working with JES spool limits

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 (**aopipppdxp**)
- 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 JES2 and JES3 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, JES2 and JES3 display a warning message on the system console and let 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.

Tip: 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.

Related task: Your administrator can limit the amount of data that Infoprint Server writes to the JES spool for any one print job. For information, see the description of the **Maximum document size** field or **maximum-document-size** attribute in *z/OS Infoprint Server Operation and Administration*.

Editing the `aopd.conf` file for Print Interface

The `aopd.conf` file contains attributes that Infoprint Server uses when it starts. The default location of this file is `/etc/Printsrv/aopd.conf`.

If dynamic configuration is not enabled, you can specify the following attribute in the `aopd.conf` file. If you change this attribute, you must stop and restart all Infoprint Server daemons for the change to take effect.

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.

The prefix must be two alphanumeric or national (@, \$, and #) 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 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.
- 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.

Tips:

- z/OS also assigns a unique z/OS job ID to each output data set. The z/OS job ID might be different from the job ID of the output data set on the JES spool. JES operator commands use this z/OS job ID.
- The job ID that Print Interface returns to the job submitter does not contain this prefix.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the `aopd.conf` file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: `job-prefix = PS`

Example: job-prefix = 'P#'

Related information: For information about:

- All attributes in the **aopd.conf** file, see Appendix B, “Infoprint Server files,” on page 389.
- How to edit the **aopd.conf** file, see “Steps for editing the Infoprint Server configuration file (aopd.conf)” on page 42.
- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 68.

Writing a filter program

Note

This section contains programming-interface information.

Infoprint Server lets your installation write filter programs to monitor and modify data. Infoprint Server calls the filter program before Print Interface or NetSpool writes the data to an output data set on the JES spool. For batch jobs submitted with JCL to IP PrintWay extended mode, Infoprint Server calls the filter program before IP PrintWay extended mode sends the data to the printer.

A filter can inspect the input data and do these types of functions:

- Convert data in a document from one data format to another
- Add a header or trailer page to each document
- Add printer instructions to the beginning of each document
- Add, delete, or modify data in a document
- Terminate processing of a document
- Specify job attributes, such as the number of copies, email address, or form definition. Only a DLL filter can do this function. A UNIX filter cannot do this function.

You can write these types of filters:

- **DLL filter:** A filter that resides in a dynamic link library (DLL) that **cfilter.h** defines.
- **UNIX filter:** A filter that reads from **stdin** and writes to **stdout**.

For some applications, you might need to write a DLL filter instead of a UNIX filter because a DLL filter can inspect more information about the input data stream and can also do more functions. For example, only in a DLL filter can you change the attribute that indicates the format of the data stream. If you want to change the data format, therefore, you must write a DLL filter instead of a UNIX filter. For more information about differences between DLL and UNIX filters, see “Writing a DLL filter” on page 123 and “Writing a UNIX filter” on page 124.

Options:

- Instead of writing a filter to add header pages and trailer pages to output data sets that IP PrintWay transmits to the target printer, you can write an IP PrintWay Begin Data Set and an IP PrintWay End Data Set exit routine. For more information, see “Begin Data Set exit” on page 258 and “End Data Set exit” on page 276.

- Instead of writing a filter to specify printer instructions for IP PrintWay to add to the beginning and end of each data set, the administrator can specify these instructions in a printer definition. For more information, see *z/OS Infoprint Server Operation and Administration*.
- Instead of writing a filter to add header and trailer pages to output data sets that PSF processes, you can write a PSF separator page exit. For information, see *PSF for z/OS: Customization*.

Filter options

You can pass filter options and operands to your DLL or UNIX filter. The administrator can specify filter options and operands in the printer definition, and the user can specify filter options in the **filter-options** job attribute.

The administrator can use the special % operator to pass the value of a job attribute to the filter. For example, if a user specifies a job attribute on the **lp** command, the value of the attribute can be passed to the filter. Use this format:

%attribute_name

% Causes the entire *%attribute-name* substring to be replaced by the value of the attribute.

attribute-name

Is the name of any job attribute, as defined in *z/OS Infoprint Server User's Guide*.

If an attribute value is empty or contains special character or spaces, the value is enclosed in quotation marks and passed to the filter as one argument.

Examples: Using the % operator:

In this example, **%address-text** is specified as an operand for UNIX filter **test_filter** in the printer definition:

```
spawn /usr/local/bin/test_filter %address-text
```

If the **lp** command specifies this **address-text** job attribute:

```
lp -o "address-text={a1 a2}"
```

Filter **test_filter** is passed these arguments:

```
/usr/local/bin/test_filter "{ a1 a2 }"
```

The filter program receives `/usr/local/bin/test_filter` as the first argument and `{ a1 a2 }` as the second argument (without quotation marks).

Using the % operator with a filter written in the REXX language:

Sometimes the values for job attributes are null. If you use REXX to write your filter, use a **parse** statement like this one to parse arguments, so that null arguments are properly passed to the program:

```
parse arg arg1 ' ' arg2 ' ' arg3
```

Sometimes the values for job attributes contain embedded spaces. Because the **parse** statement shown above does not correctly receive an argument that contains embedded spaces, use an alternative delimiter such as `/` to delimit filter options that use the % operator. Make sure that the delimiter you select does not have a

special meaning to the shell and does not occur in the data. For example, if the administrator uses / to delimit filter options in the printer definition as shown:

```
spawn /usr/local/bin/test_filter /%name-text/%document-format/
```

And, if the **lp** command specifies the **name-text** job attribute:

```
lp -o "name-text='mike smith'" text.file
```

The command, after substitution of the attribute values, is:

```
/usr/local/bin/test_filter /"mike smith"/text/
```

You can use this REXX **parse** statement to parse the arguments:

```
parse arg '/' name_text '/' doc_format '/'
```

For more information about how to specify the filter name in a printer definition, see “Invoking a DLL or UNIX filter” on page 125 and see *z/OS Infoprint Server Operation and Administration*.

Writing a DLL filter

A DLL filter can convert data to a different format. For example, a DLL filter can convert line data to text format or to PDF format. If a DLL filter changes the data format, the filter must indicate the new data format in the **document-format** job attribute. DLL filters must convert the data to the printer code page.

To assist you in writing a new DLL filter, IBM provides the source code for the **aopfiltr.so** DLL filter in these files:

- **cfilter.h**, a header file that contains declarations and interface descriptions. Do not change the contents of this file.
- **cfilter.c**, the source code for the **aopfiltr.so** filter.

Both files are in directory **/usr/lpp/Printsrv/samples**. For more information about **aopfiltr.so**, see “Filters provided with Infoprint Server” on page 126.

To customize the **aopfiltr.so** filter:

- Modify file **cfilter.c** as necessary, using the comments in the prologs of the file.
- Link the filter as a dynamic link library (DLL), as described in the comments in file **cfilter.c**.

Rule: Use the z/OS UNIX **extattr** command to set the **a** extended attribute for the filter. To use the **extattr** command, you must have at least read access to the BPX.FILEATTR.APF resource in the FACILITY class profile. For example:

```
extattr +a filtername.dll
```

For a more secure environment, do not link the filter with the AC=1 attribute. When you omit AC=1, the exit runs APF-authorized only when an APF-authorized program, such as Infoprint Server, calls it.

Input to a DLL filter: A DLL filter has access to this information:

- Filter options and operands.
- Environment variables.
- Input data in a document. Text and line-data documents have already been converted to the printer code page.
- Name of the printer code page used to convert the data.
- Format of the data, as determined by Infoprint Server or as specified by the job submitter.

- The LPD control file in a format defined by RFC 1179.
- The type of carriage controls in the input file.
- The name of the input file.
- The name of the printer definition for the print request.
- Job attributes. Job attributes can be specified either by the job submitter or in the printer definition. For a list of Infoprint Server job attributes, see *z/OS Infoprint Server User's Guide*.

Output from a DLL filter: A DLL filter can return this output to Infoprint Server:

- Modified data.
- Format of the data.
- The type of carriage controls.
- Job attributes. Infoprint Server uses job attributes when it allocates data sets on the JES spool. For example, if the filter specifies the **copies** or **form-definition** job attribute, Infoprint Server allocates the data set on the JES spool as if the COPIES or FORMDEF JCL parameter were specified. For a list of Infoprint Server job attributes and corresponding JCL parameters, see *z/OS Infoprint Server User's Guide*.
- Printer code page. Text and line-data documents are converted to this code page before the data is passed as input to the filter.
- Return code. If the return code indicates an error, Infoprint Server ends processing of the document.

Writing a UNIX filter

A UNIX filter can be a shell executable, such as a C program, REXX exec, or a shell script. Infoprint Server spawns a UNIX filter as a separate process. (**cat** found on all UNIX systems is an example of a UNIX filter.)

You can write a UNIX filter to modify data before Infoprint Server formats it. However, a UNIX filter must not convert data to a different format. For example, a UNIX filter must not convert line data to text format or to PDF format. Also, a UNIX filter must not convert data to a different code page. After Infoprint Server calls a UNIX filter, Infoprint Server processes the data as usual. It assumes that the document format and document code page are unchanged.

The shell script in Figure 16 shows an example of a simple UNIX filter:

```
#!/bin/sh
# DESCRIPTION:
#   This filter processes EBCDIC (not ASCII).
#   Terminate each line with "carriage return + line feed".
#=====
awk '{ printf "%s\r\n", $0 }';
exit
```

Figure 16. Sample UNIX filter

Rule: If you use the AOPPRINT JCL procedure or the AOPBATCH program to print, set the BPX_SHAREAS=YES environment variable in the AOPPRINT JCL procedure and the AOPBATCH program.

Input to a UNIX filter: A UNIX filter can access this information:

- Filter options and operands.
- Environment variables.

- The input data in **stdin**. All data is in the document code page.

Output from a UNIX filter: A UNIX filter can return this output to Infoprint Server:

- Modified data in **stdout**. The modified data must be in the document code page. A UNIX filter must not convert data to a different format or change any document attributes.
- A return code. If the return code indicates an error, Infoprint Server ends processing of the document and does not allocate the data set on the JES spool.

Tip: If you run IP PrintWay extended mode and your UNIX filter transforms line data to PCL, PDF, or PostScript format, consider specifying the `AOP_SUPPRESS_POST_UNIX_FILTER_FORMATTING` environment variable or, if dynamic configuration is enabled, the **suppress-post-unix-filter-formatting** attribute.

Invoking a DLL or UNIX filter

To request that Infoprint Server invoke a filter, the administrator must specify the filter name and filter options in the printer definition for the target printer. The full path name is required unless the filter is in a directory named in the `LIBPATH` environment variable (for DLL filters) or in the `PATH` environment variable (for UNIX filters). For UNIX filters, the administrator must type **spawn** before the filter name.

The administrator specifies the filter in the **Filter** field of the Processing section of a printer definition, or in a Processing component. The administrator can specify different filters for different input data formats. For more detailed information, see *z/OS Infoprint Server Operation and Administration*.

Programming considerations

Consider these points when coding a filter program:

- You should be an experienced programmer who is aware of overall system implications.
- Do not send a long-term explicit or implied `WAIT` in the filter program because it can reduce the throughput of the Print Interface LPD or IPP Server.
- The filter program runs in 31-bit addressing mode.
- Code the filter program to be reentrant.
- All filter programs run in problem state.
- Programming exceptions cause Infoprint Server to abend. An `ESTAE` is in effect while a filter program is running.
- A DLL filter, in the initialization function, can obtain a work area and pass a pointer to that work area to other functions in the filter program. Using the work area, one function can pass data to another function.
- Future Infoprint Server program maintenance might require that you recompile your filter programs.

When a UNIX filter reads from **stdin**, the read function can obtain fewer than the requested number of bytes. You must code your UNIX filter accordingly.

In most programming and scripting languages, the read function obtains the requested number of bytes if the program reads from a file on disk. The read function can obtain fewer than the requested number of bytes if the program reads from a pipe or socket. When Infoprint Server runs a UNIX filter, the UNIX filter's

stdin is a pipe. The read function returns zero bytes when end-of-file (end-of-input) is reached on a file, pipe, or socket.

If a UNIX filter needs to obtain a specific number of bytes from **stdin** before it can process the data, it must continue to invoke the read function until it has obtained that number of bytes or end-of-file is reached. Each time the program invokes the read function it must adjust the buffer pointer and bytes requested to account for data that has already been obtained. It is often convenient to create a function in the UNIX filter for this purpose. Here is some representative pseudo-code:

```
function: read_x_bytes
  input arguments: buffer_ptr
                  bytes_requested
  returns:         success or failure
  {
    curr_ptr = buffer_ptr
    total_bytes_obtained = 0

    do {

      bytes_obtained = read( curr_ptr,
                            bytes_requested - total_bytes_obtained )
      curr_ptr = curr_ptr + bytes_obtained
      total_bytes_obtained = total_bytes_obtained + bytes_obtained

    } while ((bytes_obtained > 0) AND
             (total_bytes_obtained < bytes_requested))

    if (total_bytes_obtained EQUALS bytes_requested)
      return success
    else
      return failure
  }
```

Testing filters

To test your filter, specify the name of the filter in a printer definition and use the **lp** command to print a document to that printer definition. As an alternative, you can use the **filter** command to test your filter. For a description of the **filter** command, see `/usr/lpp/Printsrv/samples/cfilter.h`.

Modifying filters

After modifying a DLL filter, you must restart the Infoprint Server daemons to load the new version of the filter if Infoprint Server has already used the old version of the filter with the same name. You do not need to restart Infoprint Server when you modify a UNIX filter or if you have tested the filter only with the **lp** or **filter** command.

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 Ricoh InfoPrint Manager for AIX or Windows.
- **aoprxf.so**: Sends data to be transformed to Ricoh InfoPrint Transform Manager for Linux.
- **lpd_compat.so**: Provides an LPD filter with similar function to the z/OS Communications Server LPD.

Customizing JES to specify no blank truncation

JES2 and JES3, by default, truncate 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 about these JES parameters and how to restart JES after you set them, 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.

Tip: 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.

Setting environment variables for Print Interface

If dynamic configuration is not enabled, Print Interface uses the following environment variable, which you can set in the **aopstart** EXEC:

AOP_ALLOW_ALL_CHARACTERS_IN_LINE_DATA

This environment variable determines how Infoprint Server detects the data format of documents that contain line data. Valid line data contains carriage control characters, table reference characters, binary zeroes, and printable characters. However some applications might create line data that contains unprintable characters.

If this environment variable is set to YES, if line data contains unprintable characters, Infoprint Server detects the data format as “line data”.

If this environment variable is not set, if line data contains unprintable characters, Infoprint Server detects the data format as “other”. Line data that Infoprint Server detects as “other” might not print correctly because Infoprint Server does not convert it to the printer’s code page.

Tips:

- Job submitters can specify the data format in the **document-format** job attribute. For example, job submitters can specify **document-format=line** in the PRTATTRS parameter on the OUTPUT JCL statement. If the **document-format** job attribute is specified, Infoprint Server ignores this environment variable because it does not detect the data format.

- Set this environment variable to YES if Infoprint Server does not detect the line data format and job submitters cannot specify the **document-format=line** job attribute.
- This environment variable does not affect how NetSpool checks SCS and 3270 data streams for validity. The input SCS and 3270 data streams must contain only valid characters.

Rules:

- If dynamic configuration is enabled, specify the equivalent **allow-all-characters-in-line-data** attribute in the system configuration definition.
- If dynamic configuration is not enabled, specify this environment variable in the **aopstart EXEC**.

Default: The environment variable is not set.

Example: AOP_ALLOW_ALL_CHARACTERS_IN_LINE_DATA=YES

Print Interface also uses the LANG, LIBPATH, NLSPATH, and PATH environment variables.

Related information: For information about:

- How to edit the **aopstart EXEC**, see “Editing the aopstart EXEC” on page 57
- The LANG, LIBPATH, NLSPATH, and PATH variables, see “Setting environment variables for the Printer Inventory Manager” on page 43

Editing the system configuration definition for Print Interface

The system configuration definition in the Printer Inventory contains attributes that let you customize Infoprint Server while Infoprint Server is running. If dynamic configuration is enabled, you can specify these attributes in the system configuration definition to customize Print Interface:

allow-all-characters-in-line-data = yes | no

This attribute determines how Infoprint Server detects the data format of documents that contain line data. Valid line data contains carriage control characters, table reference characters, binary zeroes, and printable characters. However some applications might create line data that contains unprintable characters.

yes Infoprint Server detects the data format as “line data” even if the input data contains unprintable characters.

no Infoprint Server detects the data format as “other” if the input data contains unprintable characters. Line data that Infoprint Server detects as “other” might not print correctly because Infoprint Server does not convert it to the printer's code page.

Tips:

- Job submitters can specify the data format in the **document-format** job attribute. For example, job submitters can specify **document-format=line** in the PRTATTRS parameter on the OUTPUT JCL statement. If the **document-format** job attribute is specified, Infoprint Server ignores this attribute because it does not detect the data format.
- Specify **allow-all-characters-in-line-data=yes** if Infoprint Server does not detect the line data format and job submitters cannot specify the **document-format=line** job attribute.

- This attribute does not affect how NetSpool checks SCS and 3270 data streams for validity. The input SCS and 3270 data streams must contain only valid characters.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify the equivalent AOP_ALLOW_ALL_CHARACTERS_IN_LINE_DATA environment variable in the **aopstart** EXEC.
- This attribute does not apply to IP PrintWay basic mode.

Default: allow-all-characters-in-line-data = no

Example: allow-all-characters-in-line-data = yes

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.

The prefix must be two alphanumeric or national (@, \$, and #) 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 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.
- 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.

Tips:

- z/OS also assigns a unique z/OS job ID to each output data set. The z/OS job ID might be different from the job ID of the output data set on the JES spool. JES operator commands use this z/OS job ID.
- The job ID that Print Interface returns to the job submitter does not contain this prefix.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: job-prefix = PS

Example: job-prefix = 'P#'

Example: This screen shows a portion of the ISPF panel that you use to edit the system configuration definition for Print Interface:

General:
 _ Allow all characters in line data
 Job ID prefix . . PS

Related information: For information about:

- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 68.
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 80.

Customizing the z/OS UNIX printing commands

Print Interface provides enhanced z/OS UNIX shell commands for printing: **lp**, **lpstat**, and **cancel**. For information about how to use these commands, see *z/OS Infoprint Server User's Guide*.

These environment variables 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 **lp**, **lpstat**, and **cancel** commands.

AOOPTIONS

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 AOOPTIONS 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. For the format of the job attributes and values, see *z/OS Infoprint Server User's Guide*.

AOPPATH

The directory that contains attribute files used by the **lp** command. This variable is *optional*. If you do not create attribute files for the **lp** command, you do not need to set this variable. For information about attribute files, see *z/OS Infoprint Server User's Guide*.

PRINTER or LPDEST

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 Server ISPF panels to define a default printer. See *z/OS Infoprint Server Operation and Administration*. The printer named in either LPDEST or PRINTER overrides the printer named on the ISPF configuration panel.

To use the z/OS UNIX printing commands, you do not need to start any daemons in addition to the Printer Inventory Manager daemon.

Customizing the AOPPRINT JCL procedure

Infoprint Server provides the AOPPRINT JCL procedure, which local users can use to submit the **lp** command from a batch job.

AOPPRINT is shipped in SYS1.IBM.PROCLIB. However, during install AOPPRINT might have been copied to another data set in the PROCLIB concatenation.

For information about how to use the AOPPRINT JCL procedure, see *z/OS Infoprint Server User's Guide*.

Figure 17 shows the AOPPRINT procedure.

```
/*-----  
/* AOPPRINT - OS/390 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
```

Figure 17. AOPPRINT JCL procedure in SYS1.IBM.PROCLIB

You can customize the AOPPRINT JCL procedure for your installation in these 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 don't change the name, the default is **p1**. The administrator should create a printer definition with the default name.
- Specify environment variables in an STDENV data definition (DD), as shown in the procedure. If your installation installed Infoprint Server in the default directory, **/usr/lpp/Printsrv/**, you do not need to specify the LIBPATH, NLSPATH, and PATH environment variables. Specify each variable on a separate line, in the format *variable=value*:

```
LIBPATH=/usr/mylib  
PATH=/usr/mylib
```

Tip: The **lp** command returns MVS return codes (0, 4, 8) instead of UNIX exit values (0, 1) when you set the AOP_MVS_RETURN_CODES=YES environment variable.

- If any printer definitions specify an installation-written UNIX filter, specify the **_BPX_SHAREAS=YES** environment variable. If you do not set **_BPX_SHAREAS=YES**, you can receive this error message:
lp: AOP003E cat: EDC5157I An internal error has occurred.
- If you did not add the Language Environment runtime libraries (*hlq.SCEERUN* and *hlq.SCEERUN2*) and the C++ runtime library (*hlq.SCLBDLL*) to the system LNKLST, concatenate them to a STEPLIB DD statement.

To use the AOPPRINT procedure, you do not need to start any daemons in addition to the Printer Inventory Manager daemon.

Related information: For information about how to access the Language Environment runtime libraries and improve the performance of runtime libraries, see *z/OS UNIX System Services Planning*.

Customizing the Print Interface LPD

This section describes how to customize the Print Interface Line Printer Daemon (LPD). You need to customize the Print Interface LPD if you want to:

- Submit print requests to Infoprint Server with commands that use the LPR to LPD protocol, such as:
 - LPR and LPQ commands from z/OS, VM, z/VM®, IBM I, UNIX, and Windows systems
 - **lpr**, **enq** and **qstat** commands from AIX systems
 - **lprafp** command from Windows and UNIX systems
- Submit print requests to Infoprint Server with the Infoprint Port Monitor from Windows systems.
- Transform data sets that Print Interface does not allocate on the JES spool (for example, data sets that NetSpool allocates on the JES spool and that the z/OS system allocates on the JES spool) and that IP PrintWay basic mode transmits to remote printers. For information about how to select the IP PrintWay basic mode transform function in a printer definition, see the description of the **resubmit for filtering** option in *z/OS Infoprint Server Operation and Administration*.

Tip: You do not need to customize the Print Interface LPD to print from Windows systems that use the native Windows Server Message Block (SMB) protocol or from any platform that uses Internet Printing Protocol (IPP).

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.

This section describes these customization tasks. Required tasks are required by all installations that want to use the Print Interface LPD. Optional tasks are required only if the listed condition applies to your installation.

Table 19. Customization tasks for Print Interface LPD

Task	Condition	See page
Customizing z/OS Communications Server (TCP/IP)	Required	133
Editing the aopd.conf file for the LPD	Required	134
Editing the system configuration definition for the LPD	Optional: To specify dynamic attributes	135
Using the LPD compatibility filter	Optional: To obtain formatting similar to z/OS Communications Server LPD	136
Customizing LPR clients	Optional: To print from a remote client	137

Related tasks: After you customize the Print Interface LPD, do these tasks. For information about these tasks, see *z/OS Infoprint Server Operation and Administration*:

- Create a printer definition in the Printer Inventory for each printer and specify the fields in the printer definition that Print Interface requires.
- Run the **aopstart** command or the AOPSTART procedure to start the Print Interface LPD.

Customizing z/OS Communications Server (TCP/IP)

z/OS Communications Server also provides an LPD (LPSERVE) on the z/OS system. You must decide whether to run (1) *only* the Print Interface LPD or (2) *both* the Communications Server LPD and the Print Interface LPD on the z/OS system.

Guideline: 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. For more information, see “Running the Print Interface LPD and the Communications Server LPD on the same z/OS system.”

Running only the Print Interface LPD on the z/OS system

To run only the Print Interface LPD, and not the Communications Server LPD:

- To use 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. See “PORT and PORTRANGE statements” on page 118.
- Do not start the Communications Server LPD. For information about how to remove the LPSERVE procedure name from the AUTOLOG statement in the *hlq.PROFILE.TCPIP* data set, see “AUTOLOG statement” on page 118.
- Ask your administrator to create printer definitions in the Printer Inventory with the same name as the printers that are defined with SERVICE statements in the Communications Server LPD Server configuration data set, *hlq.SEXAINST(LPDDATA)*.
- 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. For information, see “Using the LPD compatibility filter” on page 136.

Running the Print Interface LPD and the Communications Server LPD on the same z/OS system

To run both the Print Interface LPD and the Communications Server LPD on the same z/OS system, you can either run (1) each LPD on a *separate* TCP/IP stack or (2) both LPDs on the *same* TCP/IP stack.

Guideline: 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 run 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 as described in “Editing the aopstart EXEC” on page 57. 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.

For more information about how to configure multiple TCP/IP stacks and set the `_BPXK_SETIBMOPT_TRANSPORT` variable, see:

- *z/OS Communications Server: IP Configuration Guide*
- *z/OS UNIX System Services Planning*

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.

Limitations: 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 Infoprint Port Monitor for Windows, however, is an exception. You can 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.

Editing the aopd.conf file for the LPD

The **aopd.conf** file contains attributes that Infoprint Server uses when it starts. The default location of this file is `/etc/Printsrv/aopd.conf`.

To start the LPD, you must specify the following attribute in the **aopd.conf** file:

```
start-daemons = { lpd }
```

To start the LPD, add **lpd** to the values in this attribute. Enclose the values in braces. If this attribute is not specified, the Printer Inventory Manager daemons and the LPD start.

Rule: If you change this attribute while Infoprint Server is running, start Infoprint Server again. You do not need to stop any Infoprint Server daemons before restarting.

Default: `start-daemons = {lpd}`

If dynamic configuration is not enabled, you can also specify the following attribute in the **aopd.conf** file:

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. 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.

Tips:

- Assign port 515 to the Print Interface LPD so that you can use all of the functions that Infoprint Server provides. These limitations apply if you assign a port other than port 515:
 - 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 Infoprint Port Monitor for Windows, however, is an exception. You can configure the Port Monitor to send print requests to any port.
 - IP PrintWay basic mode cannot transform data with the **resubmit for filtering** option. This is because IP PrintWay resubmits data 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.

Rules:

- Make sure that the port is not used by any other service on the same TCP/IP stack.
- This port number must not be reserved in the *hlq.PROFILE.TCPIP* data set. For more information about the PORT statement, see “Editing the *hlq.PROFILE.TCPIP* data set” on page 117.
- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: lpd-port-number = 515

Related information: For information about:

- How to edit the **aopd.conf** file, see “Steps for editing the Infoprint Server configuration file (aopd.conf)” on page 42.
- All attributes in the **aopd.conf** file, see Appendix B, “Infoprint Server files,” on page 389.
- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 68.

Editing the system configuration definition for the LPD

The system configuration definition in the Printer Inventory contains attributes that let you customize Infoprint Server while Infoprint Server is running. If dynamic configuration is enabled, you can specify this attribute in the system configuration definition to customize the LPD:

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. 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.

Tips:

- Assign port 515 to the Print Interface LPD so that you can use all of the functions that Infoprint Server provides. These limitations apply if you assign a port other than port 515:
 - 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 Infoprint Port Monitor for Windows, however, is an exception. You can configure the Port Monitor to send print requests to any port.
 - IP PrintWay basic mode cannot transform data with the **resubmit for filtering** option. This is because IP PrintWay resubmits data 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.

Rules:

- Make sure that the port is not used by any other service on the same TCP/IP stack.
- This port number must not be reserved in the *hlq.PROFILE.TCPIP* data set. For more information about the PORT statement, see “Editing the *hlq.PROFILE.TCPIP* data set” on page 117.
- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: lpd-port-number = 515

Example: This screen shows a portion of the ISPF panel that you use to edit the system configuration definition for the LPD:

```
Infoprint Server: System Configuration

Print Interface:

LPD port number . . 515
```

Related information: For information about:

- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 68.
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 80.

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.

For more information, see:

- *z/OS Infoprint Server Operation and Administration* for information about how to specify the LPD compatibility filter in a printer definition.
- *z/OS Infoprint Server User's Guide* for information about how to specify LPD compatibility filter options in the **filter-options** job attribute.

Customizing LPR clients

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 this information:

- The IP address or host name of the z/OS system where the Print Interface LPD is running. If you run more than one TCP/IP stack, this is the IP address or host name of the stack where the Print Interface LPD is running.
- The port number at which the Print Interface LPD is listening. The port is the value you specified in **lpd-port-number** attribute. The default port is 515.
- The name of the printer definition for the target printer defined in the Printer Inventory. This name is case-sensitive.
- The printer driver to use for the target printer.

For printing to AFP printers, Windows users can download the AFP Printer Driver for Windows. If you have specified either the PCL to AFP transform or the PostScript to AFP transform in the printer definition, users can use a PostScript or PCL printer driver because the transform converts the data to AFP format on the z/OS system.

- Whether the LPR must use a source port in the range of 721 to 731. Some LPDs require that the source port be in this range, which is defined by RFC 1179. However, the Print Interface LPD does *not* require that the LPR be in this range. Generally, performance on the system where the LPR runs is improved if you do not restrict the range for the source port.

For additional information about how to submit jobs from remote systems and how to configure the Infoprint Port Monitor for Windows, see *z/OS Infoprint Server User's Guide*.

Sending the control file first

Some LPRs let you specify whether the LPR should send the LPD control file first (before the data) or last (after the data file). The Print Interface LPD can receive the

LPD control file either first or last. However, IBM suggests that the LPR send the control file first because the Print Interface LPD uses less storage and can process the data file more quickly.

“Increasing the time-out value on an AIX system” shows an example of how to configure an AIX print queue to send the control file first.

Increasing the time-out value on an AIX system

When sending large files to the Print Interface LPD, especially when the printer definition specifies a data transform, users on an AIX system might need to change the time-out value used by the AIX system. The time-out value is the amount of time the AIX system waits for an acknowledgment from the LPD that transmission was successful. Because the Print Interface LPD returns an acknowledgement only after data is transformed to a format accepted by the printer and written successfully to the JES spool, the LPD might send an acknowledgement only after several minutes. If the document contains several gigabytes of data that need to be transformed, the LPD might send an acknowledgement only after an hour or more.

On an AIX system, the **rembak** command, which is used by the AIX queuing subsystem to send jobs to remote printers, uses a default time-out value of 90 seconds. IBM suggests that you change this default to a long time-out value when sending files to Infoprint Server, especially when you send large files that need to be transformed. To change the value, specify the **-T** option on the **rembak** command. You can specify the **-T** option in the **/etc/qconfig** file in the **backend** option under the queue device name.

This example specifies a timeout value of 5 minutes. This example also shows these options:

- **-X** option, which lets users send Infoprint Server job attributes on the **enq -o** option.
- **-C** option, which sends the control file first, before the data.

```
backend = /usr/lib/lpd/rembak -T 5 -X -C
```

After you make this change, delete the **/etc/qconfig.bin** file. Then use these commands to stop and restart the queue daemon:

```
stopsrc -s qdaemon  
startsrc -s qdaemon
```

Customizing the z/OS SMB server to use Print Interface

Print Interface, with support that the z/OS SMB server provides, lets Windows users print using the standard Windows Server Message Block (SMB) protocol. Windows users can print on any printer defined in the Printer Inventory. 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.

Tip: As an alternative to using SMB protocol, Windows users can use the Internet Printing Protocol or the Infoprint Port Monitor for Windows. For a comparison that can help you decide which method to use to print from Windows systems, see Chapter 9, “Installing and customizing the Infoprint Server Windows client,” on page 369.

Table 20 summarizes the tasks to customize the z/OS SMB server. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies to your installation.

Table 20. Customization tasks for SMB

Task	Condition	See page
Customizing the z/OS SMB server	Required	139

Related tasks:

- Run the **aopstart** command or the AOPSTART procedure to start Infoprint Server.
- Create a printer definition in the Printer Inventory for each printer that you plan to define to SMB. In the printer definitions, specify the printer attributes required by Print Interface. For information, see *z/OS Infoprint Server Operation and Administration*.

Customizing the z/OS SMB server

Before Windows users can print using SMB protocol, you must customize the z/OS SMB server. As part of SMB customization, do these tasks, which are more fully described in *z/OS Distributed File Service SMB Administration*:

- Create an SMB *printer share* for each printer definition in the Printer Inventory to which Windows users need to print. When you create the printer share, consider these points:
 - For the name of the printer share, you can use either the name of the printer definition or a different name. Using the same name can make it easier to correlate the SMB printer name with the printer definition in the Infoprint Server Printer Inventory.
 - For the printer type, specify the printer driver for the printer. The printer driver must reside on the client's system. If the printer is an AFP printer, you can download the AFP printer driver. As an alternative, if the administrator has configured the printer definitions in the Printer Inventory to use the PCL or PostScript (and PDF) to AFP data transforms, you can specify generic PCL or PostScript drivers for AFP printers.
- Add the directory that contains Infoprint Server library files to the LIBPATH variable in the SMB environment variable file. If you installed Infoprint Server in the default directory, the directory name is **/usr/lpp/Printsrv/lib**. SMB attempts to use Infoprint Server if it finds the Infoprint Server library file named **aopapi.dll** in one of the directories contained in the LIBPATH variable. Therefore, *before* you add the Infoprint Server library directory to the LIBPATH variable and start SMB, make sure that Infoprint Server is started and that the administrator has created the required printer definitions.

If you decide not to use Infoprint Server with SMB, do not add the Infoprint Server library directory to the LIBPATH variable in the SMB environment file. If the Infoprint Server library directory is already specified in the LIBPATH variable, remove it.

Customizing the Print Interface IPP Server

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. For the job attributes supported by the IPP Server, see Table 22 on page 144. The IPP Server can also provide job status to the IPP client. The IPP Server does *not*, however, support canceling a print request.

Tip: 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.

This section describes these required and optional tasks to customize the IPP Server:

Table 21. Customization tasks for Print Interface IPP Server

Task	Condition	See page
Ensuring sufficient memory is available to start the IPP Server	Required	140
Editing the <code>aopd.conf</code> file for the IPP Server	Required	141
Setting environment variables for the IPP Server	Optional: To change the default configuration	142
Editing the system configuration definition for the IPP Server	Optional: To specify dynamic attributes	142
Customizing the IPP workstation client	Required	143

Related tasks: After you customize the Print Interface IPP Server, do these tasks. For information about these tasks, see *z/OS Infoprint Server Operation and Administration*:

- Create a printer definition in the Printer Inventory for each printer and specify fields in the printer definition that Print Interface requires.
- Run the `aopstart` command or the AOPSTART procedure to start the IPP Server daemon.

Ensuring sufficient memory is available to start the IPP Server

The IPP Server requires at least 200 megabytes (MB) of memory. Therefore, you might need to increase the region size before you start the IPP Server. The way you increase the region size depends on how you start Infoprint Server:

- **AOPSTART procedure:** Specify a region size of 200 MB in the REGION parameter on the EXEC statement in the AOPSTART JCL procedure. If the REGION parameter is not specified, the default region size defined for your installation is used.
- **aopstart command:** If you enter the `aopstart` command on the z/OS UNIX command line, make sure the MAXASSIZE parameter in the BPXPRMxx member of SYS1.PARMLIB is at least 200 MB. The MAXASSIZE parameter determines the system-wide maximum region size for an address space. (You can set the MAXASSIZE parameter dynamically with the SETOMVS command.)

Examples:

- This statement sets the maximum region size to 200 MB:

```
MAXASSIZE(209715200) /* 200*1024*1024 = 200MB */
```

- This command sets the maximum region size to 200 MB:

```
SETOMVS MAXASSIZE=209715200
```

Tips:

- The ASSIZEMAX parameter of the RACF ADDUSER and ALTUSER commands, which lets you specify the maximum region size for a particular user, overrides the MAXASSIZE parameter.
- The z/OS IEFUSI user exit can modify the region size of an address space. IBM strongly discourages you from altering the region size of address spaces in the OMVS subsystem category. For more information about the IEFUSI exit, see *z/OS MVS Installation Exits*.

If you enter the **aopstart** command on the TSO/E OMVS command line, make sure the SIZE parameter on the TSO/E LOGON panel specifies at least 200 MB. The SIZE parameter determines the maximum region size for an address space.

Editing the aopd.conf file for the IPP Server

The **aopd.conf** file contains attributes that Infoprint Server uses when it starts. The default location of this file is **/etc/Printsrv/aopd.conf**.

To start the IPP Server, you must specify the following attribute in the **aopd.conf** file:

start-daemons = { ippd }

To start the IPP Server daemon, add **ippd** to any other values in this attribute. Enclose the values in braces. This attribute is required.

Rule: If you add **ippd** while Infoprint Server is running, start Infoprint Server again. You do not need to stop any Infoprint Server daemons before restarting.

Default: start-daemons = {lpd}

Example: start-daemons = {lpd ippd}

If dynamic configuration is not enabled, you can also specify the following attribute in the **aopd.conf** file:

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.

Rules:

- Make sure that the port is not used by any other service on the same TCP/IP stack.
- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: ipp-port-number = 631

Related information: For information about:

- How to edit the **aopd.conf** file, see “Steps for editing the Infoprint Server configuration file (aopd.conf)” on page 42.

- All attributes in the **aopd.conf** file, see Appendix B, “Infoprint Server files,” on page 389.
- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 68.

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. For more information, see “Editing the aopstart EXEC” on page 57.

CLASSPATH

The full path names of IPP Server Java Archive (JAR) files. If you installed Infoprint Server in the default directory, **/usr/lpp/Printsrv**, you do not need to set this environment variable. If you installed Infoprint Server in a different directory, specify these values:

- *directory/classes/ipp.jar*
- *directory/classes/ippreal.jar*
- *directory/classes/ippserver.jar*

directory is the directory that contains Infoprint Server files.

Default:

```
/usr/lpp/Printsrv/classes/ipp.jar:
/usr/lpp/Printsrv/classes/ippreal.jar:
/usr/lpp/Printsrv/classes/ippserver.jar
```

JAVA_HOME

The path used to locate Java files. The IPP Server can use Java 7.0, Java 6.0, or Java 6.0.1. This environment variable is optional. However, you must set it if you did not install Java files in the default Java directory.

JAVA_HOME must specify the path of the 31-bit version of Java because the IPP Server does not support the 64-bit version.

Default: The IPP Server uses Java files in these directories, in the order shown:

1. */usr/lpp/java/J7.0*
2. */usr/lpp/java/J6.0*
3. */usr/lpp/java/J6.0.1*

Example: JAVA_HOME=*/usr/lpp/java/J7.0*

Editing the system configuration definition for the IPP Server

The system configuration definition in the Printer Inventory contains attributes that let you customize Infoprint Server while Infoprint Server is running. If dynamic configuration is enabled, you can specify this attribute in the system configuration definition to customize the IPP Server:

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.

Rules:

- Make sure that the port is not used by any other service on the same TCP/IP stack.

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: ipp-port-number = 631

Example: This screen shows a portion of the ISPF panel that you use to edit the system configuration definition for the IPP Server:

Infoprint Server: System Configuration

Print Interface:
 IPP port number . . 631

Related information: For information about:

- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 68.
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 80.

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 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 system, the user must define the printer to Windows as an Internet printer. The user must also specify the Uniform Resource Identifier (URI) of the printer. The URI of any printer defined in the Printer Inventory has this format:

http://host:port/printers/printername

host The host name or IP address of the z/OS system.

port The port number where the IPP Server is listening. Specify the same number as specified in the **ipp-port-number** attribute. If the **ipp-port-number** attribute is not set, specify 631, which is the default value for the attribute.

printers A qualifier required by the Print Interface IPP Server. This qualifier is case-sensitive. You can replace printers with `servlet/IPPServlet`. For example, this URI also works:
http://host:port/servlet/IPPServlet/printername

printername The name of the printer definition in the Printer Inventory. The printer definition name is case-sensitive. You must use the same uppercase and lowercase letters as are stored in the Printer Inventory.

Example: If the IP address of the z/OS system is 9.99.1234, the IPP Server is listening at port 631, and the name of the printer definition is myprinter, use this URI:

http://9.99.1234:631/printers/myprinter

Guidelines:

1. Some IPP clients might also support the **ipp** format of the URI. In this case, the URI can have this format:

ipp://host[:port]/printers/printername

When you use the **ipp** format for the URI and omit the port, the default is 631, which is the default port at which the Print Interface IPP Server listens.

Example: If the IP address of the z/OS system is 9.99.1234, the Print Interface IPP Server is listening at port 631, and the name of the printer definition is "myprinter", use this URI:

ipp://9.99.1234/printers/myprinter

2. For detailed information about how to add an Internet printer to a Windows workstation, see *z/OS Infoprint Server User's Guide*.
3. The Print Interface IPP Server cannot receive data that has been encrypted. Therefore, the user should make sure that the IPP client does not encrypt data.

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. Table 22 lists the IPP job attributes that the Print Interface IPP Server supports for all printers and the corresponding Infoprint Server job attribute that it sets. For information about these Infoprint Server job attributes, see *z/OS Infoprint Server User's Guide*.

Table 22. IPP attributes and corresponding Infoprint Server attributes

IPP job attribute	Infoprint Server job attribute
copies	copies
job-name	title-text
job-priority	jes-priority
requesting-user-name	owner¹
1. 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.	

Customizing SAP R/3

Print Interface lets SAP R/3 users print on any printers defined in the Printer Inventory. If the printer is an AFP printer, Print Interface can automatically invoke a transform such as a PCL to AFP, PostScript to AFP, or SAP to AFP transform. The transform you need depends on the type of output the SAP R/3 output device creates.

Depending on where your SAP R/3 application servers run, different Print Interface components handle the print requests and provide different functions:

- An SAP R/3 application server runs on the same z/OS system as Infoprint Server

When an SAP R/3 application server runs on the same z/OS system as Infoprint Server, the SAP Output Management System (OMS) provided with Print Interface handles print requests. The SAP OMS lets users cancel jobs and obtain accurate job status. The SAP Callback daemon provided with Print Interface

provides immediate notification when documents have completed successfully or unsuccessfully. The Callback daemon also can provide notification to SAP R/3 applications running on other systems if the SAP R/3 spool work process runs on the same z/OS system as Infoprint Server. Figure 4 on page 10 shows this configuration.

- An SAP R/3 application server does not run on the same z/OS system as Infoprint Server

When no SAP R/3 application server runs on the same z/OS system as Infoprint Server, the Print Interface LPD handles print requests. Users *cannot* receive notification about their print requests or cancel jobs. Users can receive job status. However, the job status is not as accurate as when the SAP R/3 application server runs on the same z/OS system as Infoprint Server and the Infoprint Server OMS is used. Figure 5 on page 11 shows this configuration.

This table summarizes the customization tasks when an SAP R/3 application server runs on the *same* z/OS system as Infoprint Server:

Table 23. Customization tasks for Print Interface when SAP R/3 application server runs on same z/OS system

Task	Condition	See page
Defining an SAP user for the callback daemon	Required	146
Creating the SAP callback daemon configuration file (aopsapd.conf)	Required	146
Setting environment variables for the SAP OMS	Required	148
Defining a ROMS and LOMS to the SAP R/3 application server	Required	148
Modifying the LOMS to specify job attributes	Optional: To specify Infoprint Server job attributes	149
Creating an SAP R/3 output device	Required	150

This table summarizes the customization tasks when an SAP R/3 application server does *not* run on the same system as Infoprint Server. For example, the application server might run on an AIX or Windows system.

Table 24. Customization tasks for Print Interface when SAP R/3 application server does not run on same z/OS system

Task	Condition	See page
Creating an SAP R/3 output device	Required	150
Customizing the Print Interface LPD	Required	132

Related tasks: After you customize Print Interface for printing from SAP R/3, do these tasks:

- If you use the SAP Callback daemon, stop and restart all Infoprint Server daemons. The SAP Callback daemon, **aopsapd**, starts automatically when the first print request is received from the SAP R/3 Application Server for z/OS. For information, see *z/OS Infoprint Server Operation and Administration*.
- Create a printer definition in the Printer Inventory for each printer. In the printer definition, specify fields that Print Interface requires. For information, see *z/OS Infoprint Server Operation and Administration*.

- If the printer is an AFP printer, customize the SAP to AFP, PCL to AFP, or PostScript to AFP transform. The transform you need depends on the type of output the SAP R/3 output device creates. For information about these transforms, see *IBM Infoprint Transforms to AFP for z/OS*.

Defining an SAP user for the callback daemon

You must define an SAP user that the Print Interface SAP Callback daemon can use to log on to the SAP R/3 Application Server for z/OS. Define an SAP user to each SAP R/3 application server at which you want to receive job callback notification. To define a user for the Callback daemons:

1. From the SAP R/3 GUI, select **Tools --> Administration --> User maintenance --> Users**.
-
2. The user must be a CPIC type with XOMI permissions to log on to the CCMS system management XOM application programming interface. You can provide these authorizations in the S_XMI_XOM_A profile.
-

Tip: If you change the SAP user, make the same changes in the **aopsapd.conf** configuration file. Then stop and restart Infoprint Server daemons so that your changes take effect.

Creating the SAP callback daemon configuration file (aopsapd.conf)

The SAP Callback daemon configuration file, **aopsapd.conf**, contains information that the SAP Callback daemon needs to log on to the SAP R/3 application servers. This file is required.

To create the configuration file:

1. Copy the sample configuration file, **/usr/lpp/Printsrv/samples/aopsapd.conf**, to **/etc/Printsrv/aopsapd.conf**. For example, this z/OS UNIX command copies the sample configuration file to the default location:

```
cp /usr/lpp/Printsrv/samples/aopsapd.conf /etc/Printsrv/aopsapd.conf
```

Tip: A sample job is provided in SYS1.SAMPLIB(AOPCPETC) to copy all Infoprint Server sample configuration files from **/usr/lpp/Printsrv/samples** to the **/etc/Printsrv** directory, including the SAP Callback configuration file. Review the comments in the sample job to determine which configuration files your installation requires, and then run the job to create the configuration files.

2. Because this file contains information about how to log on to the SAP R/3 application servers, set the permissions so that only administrators can read the file. Set the owner to the SAP administration ID and set permissions to 0700. Use the **chown** and **chmod** commands to define the owner and set permissions. For example, if the SAP administration ID is **mike**, enter these commands:

```
chown mike /etc/Printsrv/aopsapd.conf
chmod 0700 /etc/Printsrv/aopsapd.conf
```

3. Edit the file for your installation using your preferred editor and save your changes. For example:

```
oedit /etc/Printsrv/aopsapd.conf
```

Edit this file using the IBM-1047 code page.

-
4. If you did *not* create the configuration file in directory `/etc/Printsrv/`, specify the full path name of the configuration file in the `AOPSAPD_CONF` environment variable in the `aopstart` EXEC and in the `/etc/profile` file.

-
5. If you edit the configuration file while Infoprint Server is running, stop and restart all Infoprint Server daemons. For example, enter these MVS commands to run the AOPSTOP and AOPSTART JCL procedures:

```
START AOPSTOP
START AOPSTART
```

Also, restart the SAP R/3 Logical Output Management System (LOMS).

Format of an entry in the SAP callback daemon configuration file

In the `aopsapd.conf` file, create one entry for each SAP R/3 system that contains an Infoprint Server job callback LOMS. An entry consists of a set of attributes, starting with the `system` attribute and ending with a semicolon. All attributes are required.

```
# comment
system system-name
    initial-target = instance-name
    client = clientname
    userid = userid
    password = password
    language = language
;
```

system *system-name*

The name of the SAP R/3 system that contains an LOMS for which you want callback notification. The SAP R/3 system can run on any z/OS or non-z/OS system. This attribute must be the first attribute in the entry.

initial-target = *instance-name*

The name of an SAP R/3 server instance on the SAP R/3 system defined in the `system` attribute. The Print Interface Callback daemon initially contacts this server instance to obtain configuration information about the LOMs defined on that SAP R/3 system.

client = *clientname*

The name of the client defined to SAP for use by the Callback daemon.

userid = *userid*

The user ID defined to SAP for use by the Callback daemon.

password = *password*

The password defined to SAP for use by the user ID.

language = *language*

The language used for messages from the SAP R/3 application server.

Note: The client name, user ID, and password must match the values specified when you defined an SAP user for the OMS Callback daemon. See “Defining an SAP user for the callback daemon” on page 146.

Example -- SAP callback daemon

This example shows two sample entries in the `aopsapd.conf` configuration file.

```
system EZ0
  initial-target = boefcs1_EZ0_00
  client = 000
  userid = mike
  password = xxxxxxxx
  language = EN
;

system EX0
  initial-target = boefcs5_EX0_00
  client = 000
  userid = mike
  password = xxxxxxxx
  language = EN
;
```

Syntax rules

When editing the SAP Callback daemon configuration file, follow these syntax rules:

- End the last attribute in each entry with a semicolon.
- Use lowercase characters for the attribute names and uppercase or lowercase characters for the attribute values.
- Code blank characters before or after the equal sign if desired.
- If a value contains blank characters or special characters (such as {}-> =, #), enclose the value in single or double quotation marks.
- Start comments with the # symbol.
- Include blank lines if desired.

Setting environment variables for the SAP OMS

This environment variables affect the behavior of the SAP OMS. Set these environment variables in the `aopstart` EXEC and in the `/etc/profile` file. For information, see “Editing the aopstart EXEC” on page 57.

AOPSAPD_CONF Specify the full path name of the SAP Callback daemon configuration file. If the configuration file is in `/etc/Printsrv/aopsapd.conf`, you do not need to set this environment variable.

Default: `/etc/Printsrv/aopsapd.conf`

LIBPATH Add the directory that contains the SAP remote function calls DLL (`librfc.dll`).

Defining a ROMS and LOMS to the SAP R/3 application server

To define the Infoprint Server Output Management System (OMS) to an SAP R/3 application server, the SAP administrator must define these entries:

- An SAP Real Output Management System (ROMS)
The ROMS defines the general capabilities of the OMS and global configuration information.
- One or more SAP Logical Output Management Systems (LOMS)
An LOMS lets you customize the capabilities for a particular set of SAP output devices.

To define a ROMS and two LOMSs that interface with Infoprint Server, import sample file `/usr/lpp/Printsrv/samples/infoprint.oms` into SAP R/3. This sample file defines a ROMS with this name and description:

- INFOPRINT: IBM Infoprint Server for OS/390®

It creates two LOMSs with these names and descriptions:

- IPS001: Infoprint Server with callback: This LOMS provides job callback notification.
- IPS002: Infoprint Server with polling: This LOMS does not provide job callback notification.

To import the file, do these steps (these steps might be different on your SAP GUI):

1. Log on to the SAP R/3 Application Server for z/OS.

2. Select **Tools --> CCMS --> Spool --> SPAD --> Spool administration**.

3. On the Spool Administration screen, select **Utilities --> To output management system --> Import**.

4. On the Spool: Saving and Loading of Definitions screen:
 - Type **/usr/lpp/Printsrv/samples/infoprint.oms** in the **Export/Import file name** field.
 - Select **Import**.
 - Select the **Execute import** and **Generate log** options.

After you import the ROMS and LOMSs, do these steps to edit them:

1. Edit the INFOPRINT ROMS. For the initialization instance, select the SAP R/3 Application Server for z/OS that runs on the same z/OS system as Infoprint Server.

2. Edit the IPS001 and IPS002 LOMSs. For the tasking target, select the SAP R/3 Application Server for z/OS that runs on the same z/OS system as Infoprint Server.

3. Edit the IPS001 LOMS. For the callback target instance, specify any server instance in the SAP R/3 system.

Modifying the LOMS to specify job attributes

The Infoprint Server LOMS contains **aopoms** commands that do the SUBMIT, DPOLL, DQUERY, JQUERY and CANCEL OMS commands. If you want to specify Infoprint Server job attributes for print requests, you can modify the **aopoms** command that does the SUBMIT command. For example, you can specify the **duplex** job attribute to obtain duplex printing on an AFP printer.

The format of the **aopoms** command that does the SUBMIT command is:

```
aopoms -c submit -d printer -j sap-spoolid -r sap-rmg -s sap-system
        -T sap-callback-target [-t title] [-n copies] [-o options] file ...
```

The options are:

- d printer**
Name of the printer definition for the target printer.
- j sap-spoolid**
SAP R/3 spool ID to be stored in the job entry.
- n copies**
Number of copies to be printed, from 1 to 255.
- o option**
One or more attribute value assignments in the format: *attribute=value*.
Separate each attribute value assignment with spaces. You can specify any job attributes defined in the *z/OS Infoprint Server User's Guide*.
- r sap-rmg**
The SAP R/3 **rmg** ID to be stored in the job entry.
- s sap-system**
The name of the SAP R/3 system.
- T sap-callback-target**
The name of the SAP R/3 server that is the target server for job callbacks.
- t title**
A title for the document. This option is equivalent to Infoprint Server **title** job attribute.

The **aopoms** commands are defined in the LOMS as templates with replacement variables used to supply specific information. The generic form of the **aopoms** SUBMIT command as defined in the Infoprint Server LOMS is:

```
aopoms -c submit -j &EI -r &EG -s &Es -T &ES -d &P -t"&T" -n &C &F
```

To specify Infoprint Server job attributes, for example the **jes-priority** job attribute, specify these **aopoms** command on one line:

```
aopoms -c submit -j &EI -r &EG -s &Es -T &ES -d &P -t ""&T" -n &C -o "jes-priority = 4" &F
```

Creating an SAP R/3 output device

To print from SAP R/3, the SAP R/3 administrator must create an SAP R/3 output device for each target printer. You might want to create more than one SAP R/3 output device for the same printer. For example, you might want to customize the LOMS differently for each SAP output device, or you might want to associate each SAP output device with a different printer definition for the same target printer. (The administrator can create more than one printer definition in the Printer Inventory for the same printer to specify different job characteristics, such as a different form definition name.)

To create an SAP R/3 output device, you need this information:

- The name of the SAP R/3 Application Server for z/OS.
- The name of the printer definition in the Printer Inventory. This name is case-sensitive.
- The data format accepted by the printer, for example, AFP, PCL, or PostScript. The data format determines the device driver you select.
- The name of the LOMS defined to SAP R/3. Default names for Infoprint Server LOMSs are:
 - **Infoprint Server with callback**
Use this LOMS if you want immediate notification of job events.
 - **Infoprint Server with polling**

Use this LOMS if you do not need immediate notification of job events. The Callback daemon returns notification only when polled by the SAP R/3 system.

- The host spool access method:
 - Use access method **E** when the SAP application server runs on the same system as Infoprint Server
 - Use access method **U** when the SAP application server runs on a different system from Infoprint Server
- For access method **U**, you also need:
 - The host name or IP address of the z/OS system where Infoprint Server runs.
 - The port number where the Print Interface LPD listens (default port is 515).

The next sections contain instructions for creating an SAP output device. Use the instructions for the correct host spool access method: **E** or **U**. The screen and field names in the instructions might be different on your SAP R/3 GUI.

Creating an SAP output device using access method **E**

If the SAP R/3 application server runs on the same z/OS system as Infoprint Server, do these steps to create an SAP R/3 output device from the SAP R/3 GUI:

1. On the Spool Administration: Initial Screen, select **Output devices**.

2. Enter change mode by pressing F8 or selecting the Change icon (a pencil).

3. On the Spool Administration: List of Output Devices screen, select **Output device --> Create**.

4. On the Spool Administration: Create Output Device screen, fill in these fields:
 - **Output device:** Type a name for the printer.
 - **Short name:** Type a short name.
 - **Device Type:** Select a printer driver that creates a data stream that the printer accepts. Or, select a printer driver that creates a data stream that can be transformed into a format that the printer accepts.
 - For AFP printers, select one of these options:
 - PCL: Select if you use the PCL to AFP transform.
 - PostScript: Select if you use the PostScript to AFP transform.
 - SAPGOF (Generic output format): Select if you use the SAP to AFP transform.
 - Tip:** If you use Infoprint Transforms to AFP V2.2, do not select SAPGOF_E (Generic output format EBCDIC) because V2.1 of the SAP to AFP transform does not accept EBCDIC output.
 - For other printers, select the PCL or PostScript option.
 - **Spool server:** Select the name of the SAP R/3 application server.

5. On the HostSpoolAccMethod tab, fill in these fields:
 - **Host spool access method:** Select **E: External output management system**
 - **Logical OMS:** Select an Infoprint Server LOMS.

- **Host printer:** Type the name of the printer definition in the Printer Inventory.
-

6. Save your changes.

Creating an SAP output device using access method U

If the SAP R/3 application server runs on a non-z/OS system, do these steps to create an SAP R/3 output device from the SAP R/3 GUI:

1. On the Spool Administration: Initial Screen, select **Output devices**.

2. Enter change mode by pressing F8 or selecting the Change icon (a red pencil).

3. On the Spool Administration: List of Output Devices screen, select **Output device --> Create**.

4. On the Spool Administration: Create Output Device screen, fill in these fields:
 - **Output device:** Type a name for the printer.
 - **Short name:** Type a short name for the printer.
 - **Device Type:** Select a printer driver that creates a data stream that the printer accepts. Or, select a printer driver that creates a data stream that can be transformed into a format that the printer accepts.
 - For AFP printers, select one of these options:
 - PCL: Select if you use the PCL to AFP transform.
 - PostScript: Select if you use the PostScript to AFP transform.
 - SAPGOF (Generic output format): Select if you use the SAP to AFP transform.

Tip: You can instead select SAPGOF_E (Generic output format EBCDIC). However, the SAP to AFP transform might not accept EBCDIC output in future releases.
 - For other printers, select the PCL or PostScript option.
 - **Spool server:** Select the name of the SAP R/3 application server.

5. On the HostSpoolAccMethod tab, fill in these fields:
 - **Host spool access method:** Select **U: Print using Berkley protocol**.
 - **Logical OMS:** Select an Infoprint Server LOMS.
 - **Host printer:** Type the name of the printer definition in the Printer Inventory.
 - **Destination host:** Type the address of the z/OS system.
 - **Port number:** Type the port where the Infoprint Server LPD listens.

6. Save your changes.

Viewing job status and job events, querying jobs, canceling jobs

After you create an SAP R/3 output device, SAP R/3 users can print on an Infoprint Server printer, view job status, view job events, query jobs, and cancel jobs. These considerations apply when users make these print requests:

- View current information and all events for a job:
 - If the status indicates the print job has completed, the print job might still remain on the JES spool.
 - If the status indicates the print job is waiting in the host spooler, the position in the queue does not reflect the order in which the job will be printed. For example, jobs not submitted through Infoprint Server might print before the job.
- Display device information:
 - Print jobs submitted to the same printer using a different printer definition name in the Printer Inventory are not listed.
 - The position of the job in the list does not reflect the order in which the job will be printed.
- Cancel a print job:

Users can cancel a print job only if Infoprint Server has not yet selected it for processing.

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 job submitter 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.

This table summarizes the customization tasks for the Print Interface subsystem. These tasks are required by all installations that want to use the Print Interface subsystem.

Table 25. Customization tasks for Print Interface subsystem

Task	Condition	See page
Editing the <code>aopd.conf</code> file for the Print Interface subsystem	Required	154
Increasing the maximum number of active transform daemons	Optional	154

Related tasks: After you customize the Print Interface subsystem, do these tasks. For information about these tasks, see *z/OS Infoprint Server Operation and Administration*:

- Create a printer definition in the Printer Inventory for each printer, or for each set of related printers, and specify fields in the printer definition that Print Interface and the transforms require.
- Run the **aopstart** command or the AOPSTART procedure to start the Print Interface subsystem daemon.

Editing the aopd.conf file for the Print Interface subsystem

The **aopd.conf** file contains attributes that Infoprint Server uses when it starts. The default location of this file is **/etc/Printsrv/aopd.conf**.

To start the Print Interface subsystem, you must specify this attribute in the **aopd.conf** file:

start-daemons = { subd }

To start the Print Interface subsystem daemon, add **subd** to the values in this attribute. Enclose the values in braces. This attribute is required.

Rule: If you add **subd** while Infoprint Server is running, start Infoprint Server again. You do not need to stop any Infoprint Server daemons before restarting.

Default: start-daemons = {lpd}

Example: In this example, the Printer Inventory Manager, the Infoprint Server Transform Manager, and the Print Interface subsystem start:

```
start-daemons = {xfd subd}
```

This attribute in the **aopd.conf** file has special considerations when you use the Print Interface subsystem:

inventory = inventory

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. It *cannot* be a lowercase letter or numeric. The remaining characters can be uppercase letters or numeric. The job submitter specifies this name in the SUBSYS parameter on the DD JCL statement.

This name must exactly match the inventory name in the NetSpool, IP PrintWay basic mode, and PSF start procedures (if PSF uses the Printer Inventory).

Rules:

- Do not change this attribute while Infoprint Server (including NetSpool and IP PrintWay basic mode) is running. Also, if PSF for z/OS uses the Printer Inventory, do not change this attribute while PSF is running.
- If you use Infoprint Central and change this attribute, restart the z/OS HTTP Server.

Default: inventory = AOP1

Related information: For information about:

- All attributes in the **aopd.conf** file, see Appendix B, “Infoprint Server files,” on page 389.
- How to edit the **aopd.conf** file, see “Steps for editing the Infoprint Server configuration file (aopd.conf)” on page 42.

Increasing the maximum number of active transform daemons

If your 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 canceled 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.

To increase the maximum number of active transforms:

1. Edit the **maximum-active** attributes in the transform configuration file, **aopxfd.conf**.

-
2. Stop and restart the Infoprint Server Transform Manager daemon so that your changes take effect. For example, enter these MVS commands to run the AOPSTOP and AOPSTART JCL procedures:

```
START AOPSTOP,OPTIONS='-d xfd'  
START AOPSTART
```

For more information about the transform configuration file, see “aopxfd.conf -- Infoprint Server transform configuration file” on page 404.

Chapter 5. Customizing Transform Interface

This information describes how to customize the Transform Interface component of Infoprint Server. Transform Interface communicates with transforms that IBM and Ricoh provide. For an overview of Transform Interface, see “Transform Interface” on page 19 and “Transforms” on page 20.

Table 26 summarizes the tasks to customize the Transform Interface. Required tasks are required by all installations that use the indicated transforms. Optional tasks are required only if the listed condition applies to your installation.

Table 26. Customization tasks for Transform Interface

Task	Condition	See page
Identifying software requirements for Transform Interface	Required	158
Editing the aopd.conf file for the Transform Manager	Optional: To use IBM transforms that run on the z/OS system	158
Creating the transform configuration file (aopxfd.conf)	Optional: To use IBM transforms that run on the z/OS system	159
Creating the remote transform configuration file (aoprxf.conf)	Optional: To use Ricoh transforms that InfoPrint Transform Manager for Linux manages	165
Setting environment variables for Transform Interface	Optional: To change the default location of configuration files	167
Customizing the TCP/IP network firewall	Optional: To use transforms that run on a remote system	168
Setting up security for transform resources	Optional: To use transform resources	168
Creating a code page conversion table	Optional: To use Ricoh transforms that InfoPrint Transform Manager for Linux manages with a custom EBCDIC code page for the z/OS locale	169

Table 27 lists other components of Infoprint Server that you might need to customize:

Table 27. Other components to customize for Transform Interface

Task	Condition	See page
Customizing the Printer Inventory Manager	Required	27
Customizing Print Interface	Optional: To transform data submitted to Print Interface or to IP PrintWay basic mode	115

Table 27. Other components to customize for Transform Interface (continued)

Task	Condition	See page
Customizing IP PrintWay	Optional: To transform data submitted to IP PrintWay basic mode or extended mode	199

Related tasks: After you customize Transform Interface, do these tasks. For information about these tasks, see *z/OS Infoprint Server Operation and Administration*:

- Specify the appropriate transform filter in the **Filter** field of the printer definitions.

Tip: If you run IP PrintWay basic mode, also select the **resubmit for filtering** option in the printer definition.

- If you edited the Infoprint Server configuration file to start the Infoprint Server Transform Manager, restart Infoprint Server.

Identifying software requirements for Transform Interface

One or more of these transform products is required:

- IBM Infoprint Transforms to AFP for z/OS (5655-N60)
- IBM Print Transform from AFP to PCL for Infoprint Server for z/OS (5655-TF2)
- IBM Print Transform from AFP to PDF for Infoprint Server for z/OS (5655-TF1)
- IBM Print Transform from AFP to PostScript for Infoprint Server for z/OS (5655-TF3)
- IBM Infoprint XT for z/OS (5655-O15)
- Ricoh InfoPrint Transform Manager for Linux
- Ricoh InfoPrint Manager for AIX
- Ricoh InfoPrint Manager for Windows

Editing the aopd.conf file for the Transform Manager

The **aopd.conf** file contains attributes that Infoprint Server uses when it starts. The default location of this file is **/etc/Printsrv/aopd.conf**.

To start the Infoprint Server Transform Manager, you must specify the following attribute in the **aopd.conf** file:

start-daemons = { xfd }

Add **xfd** to the existing values in this attribute to start the Infoprint Server Transform Manager daemon, **aopxfd**. Enclose all values in braces.

Rule: After you add **xfd** and create the transform configuration file, start Infoprint Server again. You do not need to stop any Infoprint Server daemons before restarting.

Default: start-daemons = {lpd}

Example: In this example, the Printer Inventory Manager, the LPD, and the Infoprint Server Transform Manager start:

```
start-daemons = {lpd xfd}
```

Note: You must start the Infoprint Server Transform Manager if you run IBM transforms on z/OS systems. If you run transforms that Ricoh InfoPrint

Transform Manager for Linux or Ricoh InfoPrint Manager for AIX and Windows provide, you do not need to start the Infoprint Server Transform Manager.

Related information: For information about:

- All attributes in the **aopd.conf** file, see Appendix B, “Infoprint Server files,” on page 389.
- How to edit the **aopd.conf** file, see “Steps for editing the Infoprint Server configuration file (aopd.conf)” on page 42.

Creating the transform configuration file (aopxfd.conf)

If you use IBM transforms that run on the z/OS system, you must create the transform configuration file, **aopxfd.conf**, which contains information that the Infoprint Server Transform Manager uses to manage transforms that run on the z/OS system. You must create the transform configuration file before you start the Infoprint Server Transform Manager.

If you only transform data with Ricoh InfoPrint Transform Manager for Linux or Ricoh InfoPrint Manager for AIX/Windows, do not create this file.

In the transform configuration file, create at least one entry for each transform. In the transform entry you can specify:

- Attributes that the Infoprint Server Transform Manager uses to manage the transform.
- Environment variables for the transform. Each transform uses different environment variables. For information about the environment variables you can specify for each transform, see the documentation for the transform product.

These attributes and environment variables apply whether Infoprint Server invokes a transform automatically or whether the user invokes a transform with a z/OS UNIX transform command.

Transform classes

If you want to specify different environment variables for the same transform for different printing situations, you can create different classes of the transform. For example, if you want to transform PCL data to AFP format for printers that have different resolutions or paper sizes, you can define a transform class for each combination of printer resolution and paper size.

In the **aopxfd.conf** file, you create a separate entry for each 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:

```
pc12afp.d11 -c letter_300
```
- The job submitter specifies the transform class on the z/OS UNIX transform command in the **-c** option. For example:

```
pc12afp -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 AOPPRINT procedure. For example:

```
lp -d myprinter -o "filter-options='-c letter_300'" myfile.pcl
```

Format of an entry in the transform configuration file

Each transform entry consists of a set of attributes, starting with the **transform** attribute and ending with a semicolon. This section shows the general format of a transform entry and defines the attributes in the transform entry that are common for all transforms.

```
# comment
transform transformname[_transformclass]
  start-command = "daemon [ option ]..."
  [ environment = {name -> value [ name -> value ]...} ]
  [ maximum-active = number ]
  [ maximum-idle-time = seconds ]
  [ minimum-active = number ]
;
```

transform transformname[_transformclass]

This attribute indicates the beginning of a transform entry. This attribute is required and must be the first attribute in the entry.

transformname

The name of the transform, for example, **pcl2afp** or **ps2afp**.

transformclass

The name of an optional transform class. Specify from 1 to 63 characters, including letters, numbers, or special characters. The transform name is case-sensitive. To use the transform class, you must specify the transform class when you invoke the transform.

Default: No transform class. This transform entry is used when no transform class option (-c) is specified either in the printer definition or by the job submitter.

Example: transform pcl2afp_letter_300

To specify the **letter_300** transform class on the **pcl2afp** command, type:

```
pcl2afp -c letter_300
```

start-command = "daemon [option]... "

The name of the transform daemon and options. Enclose the value in single or double quotation marks if you specify an option. This attribute is required.

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 in the **aopstart** EXEC, specify the full directory path name of the daemon.

option

One or more options that the transform supports. For information about the transform options you can specify for each transform, see the documentation for the transform product.

environment = {name -> value [name -> value]... }

Environment variables that define the transform environment. Enclose the entire set of environment variables in braces.

For information about the environment variables you can specify for each transform, see the documentation for the transform product.

These 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 runtime environment.

Tip: 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. You can specify a job name of 1 to 8 alphanumeric characters. Incorrect job names are ignored. If you specify a different job name, you might need to list it in the Transaction Name Group (TNG) you defined for the z/OS Workload Manager (see “Setting Workload Manager goals” on page 112). For more information about the `_BPX_JOBNAME` variable, see *z/OS UNIX System Services Planning*.

maximum-active = *number*

The maximum number of transform daemons that the Infoprint Server Transform Manager activates concurrently. Specify a number greater than 0 and greater than or equal to the number specified in **minimum-active**. When this number is reached, the Infoprint Server Transform Manager does not start a new transform daemon to do the transform, and a print job waits until a transform daemon is available. This attribute is optional.

Guidelines:

- A value of 2 or 3 is suitable if you transform small print jobs and do not transform many print jobs at the same time. However, if you might need to transform large print jobs at the same time, 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.
- If you use the Print Interface subsystem in your installation, do not specify a low 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.

Default: No maximum number. Transform daemons are started when needed.

maximum-idle-time = *seconds*

The number of seconds before the Infoprint Server Transform Manager shuts down an idle transform daemon and system resources are freed. Specify a number 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.

Default: An idle transform daemon is not shut down.

minimum-active = *number*

The number of transform daemons that the Infoprint Server Transform Manager activates concurrently. Also the minimum number of transform daemons that the Infoprint Server Transform Manager keeps active (that is, not shut down) even when the maximum-idle-time expires for an idle

transform daemon. Specify a number less than or equal to the number specified in the **maximum-active** attribute. This attribute is optional.

Tip: Do not specify a very high number because active transform daemons use some system resources. However, starting transform daemons uses additional CPU resources, so you should keep some transform daemons active even when they are idle.

Default: minimum-active = 0

Example of a transform entry

This example shows one sample entry in the **aopxfd.conf** configuration file for the PCL to AFP transform.

When the Infoprint Server Transform Manager starts, it starts one transform daemon for the PCL to AFP transform. The Infoprint Server Transform Manager keeps no more than two transform daemons active at any time. The Infoprint Server Transform Manager shuts down any transform daemon that is idle and frees system resources after 5 minutes. However, the Infoprint Server Transform Manager keeps at least one transform daemon active.

Because the **transform** attribute does not specify a transform class, the Infoprint Server Transform Manager uses this entry when no transform class is specified as a transform command option or a filter option when the transform is invoked.

```
transform pcl2afp
  start-command = "pcl2afpd"
  environment = { _BPX_JOBNAME -> PCL2AFPD }
  minimum-active = 1
  maximum-active = 2
  maximum-idle-time = 300 # 5 minutes
;
```

Steps for creating the transform configuration file

These steps show how to create the **aopxfd.conf** configuration file in the **/etc/Printsrv** directory, which is the default directory for Infoprint Server configuration files.

To edit an existing **aopxfd.conf** file, follow the steps in “Steps for editing the transform configuration file” on page 164.

Before you begin: You must have permission to create files in the **/etc/Printsrv** directory. If you did not change the owner and permissions for **/etc/Printsrv** after installation, you must have an effective UID of 0 to create files in this directory.

1. On a z/OS UNIX command line, switch to an effective UID of 0:

```
su
```

To use the z/OS UNIX **su** command, you must be permitted to the BPX.SUPERUSER FACILITY class profile in RACF.

2. Copy the sample configuration file, **/usr/lpp/Printsrv/samples/aopxfd.conf**, to the **/etc/Printsrv** directory:

```
cp /usr/lpp/Printsrv/samples/aopxfd.conf /etc/Printsrv/aopxfd.conf
```

Tips:

- If you copy the configuration file to another directory, specify the full path name of the file in the AOPXFD_CONF environment variable in both the **aopstart** EXEC and the **/etc/profile** file. For information, see “Setting environment variables for Transform Interface” on page 167.
- You can use the sample job provided in SYS1.SAMPLIB(AOPCPETC) to copy all Infoprint Server sample configuration files, including the transform configuration file, from **/usr/lpp/Printsrv/samples** to the **/etc/Printsrv** directory. Review the comments in the sample job to determine which configuration files your installation requires, and then run the job to create the configuration files. A user with an effective UID of 0 should run this job.

3. Change the owner of the file to UID 0:

```
chown 0 /etc/Printsrv/aopxfd.conf
```

4. Change the group owner to AOPADMIN. AOPADMIN is the suggested RACF group name for Infoprint Server administrators. However, you might have used a different name for this group.

```
chgrp AOPADMIN /etc/Printsrv/aopxfd.conf
```

5. Change the permissions of the file so that it is readable by everyone and writeable only by UID 0 and members of the AOPADMIN group:

```
chmod 664 /etc/Printsrv/aopxfd.conf
```

6. Edit the file using your preferred editor and save the changes:

```
oedit /etc/Printsrv/aopxfd.conf
```

Edit this file using the IBM-1047 code page. For the syntax rules for editing the file, see “Syntax rules”.

7. If you switched to an effective UID of 0 in step 1, switch back to your own UID:

```
exit
```

8. Start the Infoprint Server Transform Manager. For example, enter this MVS command to run the AOPSTART JCL procedure:

```
START AOPSTART
```

Tip: Before running the AOPSTART procedure, make sure that you added the **xfd** value to the **start-daemons** attribute in the Infoprint Server configuration file. See “Editing the **aopd.conf** file for the Transform Manager” on page 158.

9. Check for error messages in the **stderr** file of each transform daemon. If you find any error messages, fix the errors and restart the Infoprint Server Transform Manager.

Each instance of a transform daemon writes messages to its own **stderr** file, which is cleared when the Infoprint Server Transform Manager restarts. To find the transform's **stderr** file, look in directory *base-directory/xfd*, where

base-directory is the directory specified in the **base-directory** attribute of the Infoprint Server configuration file, **aopd.conf**. The default base directory is **/var/Printsrv**.

The **stderr** file-naming convention is:

transform[_class].n.stderr

To read the **stderr** file, you must be a member of the RACF group for Infoprint Server administrators (AOPADMIN).

Syntax rules: When you edit this file, follow these syntax rules:

- Start comments with the # symbol
- End the last attribute in a transform entry with a semicolon.
- Code blank characters before and after equal signs if desired.
- Use lowercase characters for the attribute names. Use uppercase or lowercase characters for the attribute values.
- Environment variable names are case-sensitive. Typically, environment variable names are all uppercase.
- If a value contains blank characters or special characters (such as {}-> =, # &), enclose the value in single or double quotation marks.
- To continue a line that contains a character string enclosed in quotation marks, end the line with a back slash and do not type any characters, even blank characters, after the back slash. For example:

```
environment = {AOP_FONTLIB -> "sys1.font300 \
               sys1.sfntilib inst.font300"}
```

Steps for editing the transform configuration file

These steps show how to edit an existing **aopxfd.conf** configuration file. If you have not created the **aopxfd.conf** file yet, follow the steps in “Steps for creating the transform configuration file” on page 162.

Before you begin: You must be a member of the RACF group for Infoprint Server administrators (AOPADMIN) or have an effective UID of 0.

1. Edit the file using your preferred editor and save the changes:

```
oedit /etc/Printsrv/aopxfd.conf
```

Edit this file using the IBM-1047 code page. For the syntax rules for editing the file, see “Syntax rules”.

-
2. Stop and restart the Infoprint Server Transform Manager. For example, enter these MVS commands to run the AOPSTOP and AOPSTART JCL procedures.

```
START AOPSTOP,OPTIONS='-d xfd'
START AOPSTART
```

-
3. Check for error messages in the **stderr** file of each transform daemon. If you find any error messages, fix the errors and restart the Infoprint Server Transform Manager.

Each instance of a transform daemon writes messages to its own **stderr** file, which is cleared when the Infoprint Server Transform Manager restarts. To find the transform's **stderr** file, look in directory *base-directory/xfd*, where

base-directory is the directory specified in the **base-directory** attribute of the Infoprint Server configuration file, **aopd.conf**. The default base directory is **/var/Printsrv**.

The **stderr** file-naming convention is:

transform[_class].n.stderr

To read the **stderr** file, you must be a member of the RACF group for Infoprint Server administrators (AOPADMIN).

Syntax rules: When you edit this file, follow these syntax rules:

- Start comments with the # symbol.
- End the last attribute in a transform entry with a semicolon.
- Code blank characters before and after equal signs if desired.
- Use lowercase characters for the attribute names. Use uppercase or lowercase characters for the attribute values.
- Environment variable names are case-sensitive. Typically, environment variable names are all uppercase.
- If a value contains blank characters or special characters (such as {}-> =, # &), enclose the value in single or double quotation marks.
- To continue a line that contains a character string enclosed in quotation marks, end the line with a back slash and do not type any characters, even blank characters, after the back slash. For example:

```
environment = {AOP_FONTLIB -> "sys1.font300 \
               sys1.sfntilib inst.font300"}
```

Creating the remote transform configuration file (aoprxf.conf)

The remote transform configuration file contains information that Transform Interface uses to access Ricoh InfoPrint Transform Manager for Linux. If you use InfoPrint Transform Manager for Linux, this configuration file is required.

For information about InfoPrint Transform Manager for Linux, see:

- *InfoPrint Transform Manager for Linux*
- *InfoPrint Transform Manager for Linux: afp2pdf Transform Installation and User's Guide*

Format of the remote transform configuration file

The remote transform configuration file, **aoprxf.conf**, has this 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.

Tip: The InfoPrint Transform Manager for Linux user interface lists the IP address and port number of the transform server.

Default: None

Examples:

```
remote-transform-manager-ip-address = linux1.xyz.com
remote-transform-manager-ip-address = 99.99.123.234
```

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

Example: remote-transform-manager-port-number = 6987

Steps for creating the remote transform configuration file

These steps show how to create the **aoprxf.conf** configuration file in the **/etc/Printsrv** directory, which is the default directory for Infoprint Server configuration files.

Before you begin: You must have permission to create files in the **/etc/Printsrv** directory. If you did not change the owner and permissions for **/etc/Printsrv** after installation, you must have an effective UID of 0 to create files in this directory.

1. Switch to an effective UID of 0:

```
su
```

To use the z/OS UNIX **su** command, you must be permitted to the BPX.SUPERUSER FACILITY class profile in RACF.

-
2. Copy the sample configuration file, **/usr/lpp/Printsrv/samples/aoprxf.conf**, to the **/etc/Printsrv** directory.

```
cp /usr/lpp/Printsrv/samples/aoprxf.conf /etc/Printsrv/aoprxf.conf
```

Tips:

- If you copy the configuration file to another location, specify the full path name of the file in the AOPRXF_CONF environment variable in both the **aopstart** EXEC and the **/etc/profile** file. For information, see “Setting environment variables for Transform Interface” on page 167.
- You can use the sample job provided in SYS1.SAMPLIB(AOPCPETC) to copy all Infoprint Server sample configuration files, including the remote transform configuration file, from **/usr/lpp/Printsrv/samples** to the **/etc/Printsrv** directory. Review the comments in the sample job to determine which configuration files your installation requires, and then run the job to create the configuration files. A user with an effective UID of 0 should run this job.

-
3. Change the owner of the file to UID 0:

```
chown 0 /etc/Printsrv/aoprxf.conf
```

-
4. Change the group owner to AOPADMIN. AOPADMIN is the suggested RACF group name for Infoprint Server administrators. However, you might have used a different name for this group.

```
chgrp AOPADMIN /etc/Printsrv/aoprxf.conf
```

-
5. Change the permissions of the file so that it is readable by everyone and writeable only by UID 0 and members of the AOPADMIN group:

```
chmod 664 /etc/Printsrv/aoprxf.conf
```

6. Edit the file using your preferred editor and save the changes.

```
oedit /etc/Printsrv/aoprxf.conf
```

Edit this file using the IBM-1047 code page. For the syntax rules for editing the file, see “Syntax rules”.

7. If you switched to an effective UID of 0 in step 1, switch back to your own UID:

```
exit
```

Tip: Infoprint Server picks up the changes to this file automatically. You do not need to restart Infoprint Server after you create or edit this file.

Syntax rules: When you edit this file, follow these syntax rules:

- Use lowercase characters for the attribute names. Use uppercase or lowercase characters for the values.
- Use blank characters before or after the equal sign, if desired.
- Start comments with the # symbol.
- Include blank lines, if desired.

Setting environment variables for Transform Interface

These environment variables affect the behavior of Transform Interface. Set these environment variables only if you do not create the transform configuration files in the default locations.

If you set these environment variables, set them in both of these locations:

- **aopstart EXEC:** Transform Interface uses environment variables specified in this file. For information about how to edit this file, see “Editing the aopstart EXEC” on page 57.
- **/etc/profile file:** z/OS UNIX transform commands, such as **ps2afp** and **remotexf**, use environment variables specified in this file.

Environment variable	Meaning
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	The full path name of the transform configuration file, aopxfd.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/aopxfd.conf

Transform Interface also uses the LANG, LIBPATH, MANPATH, NLSPATH, and PATH environment variables.

Examples: To specify a non-default location for the configuration files:

- Change these lines in the **aopstart** EXEC:


```
n=n+1; env.n='AOPRXF_CONF=/etc/Printsrv/aoprxf.conf'
n=n+1; env.n='AOPXFD_CONF=/etc/Printsrv/aopxfd.conf'
```

Tip: After you edit the **aopstart** EXEC, stop and restart all Infoprint Server daemons. For example, enter these MVS commands to run the AOPSTOP and AOPSTART JCL procedures:

```
START AOPSTOP
START AOPSTART
```

- Add these statements to the **/etc/profile** file:


```
export AOPRXF_CONF=/directory/aoprxf.conf
export AOPXFD_CONF=/directory/aopxfd.conf
```

directory is the absolute directory path that contains the configuration file.

Customizing the TCP/IP network firewall

If you have configured a TCP/IP firewall, make sure that the z/OS system is allowed to communicate using the TCP/IP protocol with transforms that run on remote systems.

If you use InfoPrint Transform Manager for Linux to transform data, the z/OS system must be able to connect to each InfoPrint Transform Manager transform server that the administrator has defined. The z/OS system must be able to connect to each transform server at both these ports:

- The transform server's defined port. The default port is 6986.
You specify the defined port in the **remote-transform-manager-port-number** attribute in the remote transform configuration file, **aoprxf.conf**.
- Port 6984, which is the port to which Infoprint Server sends data.

Tip: The InfoPrint Transform Manager for Linux user interface lists the IP addresses and defined port numbers of all the transform servers.

If you use Ricoh InfoPrint Manager for AIX or Windows to transform data, the z/OS system must be able to connect to InfoPrint Manager at the ports where the transform daemons are running. The administrator specifies the port for the transform daemon in the **-P** option of the **aoprxf.dll** transform filter in the printer definition.

Setting up security for transform resources

The user ID that starts Infoprint Server must have permission to read any transform resource files that reside on the z/OS system. For example, the AFP to PCL, AFP to PDF, and AFP to PostScript transforms use AFP resources, such as form definitions and fonts, that reside on the z/OS system. And, the SAP to AFP transform uses configuration files that reside on the z/OS system.

For information about how to set up security for transform resources, see:

- *IBM Print Transforms from AFP for Infoprint Server for z/OS*
- *IBM Infoprint Transforms to AFP for z/OS*

Transforms that run on non-z/OS systems might use transform resources that reside on the z/OS system. For example, InfoPrint Transform Manager for Linux transforms use a PostScript halftone dictionary, which can reside on the z/OS

system. If a resource resides on the z/OS system, set the z/OS UNIX permissions for the file so that the user ID that starts Infoprint Server can read it.

Creating a code page conversion table

Transform Interface converts messages from InfoPrint Transform Manager for Linux from the ISO8859-1 code page to the code page for the z/OS locale. Therefore, if you use a custom code page for the z/OS locale, make sure that a conversion table exists. For information about how to create conversion tables, see information about code set conversion utilities in *z/OS XL C/C++ Programming Guide*.

Chapter 6. Customizing NetSpool

This information describes the NetSpool component of Infoprint Server. You must customize NetSpool if you want to print from VTAM applications, such as CICS and IMS.

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 creates output data sets on the JES2 or JES3 spool. For an overview of NetSpool and how it fits into your system, see “NetSpool” on page 12.

Table 28 summarizes the tasks to customize NetSpool. Required tasks are required by all installations that want to use NetSpool. Optional tasks are required only if the listed condition applies to your installation.

Table 28. Customization tasks for NetSpool

Task	Condition	See page
Identifying software requirements for NetSpool	Required	172
Updating the BPXPRMxx member of SYS1.PARMLIB	Required	172
Editing the aopd.conf file for NetSpool	Required	172
Customizing the NetSpool message log	Optional: To record messages in the NetSpool log	174
Customizing the NetSpool startup procedure	Required	174
Working with JES spool limits	Required	180
Changing Language Environment runtime options	Optional: To change default options	181
Writing NetSpool exit and filter programs	Optional: To customize how NetSpool processes SCS and 3270 data streams	181
Setting environment variables for NetSpool	Optional: To print line data that contains unprintable characters	194
Editing the system configuration definition for NetSpool	Optional: To specify dynamic attributes	195

Table 29 lists the other components of Infoprint Server that you might need to customize when you use NetSpool:

Table 29. Other components to customize for NetSpool

Task	Condition	See page
Customizing the Printer Inventory Manager	Required	27
Customizing the Transform Interface ¹	Optional: To transform data from one format to another	157

Table 29. Other components to customize for NetSpool (continued)

Task	Condition	See page
Customizing IP PrintWay	Optional: To print data to remote printers in your TCP/IP network or send data to email destinations	199
1. This task is not required to transform SCS or 3270 data to line data or PCL data. However, it is required to transform line data or PCL data to another format such as to AFP or PDF format		

Related tasks: After you customize NetSpool, the administrator and operator must do these tasks. For information about these tasks, see *z/OS Infoprint Server Operation and Administration*:

- The administrator must create or update the printer definition for each printer in the Printer Inventory and specify printer attributes required by NetSpool.
- The administrator must define each NetSpool printer logical unit (LU) to VTAM, and modify VTAM resource definitions if necessary.
- The operator must start NetSpool and vary the NetSpool printer LUs active.

Identifying software requirements for NetSpool

No additional software is required.

Updating the BPXPRMxx member of SYS1.PARMLIB

When you use NetSpool, 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:

Parameter	Meaning
MAXPROCUSER	The maximum number of processes that a single z/OS UNIX user ID can have concurrently active. Guideline: Add at least 70 for Infoprint Server processes.
MAXPROCSYS	The maximum number of processes that the z/OS system allows. Guideline: Add 200 for processes used by both NetSpool and Print Interface. If you also run IP PrintWay, add an additional 50 (extended mode) or an additional 70 for each IP PrintWay FSS (basic mode).

Example: Specify these statements in the BPXPRMxx member:

```
MAXPROCUSER(70) /*70(Infoprint Server) */
MAXPROCSYS(450) /*200(Print Interface/NetSpool)+50(PrintWay extended)+200(initial)*/
```

Tip: You can use the SETOMVS or SET OMVS command to dynamically increase the MAXPROCUSER and MAXPROCSYS values. However, to make a permanent change, edit the BPXPRMxx member that is used for IPLs. For more information about the BPXPRMxx member, see *z/OS MVS Initialization and Tuning Reference*.

Editing the aopd.conf file for NetSpool

The **aopd.conf** file contains attributes that Infoprint Server uses when it starts. The default location of this file is **/etc/Printsrv/aopd.conf**.

To start the NetSpool daemon, you must specify the following attribute in the **aopd.conf** file:

start-daemons = {netd}

To start the NetSpool daemon, add **netd** to the values in this statement. Enclose the values in braces. This statement is required.

Rule: If you add **netd** while Infoprint Server is running, start Infoprint Server again. You do not need to stop any Infoprint Server daemons before restarting.

Default: start-daemons = {lpd}

Example: In this example, the Printer Inventory Manager daemons, the LPD, and the NetSpool daemon start:

```
start-daemons = {lpd netd}
```

If dynamic configuration is not enabled, you can also specify this attribute in the **aopd.conf** file:

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.

The prefix must be two alphanumeric or national (@, \$, and #) 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 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.
- 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.

Tips:

- z/OS also assigns a unique z/OS job ID to each output data set. The z/OS job ID might be different from the job ID of the output data set on the JES spool. JES operator commands use this z/OS job ID.
- The job ID that Print Interface returns to the job submitter does not contain this prefix.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: job-prefix = PS

Example: job-prefix = 'P#'

Related information: For information about:

- How to edit the **aopd.conf** file, see “Steps for editing the Infoprint Server configuration file (aopd.conf)” on page 42.
- All attributes in the **aopd.conf** file, see Appendix B, “Infoprint Server files,” on page 389.
- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 68.

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 and most transforms in one place.
- 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 consider using the common message log to see all 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.

The NetSpool message-log data set requires these specifications:

- Sequential organization (PS).
- Fixed block format (FBA).
- Logical record length (LRECL) of 57 at a minimum. 80 is suggested.
- Block size (BLKSIZE) of about 10 records per block. 800 is suggested.

Secondary space is unnecessary, because the ANFMFILE program formats all available space. To change the size of the NetSpool message-log data set, rerun the ANFMFILE program.

Tip: The ANFMFILE program produces a D37 abend when it completes processing.

Customizing the NetSpool startup procedure

Before starting NetSpool, you must create a startup procedure. The startup procedure identifies:

- Program name, region size, time-out value, classes of printer LUs, tracing option, console name, and the name of the Printer Inventory
- The location of the NetSpool message-log data set
- The location of the trace options file
- The location of exit programs

This table summarizes the customization tasks related to setting up the startup procedure. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies to your installation.

Task	Condition	See page
Determining the number of NetSpool tasks to start	Required	175
Using the Japanese version	Optional: To receive messages in Japanese	175
Using an extended MCS console	Optional: To receive unsolicited messages at an extended console	176
Creating a user ID to assign to the NetSpool startup procedure	Required	176
Defining the NetSpool startup procedure to RACF	Required	176
Editing the NetSpool startup procedure	Optional: To change default settings	177

Determining the number of NetSpool tasks to start

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

Tip: If you start more than one NetSpool task, you cannot use Infoprint Central to work with NetSpool LUs.

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.

Using the Japanese version

A Japanese version is available for the NetSpool task. If you ordered the Japanese version, a new message data set is provided in a separate FMID. To install the message data set associated with the language feature, see *z/OS Program Directory*.

To receive messages in Japanese, specify the LANGUAGE parameter in the EXEC statement, using the last three characters in the name of the message data set. For more information about the LANGUAGE parameter, see “Editing the NetSpool startup procedure” on page 177.

If the language feature is installed, you can still use the English message data set by either removing the LANGUAGE parameter from the EXEC statement or by specifying LANGUAGE=ENU and starting the NetSpool task again.

To receive messages from the NetSpool daemon in Japanese, specify the LANG=Ja_JP environment variable. See “Setting environment variables for the Printer Inventory Manager” on page 43.

Be certain your console supports the language you install. NetSpool sends all messages to the console, an optional NetSpool message-log data set, and the optional common message log. Some NetSpool messages are sent before NetSpool loads the message data set. These messages are sent in English.

You must configure an extended MCS console on TSO/E to view messages in Japanese. For information about customizing TSO/E, see *z/OS TSO/E Customization*. Messages sent by NetSpool are already translated into Japanese.

Using an extended MCS console

When using extended MCS consoles, messages sent in response to NetSpool operator commands are routed to the console that issued the command. The NetSpool started task sends unsolicited messages to an extended MCS console when the CONSNAME parameter in the startup procedure specifies the name of an extended MCS console.

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 access to z/OS UNIX.

Steps for creating a RACF user ID with access to z/OS UNIX

1. Create a user ID with an OMVS segment 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. You can specify any home directory or use the default home directory.

Example: This RACF command defines user ID AOPSTC with a default group of AOPOPER and home directory `/tmp`. 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) PROGRAM('/bin/sh')) HOME (/tmp)
DFLTGRP(AOPOPER) NOPASSWORD
```

2. Give the user ID RACF update access to the NetSpool message-log data set.

Example: This RACF command gives user ID AOPSTC update access to data set USER.MSGFILE:

```
PERMIT USER.MSGFILE ACCESS(UPDATE) ID(AOPSTC)
```

Defining the NetSpool startup procedure to RACF

You must define a profile for the NetSpool startup procedure in the RACF STARTED class.

Example: These RACF commands define a profile for the APIJPJCL procedure. This 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 APIJPJCL.* STDATA(USER(AOPSTC) GROUP(AOPOPER))
SETROPTS RACLIST(STARTED) REFRESH
```

Tip: If you want to use the started procedure table (ICHRIN03) instead, see *z/OS Security Server RACF Security Administrator's Guide* for more information.

Editing the NetSpool startup procedure

Infoprint Server provides procedure APIJPJCL, which you can use to start NetSpool. You can modify this procedure for your installation.

APIJPJCL is shipped in SYS1.IBM.PROCLIB. However, during install APIJPJCL might have been copied to another data set in the STC PROCLIB concatenation.

Steps for editing the NetSpool startup procedure

1. Copy the APIJPJCL to a procedure library that is known to JES. For information about procedure libraries, see *z/OS JES2 Initialization and Tuning Guide* and *z/OS JES3 Initialization and Tuning Guide*.

2. Edit the APIJPJCL procedure if any of these conditions apply to your installation:

- If you did not add the Language Environment runtime libraries (*hlq.SCEERUN* and *hlq.SCEERUN2*) and the C++ runtime library (*hlq.SCLBDLL*) to the system LNKLST, add them to the STEPLIB DD statement.
- If you use a NetSpool exit and the library that contains the exit is not in the standard z/OS 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.
- If you want to receive NetSpool unsolicited messages on an alternate console, specify the CONSNAME parameter in the EXEC statement.

This procedure uses symbolic parameters with default values for some of the parameters that you might want to change for your installation. For more information about symbolic parameters, see *z/OS MVS JCL Reference*.

3. If a NetSpool task is running, restart it to pick up the changes. For information about how to stop and start NetSpool, see *z/OS Infoprint Server Operation and Administration*.

Related information: For information about how to access the Language Environment runtime libraries and improve the performance of runtime libraries, see *z/OS UNIX System Services Planning*.

APIJPJCL procedure

Figure 18 shows the NetSpool startup procedure.

```
//NETSPOOL PROC LUCLASS=(1),INV='AOP1',DURATION=NOLIMIT,  
//          SIZE=31M,MSGFILE=USER.MSGFILE  
//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 18. NetSpool startup procedure — *SYS1.IBM.PROCLIB(APIJPJCL)*

The JCL statements and parameters that you can include in your NetSpool startup procedure are:

label PROC

Marks the beginning of the procedure. The sample PROC statement in Figure 18 specifies default values for symbolic parameters coded in the sample. If you plan to use the sample procedure, modify these default values to suit your installation.

**label EXEC PGM=APIPPAAA,REGION=*size*,TIME=NOLIMIT,
PARM='LUCLASS=(*class*[,...]),INV=*inventory*[,LANGUAGE=*language_id*] [
,CONSNAM=*console_name*] [,TRACE=ON]'**

Invokes the NetSpool program.

PGM=APIPPAAA

The NetSpool program entry point.

REGION=*size*

The maximum amount of virtual storage the NetSpool task can use. For more information about specifying region size, see *z/OS MVS JCL Reference*.

TIME=NOLIMIT

Specify TIME=1440 or TIME=NOLIMIT to prevent TIMEOUT abends.

PARM=

Specify a maximum of 100 characters for the value of the PARM parameter, as described in *z/OS MVS JCL Reference*.

LUCLASS=(*class*[,...])

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

If you want to start a NetSpool task without starting any printer LUs, specify a class number, from 1 to 64, that does not match any classes defined for printer LUs in the Printer Inventory. To start individual printers, use Infoprint Central or the NetSpool LUNAME ADD command described in *z/OS Infoprint Server Operation and Administration*.

This parameter is required.

INV=inventory

The name that you assigned to the Printer Inventory in the **inventory** attribute in the **aopd.conf** configuration file. Specify AOP1 (the default name) if you did not create an **aopd.conf** configuration file or if the **inventory** attribute is omitted. This name is case-sensitive. This parameter is required.

CONSNAME=console_name

The name of an MCS or extended MCS console where you want NetSpool to display its unsolicited messages. Unsolicited messages are sent by NetSpool as a result of an unexpected error, such as a VTAM or data error. For *console_name*, specify a name that is defined in a CONSOLE statement in the CONSOLxx member of SYS1.PARMLIB. You can abbreviate the CONSNAME parameter: **CON=console_name**.

Translated NetSpool messages can be displayed correctly when CONSNAME specifies the name of an extended MCS console that is configured for the language of the translated messages.

If the CONSNAME parameter is not specified, unsolicited NetSpool messages are displayed at the console that used the START NetSpool command to start NetSpool.

LANGUAGE=language_id

The language that this NetSpool task is to use for messages.

Tip: You can abbreviate this parameter: **LNG=language_id**

NetSpool appends the *language_id* to the prefix APIMT to determine the name of the message data set to load. If you specify Japanese, first install the Japanese version. If you omit the LANGUAGE parameter, or if NetSpool does not find the specified message data set, NetSpool sends messages in English. (NetSpool uses the default message data set APIMTENU.)

Example:

```
LNG=JPN
```

TRACE=ON

Specifies that NetSpool is to read the data set in the TRACEOPT DD statement to obtain the trace options. The tracing will start when NetSpool is started.

STEPLIB DD DSN=dsname,DISP=SHR

The name of a data set that contains NetSpool exit programs written by your installation. The library must be APF authorized. This statement is optional. Specify it if you are replacing any of the IBM-supplied exit programs.

Also, concatenate the Language Environment libraries (*hlq.SCEERUN* and *hlq.CEERUN2*) and the C++ runtime library (*hlq.SCLBDLL*) if you did not add them to the system LNKLIST.

APIMMSG DD DSN=dsname,DISP=SHR

The name of the NetSpool message-log data set. This statement is optional. If

you include this statement, this NetSpool task writes messages to the message-log data set as well as to the operator console. If you specify a language identifier other than ENU in the LANGUAGE parameter on the NetSpool startup procedure, the message-log data set contains the translated messages. For more information, see “Customizing the NetSpool message log” on page 174.

TRACEOPT DD DSN=*dsname*,DISP=SHR

The name of the trace options data set. Specify DISP=SHR to allow more than one NetSpool started task to use the data set. The trace options data set can be either a member of a PDS or a sequential data set.

Related information: For information about how to access the Language Environment runtime libraries and improve the performance of runtime libraries, see *z/OS UNIX System Services Planning*.

Working with JES spool limits

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 (**aopippdpx**)
- 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 JES2 and JES3 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, JES2 and JES3 display a warning message on the system console and let 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.

Tip: 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.

Related task: Your administrator can limit the amount of data that Infoprint Server writes to the JES spool for any one print job. For information, see the description of the **Maximum document size** field or **maximum-document-size** attribute in *z/OS Infoprint Server Operation and Administration*.

Changing Language Environment runtime options

Figure 19 shows the runtime options for the Language environment that NetSpool establishes.

```
TRAP(OFF)
POSIX(ON)
ALL31(ON)
ENVAR("_CEE_ENVFILE=//DD:STDENV")
RPTOPTS(ON)
RPTSTG(ON)
MSGFILE(,,,ENQ)
STACK(8K,64K,ANY,KEEP)
HEAP(8K,8K,.,FREE)
ANYHEAP(8K,8K)
THREADHEAP(8,4K,ANY,FREE)
BELOWHEAP(8,4K,FREE)
LIBSTACK(8,0)
```

Figure 19. Language Environment runtime options for NetSpool

To change any of the Language Environment runtime options:

1. Create a CEEUOPT object file. You can use sample job CEE.SCEESAMP(CEEWUOPT) to create the CEEUOPT.
2. Create an SMPE usermod to link-edit the CEEUOPT object file with NetSpool module SYS1.LINKLIB(APIIOPN). Place the CEEUOPT text deck first when you link-edit it with module APIIOPN.
3. Put the new module into a link library and add that library to the NetSpool startup procedure with a STEPLIB or JOBLIB DD statement.

For more information, see:

- For information about Language Environment runtime options, see *z/OS Language Environment Programming Guide*.
- For information about how to create a CEEUOPT file and link-edit it with another module, see *z/OS Language Environment Customization*.

Writing NetSpool exit and filter programs

Note

This section contains programming-interface information.

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.

If the NetSpool exits do not meet your needs, you can write a filter program. A filter program can inspect the line or PCL output data stream and modify the data stream, add separator pages, or specify job attributes (DLL filter only), such as the **copies** or the **mail-to-addresses** job attribute. An application programmer can also code job attributes in the input data stream, or an administrator can specify job attributes in the printer definition.

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.

Tip: You can specify only one filter for each data format in the printer definition. Therefore, if you write your own filter, you cannot also specify a transform in the printer definition.

Related information:

- For information about how to write filter programs, see “Writing a filter program” on page 121.
- For information about how to specify job attributes in the application, see *z/OS Infoprint Server User's Guide*.
- For information about how to specify job attributes in the printer definition (in the Allocation section), and how to specify filters in the printer definition, see *z/OS Infoprint Server Operation and Administration*.

Programming considerations

Consider these points when coding exit programs:

- You should be an experienced programmer and aware of overall system implications.
- Do not do an explicit or implied WAIT in an exit because it puts NetSpool into a WAIT state.
- The exits can run in either 24-bit or 31-bit addressing mode. IBM suggests 31-bit addressing to allow more printer LUs to run in an address space.
- Use only the control block passed to the exit. Code references to fields in the control block using the names associated with each field in the DSECT control block. Do not use absolute displacements, because these displacements can change from one release to another.
- The exit programs run in supervisor state in key 8.
- The exit programs must be non-reentrant, which is the default value for the IBM assembler and binder.
- An ESPIE is in effect while the exit programs are running. Programming exceptions cause the printer session and the application session to terminate.

Guideline: Future NetSpool program maintenance might require that you reassemble these exits.

APIPP1X2 control block

When the exits are called, register 1 contains the address of the APIPP1X2 control block, mapped by DSECT S2STG, shown in Figure 20 on page 184.

Offset	(Hex)	Type	Length	Name	Description
0	(0)	Structure	*	S2STG	Exit parameter area
0	(0)	Character	5	S2EYEC	Set to 'S2STG'
5	(5)		1	*	Reserved
=====					
The following flags are input fields to all exits. The exits must not change these fields.					
=====					
5	(5)	Unsigned	1	S2FLAGS	Flags for LU type and formatting option
		...1		S2LU1	Set for LU1; not set for LU0 and LU3
	1		S2FPCL	Set when NetSpool converts to PCL
=====					
The following fields are input to APIPPTD1 and APIPPTD2. The exits must not change these fields.					
=====					
Horizontal Page Format:					
6	(6)	Signed	2	S2COL	Next column number in output
8	(8)	Signed	2	S2LM	Left margin column number
10	(0A)	Signed	2	S2RM	Right margin column number
12	(0C)	Signed	2	S2MPP	Maximum presentation position (line length)
Vertical Page Format:					
14	(0E)	Signed	2	S2LINE	Next output line number
16	(10)	Signed	2	S2TM	Top margin line number (channel 1)
18	(12)	Signed	22	S2C2T12	Channel 2-12 line numbers, 2 bytes per channel
40	(28)	Signed	2	S2BM	Bottom margin line number
42	(2A)	Signed	2	S2MPL	Maximum page length
=====					
The following fields are input to all exits. The exits must not change these fields.					
=====					
44	(2C)	Character	8	S2PLUNAM	LU name of the application generating the print request - primary LU (PLU)
52	(34)	Character	8	S2LPTNAM	Name of the printer LU - secondary LU (SLU)
60	(3C)	Character	8	S2PRTCLS	Classes of the printer LU.
					Each bit represents a class, from 1 to 64:
		1...		Value	Class
		.1..		1	Class 1
	1		1	Class 2
				1	Class 8
					This pattern repeats for 7 more bytes.
					More than one bit can be set at a time.
=====					
The following field can be set by exit APIPPTD1.					
=====					
68	(44)	Character	1	S2DBCSM	Number of bytes to allocate per DBCS character:
				Value	Meaning
				0	Two positions per character (default)
				1	One position per character
=====					
The following field can be set by exit APIUBF3.					
=====					
69	(45)	Unsigned	1	S2OUTFLG	Flags for output
		1...		S2VLN255	Maximum 3270 variable line length is 255 characters (default is 132)
=====					
The following fields can be set by exits APIPPTD1 and APIPPTD2.					
=====					
70	(46)	Signed	2	S2NLNUM	New line number for output
72	(48)	Signed	2	S2NCNUM	New column number for output

Figure 20. Format of APIPP1X2 control block (Part 1 of 2)


```

=====
The following field is input to exits APIPPTD2 and APIUGEX. The exits must not change this field.
=====
  76  (4C) Character   275  S2TPDATA  APIPPTD2: Transparent data found in input data:
                                1 byte: X'35'
                                1 byte: Length of data
                                * bytes: Transparent data
                                APIUGEX: Graphic Escape character in EBCDIC:
                                1 byte: In EBCDIC
                                1 byte: In printer code page
 351  (15F)           5   *           Reserved
=====
The following fields can be set by all exits.
=====
  356  (164) Signed    4   S2WRKLEN  Length of data in S2WRKARA
  360  (168) Character 275  S2WRKARA  Data returned by exit to include in output
                                APIPPTD1: SCS controls or transparent data
                                APIPPTD2: Modified transparent data and SCS controls
                                APIUBF3: Line data or PCL data to add
                                APIUGEX: Line data or PCL data to replace the Graphic
                                Escape character sequence X'08XX'
=====

```

Figure 20. Format of APIPP1X2 control block (Part 2 of 2)

Beginning of File exit for SCS data streams (APIPPTD1)

The Beginning of File exit for SCS data streams (APIPPTD1) receives control when NetSpool starts a new output data set for SCS input data streams. You can use this exit to add up to 275 bytes of data to the start of a data set or to close the VTAM session with the printer.

NetSpool calls the same exit for all LU type 1 sessions. However, your exit can do different functions based on these input fields in the APIPP1X2 control block:

- S2FLAGS: Type of output data stream NetSpool creates for this printer— a line or PCL data stream
- S2LPTNAM: NetSpool printer name, also known as the secondary LU (SLU) name
- S2PLUNAM: LU name of the VTAM application, also known as the primary LU (PLU) name
- S2PRTCLS: LU classes to which the NetSpool printer belongs

You can use this exit to add SCS controls, printable text, and PCL commands. Follow these guidelines when you add data:

- You can add any SCS controls that NetSpool supports. For more information about using LU classes, see *z/OS Infoprint Server User's Guide*.
- If IP PrintWay controls the printer, you can add text data and any PCL commands that your printer supports as transparent data. Code text data in ASCII representation, and precede text and PCL commands with an SCS Transparent (TRN) control — X'35' followed by a 1-byte length field. Either NetSpool or IP PrintWay removes the SCS TRN control before transmitting the text and PCL commands, unchanged, to the printer.
- If you add transparent data in a TRN control, you might need to adjust the column and line number of the next print position. By default, when NetSpool finds a TRN control, NetSpool increases the column and line number at which the next character prints by the length of the transparent data, as specified in the TRN control. In your exit, you can change the column and line number where

the next character after the transparent data prints. To assist in calculating the desired line and column numbers, NetSpool provides the exit with the column and line number of the next print position before any transparent data is added.

- Any PCL commands that you add in this exit override any PCL commands specified in the IP PrintWay **Document header** field in the printer definition. They also override any PCL commands that NetSpool automatically generates as a result of PCL conversion values specified in the printer definition, such as the print density or line density.

Input to APIPPTD1

At input to the Beginning of File exit (APIPPTD1), the contents of the registers are:

- Register 1** Address of the APIPP1X2 control block
- Register 13** Address of an 18-word save area for saving the caller's registers
- Register 14** Return address

The APIPP1X2 control block is described in Figure 20 on page 184. These fields are valid at entry:

S2FLAGS	Type of conversion and VTAM session:
	S2LU1 Set for VTAM LU1 session. Not set for VTAM LU0 or LU3 sessions.
	S2PCL Set when NetSpool will convert the input data stream to a PCL data stream.
S2COL	The column number of the next print position
S2LM	Left margin column number
S2RM	Right margin column number
S2MPP	Maximum presentation position (line length)
S2LINE	The line number of the next print position
S2TM	Top margin line number (channel 1)
S2C2T12	Channel 2-12 line numbers
S2BM	Bottom margin line number
S2MPL	Maximum page length
S2PLUNAM	LU name of the application generating the print request (PLU).
S2LPTNAM	Name of the printer LU that NetSpool is processing (SLU).
S2PRTCLS	Logical unit classes to which the printer LU belongs. Each bit in this field represents a class, with the high order-bit of the leftmost byte representing class 1. For a description of the bits, see Figure 20 on page 184.

Output from APIPPTD1

At return from the Beginning of File exit (APIPPTD1), restore the contents of all registers, and place one of these codes in register 15 to tell NetSpool what action to take:

Code Meaning

- | | |
|---|--|
| 1 | Add the data from S2WRKARA to the output data set. |
| 2 | Do <i>not</i> add data from S2WRKARA to the output data set. |
| 4 | Close the VTAM session for this printer LU. |
| 8 | Do <i>not</i> call exit APIPTD2 for any transparent data found in the data stream for this output data set. Do <i>not</i> add data from S2WRKARA to the output data set. |
| 9 | Do <i>not</i> call exit APIPTD2 for any transparent data found in the data stream for this output data set. Add the data from S2WRKARA to the output data set. |

Also, set these fields in the APIPP1X2 control block:

S2DBCSM	The number of character positions NetSpool is to allocate in an output line for each double-byte character.
Value Meaning	
0 Allocate two print positions for each DBCS character. This is the default value set by NetSpool.	
1 Allocate only one print position for each DBCS character.	
S2NCNUM	The column number of the next print position after the transparent data. If this field is zero, NetSpool determines the column number.
S2NLNUM	The line number of the next print position after the transparent data. If this field is zero, NetSpool determines the line number.
S2WRKLEN	The length of the data in the work area, S2WRKARA.
S2WRKARA	The transparent-data control (TRN) and data to be added to the data stream. You can also add other SCS controls that NetSpool supports. Valid controls are described in <i>z/OS Infoprint Server User's Guide</i> .

Examples

1. To add no data to the beginning of the data set, set return code 2 or 8 in register 15.
2. To add data to the beginning of the data set:
 - Put the data in the S2WRKARA field.
 - Put the length of the data in S2WRKARA into the S2WRKLEN field.
 - Set return code 1 or 9 in register 15.

Beginning of File exit for 3270 data streams (APIUBF3)

The Beginning of File exit for 3270 data streams (APIUBF3) receives control when NetSpool starts a new output data set for 3270 input data streams. You can use this exit to do these functions:

- Add up to 275 bytes of printable text and PCL commands to the start of a data set.
- Print data that is wider than 132 characters (up to 255 characters) on the same line.

NetSpool calls the same exit for all LU type 0 and LU type 3 sessions. However, your exit can do different functions based on these input fields in the APIPP1X2 control block:

- **S2FLAGS**: Type of output data stream NetSpool creates for this printer— a line or PCL data stream
- **S2LPTNAM**: NetSpool printer name, also known as the secondary LU (SLU) name
- **S2PLUNAM**: LU name of the VTAM application, also known as the primary LU (PLU) name
- **S2PRTCLS**: LU classes to which the NetSpool printer belongs

If your exit adds printable text and PCL commands, follow these guidelines:

- If NetSpool converts the input data stream to a line data stream:
 - If the printer accepts EBCDIC line data (such as an AFP printer that PSF controls), you can add text. Code the text data in EBCDIC representation, and precede the data with a carriage control character.
 - If IP PrintWay controls the printer, you can add text and any PCL commands that your printer supports. Code text data in ASCII representation, and precede the text and PCL commands with a X'35' character followed by a 1-byte length field. IP PrintWay removes the X'35' and length field before transmitting the text and PCL commands, without change, to the printer.
- If NetSpool converts the input data stream to a PCL data stream:
 - You can add text and any PCL commands that your printer supports. Code text data in ASCII representation. NetSpool and IP PrintWay transmit the text and PCL commands to the printer without change.
 - If you add text, be sure to add a line termination control, such as a Carriage Return/New Line control sequence or a Form Feed control, after the text data to make sure that subsequent text starts either on a new line or page.
- Any PCL commands that you add in this exit override PCL commands specified in the IP PrintWay **Document header** field in the printer definition. They also override any PCL commands that NetSpool automatically generates as a result of PCL conversion values specified in the printer definition, such as print density or line density values.

Input to APIUBF3

At input to the Beginning of File exit (APIPPTD1), the contents of the registers are:

- Register 1** Address of the APIPP1X2 control block
- Register 13** Address of an 18-word save area for saving the caller's registers
- Register 14** Return address

The APIPP1X2 control block is described in Figure 20 on page 184. These fields are valid at entry:

- S2FLAGS** Type of conversion and VTAM session:
 - S2LU1** Set for VTAM LU1 session. Not set for VTAM LU0 or LU3 sessions.
 - S2PCL** Set when NetSpool will convert the input data stream to a PCL data stream.
- S2PLUNAM** LU name of the application generating the print request — primary LU (PLU).

- S2LPTNAM** Name of the printer LU that NetSpool is processing — secondary LU (SLU).
- S2PRTCLS** Logical unit classes to which the printer LU belongs. Each bit in this field represents a class, with the high order-bit of the leftmost byte representing class 1. For a description of the bits, see Figure 20 on page 184.

Output from APIUBF3

At return from the Beginning of File exit (APIUBF3), restore the contents of all registers, and place one of these return codes in register 15 to tell NetSpool what action to take:

Code Meaning

- 1 Add the data from S2WRKARA to the output data set.
- 2 Do *not* add data from S2WRKARA to the output data set.

You can set these fields in the APIPP1X2 control block:

- S2WRKLEN** Length of work area S2WRKARA.
- S2WRKARA** Data and PCL commands to be added to the output data stream.
- S2OUTFLG** Flags to customize the line data and PCL output data streams that NetSpool creates:

S2VLN255

Set this flag if you want to print data that is wider than 132 characters (up to 255 characters) on the same line, without wrapping characters to the next line. If this flag is set, NetSpool sets the maximum line length with variable length line format to 255 characters (not including carriage control). If this flag is not set, NetSpool sets the maximum line length to 132 characters. This flag is ignored for Double Byte Character Set (DBCS) data.

Tip: The Write Control Character (WCC), which is the second byte in the first RU of a chain, sets the variable length line format.

Examples

1. To add no data, set return code 2 in register 15.
2. To add data and PCL commands:
 - Put the data and PCL commands in the S2WRKARA field.
 - Put the length of field S2WRKARA in the S2WRKLEN field.
 - Set return code 1 in register 15.
3. To print data wider than 132 characters without adding data or PCL commands:
 - Set flag S2VLN255.
 - Set return code 2 in register 15.

Transparent Data Control exit (APIPPTD2)

NetSpool calls the Transparent Data Control exit (APIPPTD2) whenever NetSpool encounters the Transparent Data control (TRN) in an SCS input data stream. You can use this exit to inspect the transparent data and leave it unchanged, modify it, or delete it.

NetSpool calls the same exit for all VTAM sessions. However, your exit can do different functions based on these input fields in the APIPP1X2 control block:

- **S2FLAGS**: Type of output data stream NetSpool creates for this printer— a line or PCL data stream
- **S2LPTNAM**: NetSpool printer name, also known as the secondary LU (SLU) name
- **S2PLUNAM**: LU name of the VTAM application, also known as the primary LU (PLU) name
- **S2PRTCLS**: LU classes to which the NetSpool printer belongs

When you modify the transparent data, you can also add other SCS controls in addition to the TRN control. The SCS controls supported by NetSpool are described in *z/OS Infoprint Server User's Guide*. The total number of bytes of modified data that you can include is 275. NetSpool does not call the Transparent Data Control exit (APIPPTD2) again for any transparent data included by this exit.

To format lines and pages, NetSpool keeps track of the next output print position, that is, the column number and line number at which the next character is to print. For transparent data, NetSpool increases the column and line number by the length of the transparent data, as specified in the TRN control.

The APIPPTD2 exit lets you change the column and line number of the next print position *after* the transparent data. To assist in calculating the desired line and column numbers, NetSpool provides the exit with the column and line number of the next print position, *before* the transparent data. NetSpool also provides the exit with other page-formatting values.

Input to APIPPTD2

At input to the Transparent Data Control exit (APIPPTD2), the contents of the registers are:

Register 1	Address of the APIPP1X2 control block
Register 13	Address of an 18-word save area for saving the caller's registers
Register 14	Return address

The APIPP1X2 control block is described in Figure 20 on page 184. These fields are valid at entry:

S2FLAGS	Type of conversion and VTAM session:
S2LU1	Set for VTAM LU1 session. Not set for VTAM LU0 or LU3 sessions.
S2PCL	Set when NetSpool will convert the input data stream to a PCL data stream.
S2COL	The column number of the next print position
S2LM	Left margin column number
S2RM	Right margin column number
S2MPP	Maximum presentation position (line length)
S2LINE	The line number of the next print position
S2TM	Top margin line number (channel 1)
S2C2T12	Channel 2-12 line numbers

S2BM	Bottom margin line number
S2MPL	Maximum page length
S2PLUNAM	LU name of the application generating the print request (PLU)
S2LPTNAM	Name of the printer LU that NetSpool is processing (SLU)
S2PRTCLS	Logical unit classes to which the printer LU belongs. Each bit in this field represents a class, with the high-order bit of the leftmost byte representing class 1. For a description of the bits, see Figure 20 on page 184.
S2TPDATA	Transparent data found in the input data stream, containing the TRN control and transparent data

Output from APIPPTD2

At return from the Transparent Data Control exit (APIPPTD2), restore the contents of registers 1 through 14, and place one of these codes in register 15 to tell NetSpool what action to take:

Code Meaning

1	Modify the transparent data. S2WRKARA contains the modified data.
2	Do <i>not</i> modify the transparent data.
4	Close the VTAM session for this printer LU.
8	Do <i>not</i> call exit APIPPTD2 again for any additional transparent data found in this data set. Do <i>not</i> modify the transparent data.
9	Do <i>not</i> call exit APIPPTD2 again for any additional transparent data found in this data set. Modify the transparent data. S2WRKARA contains the modified data.

Also, set these fields in the APIPP1X2 control block:

S2NCNUM	The column number of the next print position after the transparent data. If this field is zero, NetSpool determines the column number.
S2NLNUM	The line number of the next print position after the transparent data. If this field is zero, NetSpool determines the line number.
S2WRKLEN	The length of the data in the work area, S2WRKARA. Set this field to zero to delete the transparent data.
S2WRKARA	The modified TRN control and transparent data. You can also include other SCS controls that NetSpool supports. Valid controls are described in <i>z/OS Infoprint Server User's Guide</i> .

Note: Do *not* change the TRN control or data in the S2TPDATA field.

Examples

- To include the transparent data passed to this exit, unmodified, in the output data stream, set code 2 or 8 in register 15.
- To modify the transparent data in the output data stream:
 - Put the TRN control and the modified transparent data in the S2WRKARA field.
 - Put the length of the TRN control and transparent data into S2WRKARA in the S2WRKLEN field.
 - Set code 1 or 9 in register 15.

	S2PCL	Set when NetSpool will convert the input data stream to a PCL data stream.
S2PLUNAM		LU name of the application generating the print request — primary LU (PLU).
S2LPTNAM		Name of the printer LU that NetSpool is processing — secondary LU (SLU).
S2PRTCLS		Logical unit classes to which the printer LU belongs. Each bit in this field represents a class, with the high order-bit of the leftmost byte representing class 1. For a description of the bits, see Figure 20 on page 184.
S2TPDATA		<p>First byte: The Graphic Escape (GE) character found in the input data stream. That is, the character after GE code X'08'. This character is usually in EBCDIC representation.</p> <p>Second byte: When NetSpool creates a PCL data stream, this byte contains the equivalent character from the printer's code page. This character is usually in ASCII representation. When NetSpool creates a line data stream, this byte contains X'00'.</p>

Output from APIUGEX

At return from the Graphic Escape exit (APIUGEX), restore the contents of all registers, and place one of these return codes in register 15 to tell NetSpool what action to take:

Code Meaning

- | | |
|---|--|
| 1 | Add the replacement data in S2WRKARA to the output data set. |
| 2 | Do <i>not</i> add data in S2WRKARA to the output data set. |

Also, set these fields in the APIPP1X2 control block:

S2WRKLEN Length of work area S2WRKARA.

S2WRKARA Data to replace the Graphic Escape character sequence.

Examples

- To keep the Graphic Escape character, set return code 2 in register 15.
- To replace the Graphic Escape character:
 - Put the replacement data in the S2WRKARA field.
 - Put the length of the data in S2WRKARA into the S2WRKLEN field.
 - Set return code 1 in register 15.

Sample exits and macro

NetSpool provides sample exits and a macro, written in assembler language. The sample exits are in SYS1.SAMPLIB. The macro is in SYS1.MACLIB:

- APIJPTD1, a sample Beginning of File exit for SCS data streams
- APIJPTD2, a sample Transparent Data Control exit
- APIJUBF3, a sample Beginning of File exit for 3270 data streams
- APIJUGEX, a sample Graphic Escape exit
- APIPP1X2, a macro that contains parameter area S2STG and is used in assembling the exits

You can either view the samples online or print them. You can then modify, assemble, and install the exit programs to provide your own versions of the exits. When you link-edit the samples, change the names to:

- APIPPTD1 for the Beginning of File exit for SCS data streams
- APIPPTD2 for the Transparent Data Control exit
- APIUBF3 for the Beginning of File exit for 3270 data streams
- APIUGEX for the Graphic Escape exit

Object code for the sample exits is not provided.

Installing the exits

To install the exits, do these steps to compile and link-edit the exits:

- Make sure that the name of the file is correct. For the required names, see “Sample exits and macro” on page 193.
- For a more secure environment, do not specify AC=1 when you link-edit the exit. The AC=1 link-edit specification, which makes a program APF-authorized, is not required because the NetSpool program that calls the exit is APF-authorized and it loads the exit from an APF-authorized library. When you omit AC=1, the exit runs APF-authorized only when the NetSpool program calls it.
- Put the exit in an APF-authorized library such as SYS1.LINKLIB.

Identify the library containing the exits in either:

- A STEPLIB statement in the NetSpool startup procedure. To provide a different exit program for each NetSpool started task, use a unique STEPLIB for each startup procedure.
- A library concatenated to LNKLST. If you place an exit in a LNKLST library, you must use the same exit program for each NetSpool started task.

Modifying the exits

If you replace an exit while NetSpool is running, enter this command:

```
F jobname{.id},RELOAD EXITS
```

jobname is the name of the NetSpool startup procedure.

id is the number that identifies a procedure if more than one NetSpool task is started from the same startup procedure.

When you enter the command, the exits are reloaded at the beginning of the next file:

- LU0/LU3 exits are reloaded at the beginning of the next LU0 or LU3 file.
- LU1 exits are reloaded at the beginning of the next LU1 file.

Setting environment variables for NetSpool

If dynamic configuration is not enabled, NetSpool uses this environment variable, which you can specify in the **aopstart** EXEC:

AOP_ALLOW_ALL_CHARACTERS_IN_LINE_DATA

This environment variable determines how Infoprint Server detects the data format of documents that contain line data. Valid line data contains

carriage control characters, table reference characters, binary zeroes, and printable characters. However some applications might create line data that contains unprintable characters.

If this environment variable is set to YES, if line data contains unprintable characters, Infoprint Server detects the data format as “line data”.

If this environment variable is not set, if line data contains unprintable characters, Infoprint Server detects the data format as “other”. Line data that Infoprint Server detects as “other” might not print correctly because Infoprint Server does not convert it to the printer’s code page.

Tips:

- Job submitters can specify the data format in the **document-format** job attribute. For example, job submitters can specify **document-format=line** in the PRTATTRS parameter on the OUTPUT JCL statement. If the **document-format** job attribute is specified, Infoprint Server ignores this environment variable because it does not detect the data format.
- Set this environment variable to YES if Infoprint Server does not detect the line data format and job submitters cannot specify the **document-format=line** job attribute.
- This environment variable does not affect how NetSpool checks SCS and 3270 data streams for validity. The input SCS and 3270 data streams must contain only valid characters.

Rules:

- If dynamic configuration is enabled, specify the equivalent **allow-all-characters-in-line-data** attribute in the system configuration definition.
- If dynamic configuration is not enabled, specify this environment variable in the **aopstart EXEC**.

Default: The environment variable is not set.

Example: AOP_ALLOW_ALL_CHARACTERS_IN_LINE_DATA=YES

NetSpool also uses the LANG, LIBPATH, NLSPATH, and PATH environment variables, which you can specify in the **aopstart EXEC**.

Related information: For information about:

- How to edit the **aopstart EXEC**, see “Editing the aopstart EXEC” on page 57
- Environment variables, see “Setting environment variables for the Printer Inventory Manager” on page 43

Editing the system configuration definition for NetSpool

The system configuration definition in the Printer Inventory contains attributes that let you customize Infoprint Server while Infoprint Server is running. If dynamic configuration is enabled, you can specify these attributes in the system configuration definition to customize NetSpool:

allow-all-characters-in-line-data = yes | no

This attribute determines how Infoprint Server detects the data format of documents that contain line data. Valid line data contains carriage control characters, table reference characters, binary zeroes, and printable characters. However some applications might create line data that contains unprintable characters.

- yes** Infoprint Server detects the data format as “line data” even if the input data contains unprintable characters.
- no** Infoprint Server detects the data format as “other” if the input data contains unprintable characters. Line data that Infoprint Server detects as “other” might not print correctly because Infoprint Server does not convert it to the printer's code page.

Tips:

- Job submitters can specify the data format in the **document-format** job attribute. For example, job submitters can specify **document-format=line** in the PRTATTRS parameter on the OUTPUT JCL statement. If the **document-format** job attribute is specified, Infoprint Server ignores this attribute because it does not detect the data format.
- Specify **allow-all-characters-in-line-data=yes** if Infoprint Server does not detect the line data format and job submitters cannot specify the **document-format=line** job attribute.
- This attribute does not affect how NetSpool checks SCS and 3270 data streams for validity. The input SCS and 3270 data streams must contain only valid characters.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify the equivalent AOP_ALLOW_ALL_CHARACTERS_IN_LINE_DATA environment variable in the **aopstart** EXEC.
- This attribute does not apply to IP PrintWay basic mode.

Default: allow-all-characters-in-line-data = no

Example: allow-all-characters-in-line-data = yes

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.

The prefix must be two alphanumeric or national (@, \$, and #) 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 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.
- 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.

Tips:

- z/OS also assigns a unique z/OS job ID to each output data set. The z/OS job ID might be different from the job ID of the output data set on the JES spool. JES operator commands use this z/OS job ID.

- The job ID that Print Interface returns to the job submitter does not contain this prefix.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: job-prefix = PS

Example: job-prefix = 'P#'

Example: This screen shows a portion of the ISPF panel that you use to edit the system configuration definition for NetSpool:

```
                                Infoprint Server: System Configuration
General:
_ Allow all characters in line data
Job ID prefix . . PS
```

Related information: For information about:

- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 68.
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 80.

Chapter 7. Customizing IP PrintWay

This information describes how to customize the IP PrintWay component of Infoprint Server. You must customize IP PrintWay if you want to transmit output data sets from the JES spool to (1) TCP/IP-attached ASCII printers or print servers, (2) VTAM-controlled printers in an SNA network, or (3) email destinations. For an overview of IP PrintWay and how this component fits into your system, see “IP PrintWay” on page 15.

You can run either IP PrintWay basic mode or IP PrintWay extended mode:

- IP PrintWay basic mode uses the Functional Subsystem Interface (FSI) to obtain output data sets from the JES spool.
- IP PrintWay extended mode uses the z/OS Sysout Application Programming Interface (SAPI) to obtain output data sets from the JES spool.

Tip: Run IP PrintWay extended mode to obtain better performance, improved usability, and additional functions. IBM does not plan to add new functions to IP PrintWay basic mode in future releases. For comparison information, see “Comparing IP PrintWay extended mode to basic mode” on page 202. For information about actions you need to take to migrate from IP PrintWay basic mode to extended mode, see Appendix G, “Migrating from IP PrintWay basic mode to extended mode,” on page 479.

Table 30 summarizes the tasks to customize IP PrintWay extended mode. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies.

Table 30. Customization tasks for IP PrintWay extended mode

Task (IP PrintWay extended mode)	Condition	See page
Comparing IP PrintWay extended mode to basic mode	Required	202
Identifying software requirements for IP PrintWay	Required	205
Editing the <code>aopd.conf</code> file for IP PrintWay (extended mode)	Required	205
Increasing the space available in <code>/var/Printsrv</code> directory (extended mode)	Required	207
Updating the BPXPRMxx member of SYS1.PARMLIB (extended mode)	Required	207
Creating IP PrintWay job-selection rules (extended mode)	Required	208
Authorizing IP PrintWay to select data sets from the JES spool (extended mode)	Required	214
Authorizing IP PrintWay to write SMF type 6 records (extended mode)	Required	215
Setting environment variables for IP PrintWay (extended mode)	Optional: To customize IP PrintWay processing	215
Editing the system configuration definition for IP PrintWay (extended mode)	Optional: To specify dynamic attributes	223

Table 30. Customization tasks for IP PrintWay extended mode (continued)

Task (IP PrintWay extended mode)	Condition	See page
APF-authorizing SYS1.IMAGELIB (extended mode)	Optional: To use FCBs	230
Customizing from AFP transforms for IP PrintWay (extended mode)	Optional: If you use the AFP to PCL or AFP to PostScript transform	230
Defining printer host names in the DNS	Required	231
Customizing z/OS Communications Server	Required	232, 117
Creating a VTAM APPL resource definition for IP PrintWay	Optional: To print on VTAM-controlled printers	236
Writing IP PrintWay exit and filter programs	Optional: To customize IP PrintWay processing	237
Customizing PSF for sharing network printers	Optional: To share printers with PSF	324

Table 31 summarizes the tasks to customize IP PrintWay basic mode. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies.

Table 31. Customization tasks for IP PrintWay basic mode

Task (IP PrintWay basic mode)	Condition	See page
Comparing IP PrintWay extended mode to basic mode	Required	202
Identifying software requirements for IP PrintWay	Required	205
Defining printer host names in the DNS	Required	231
Customizing z/OS Communications Server	Required	232, 117
Creating a VTAM APPL resource definition for IP PrintWay	Optional: To print on VTAM-controlled printers	236
Writing IP PrintWay exit and filter programs	Optional: To customize IP PrintWay processing	237
Updating the BPXPRMxx member of SYS1.PARMLIB (basic mode)	Optional: To start multiple IP PrintWay FSAs	301
APF-authorizing SYS1.IMAGELIB (basic mode)	Optional: To use FCBs	301
Defining the IP PrintWay functional subsystem (basic mode)	Required	302
Creating FSS and FSA definitions in the Printer Inventory (basic mode)	Optional: To print on VTAM-controlled printers or to change default configuration options	312
Creating the transmission-queue data set (basic mode)	Required	316
Creating the IP PrintWay message-log data set (basic mode)	Required	319
Customizing the IP PrintWay startup procedure (basic mode)	Required	320
Changing Language Environment runtime options (basic mode)	Optional: To change default options	324

Table 31. Customization tasks for IP PrintWay basic mode (continued)

Task (IP PrintWay basic mode)	Condition	See page
Customizing PSF for sharing network printers	Optional: To share printers with PSF	324

If you have already customized IP PrintWay to use the LPR or direct sockets transmission protocol and you now want to use the VTAM, email, or IPP protocol, you need to do the additional customization tasks summarized in Table 32.

Table 32. Customization tasks for IP PrintWay protocols

Task	Protocol					See page
	LPR	Direct sockets	IPP	VTAM	Email	
Install Coax Printer Support	No	No	No	Yes	No	205
Create a VTAM APPL definition	No	No	No	Yes	No	236
Specify the IP PrintWay APPL ID in an FSS definition (basic mode)	No	No	No	Yes	No	312
Specify the IP PrintWay APPL ID in the AOP_APPLID environment variable (extended mode)	No	No	No	Yes	No	215
Customize z/OS UNIX sendmail	No	No	No	No	Yes	234
Specify environment variables in the IP PrintWay startup procedure (basic mode)	No	No	Yes	No	Yes	320
Specify environment variables in the aopstart EXEC (extended mode)	No	No	No	Yes	Yes	215

Table 33 lists other components of Infoprint Server that you might need to customize:

Table 33. Other components to customize for IP PrintWay

Task	Condition	See page
Customizing the Printer Inventory Manager	Required	27
Customizing Infoprint Central	Optional but suggested (extended mode only)	327
Customizing Transform Interface	Optional: To transform data from one format to another	157
Customizing the Print Interface LPD	Optional: To transform data from one format to another (basic mode only)	116 and 132

Related tasks: After you customize IP PrintWay, do these tasks. For information about these tasks, see *z/OS Infoprint Server Operation and Administration*:

- Create a printer definition in the Printer Inventory for each target printer and specify attributes required by IP PrintWay. To transform data, also specify attributes required by the transforms. IP PrintWay extended mode and basic mode use the same printer definitions.

- Start IP PrintWay:
 - IP PrintWay basic mode: Start the IP PrintWay basic mode functional subsystems (FSAs).
 - IP PrintWay extended mode: Use either the **aopstart** command or the AOPSTART JCL procedure to start the IP PrintWay extended mode daemons.

Comparing IP PrintWay extended mode to basic mode

IP PrintWay extended mode provides these 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. For more information about Infoprint Central, see "New functions". For a comparison of Infoprint Server ISPF panels and Infoprint Central functions, see *z/OS Infoprint Server Operation and Administration*.
 - 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.
 - If you enable dynamic configuration, you can configure IP PrintWay extended mode without having to stop and restart it.
- New 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 data sets larger than 2 gigabytes if space is available in the file system.
 - IP PrintWay extended mode can print on printers that have IPv6 addresses.
 - IP PrintWay extended mode writes the printer address for all protocol types in the System Management Facilities (SMF) type 6 record. It also includes job accounting information 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.

- IP PrintWay extended mode can restart printing after an error from the last page that the printer reported printed successfully. This can prevent duplicate pages and save paper. The printer must support PJJL commands and the direct sockets printing protocol.
- IP PrintWay extended mode can record the number of pages that printed successfully in the SMF type 6 accounting record that IP PrintWay writes. The printer must support PJJL commands and the direct sockets printing protocol.
- Authorized users can use Infoprint Central, a web-based application, to do these additional printer functions:
 - Stop printing to printers. Restart printing to stopped printers.
 - Ping printers and run the **traceroute** command to pinpoint a network problem.
 - 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.
 - Turn printers online and offline, and reset them.
 - See all messages in the Infoprint Server common message log for a printer.
 - Link to the web pages of TCP/IP-attached printers.
 - See the status that a printer reports, such as whether the printer is offline or is out of paper or toner.
- Authorized users can use Infoprint Central to do these additional print job functions:
 - 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.
 - See all messages in the Infoprint Server common message log for a print job.
- When you send documents to email destinations, you can:
 - Include text and line-data documents inline in the body of an email.
 - Specify the email recipients in an email header in line-data documents.
 - Include a standard message in the beginning of emails.
- IP PrintWay extended mode writes messages in the Infoprint Server common message log.

Limitations:

- In a JES3 sysplex with a shared spool, if you want to run Infoprint Server on more than one system in the sysplex, you must:
 - Define different job-selection rules for IP PrintWay extended mode on each system so that IP PrintWay on each system selects different print jobs. 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 the job-selection rules for IP PrintWay running on the same system. For example, Print Interface and NetSpool

running on system A can allocate print jobs in class P, while Print Interface and NetSpool running on system B can allocate print jobs in class Q.

- 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 does.
- IP PrintWay extended mode prints all data sets in an output group on the same printer, which is the printer selected for the first data set in the output group. In addition, retry values and retention periods for the first data set in the output group apply to all data sets in the output group.

Table 34 summaries how IP PrintWay extended mode differs from IP PrintWay basic mode.

Table 34. How IP PrintWay extended mode differs from basic mode

Area	How IP PrintWay basic and extended modes differ
Administration	IP PrintWay basic mode and IP PrintWay extended mode can use the same printer definitions. However, some fields in the printer definitions apply only to basic mode or extended mode. Fields that do not apply to are ignored.
Application programming	User exits for IP PrintWay basic mode and extended mode differ in these ways: <ul style="list-style-type: none"> • IP PrintWay extended mode does not support the Message exit. • IP PrintWay extended mode does not pass some values in the ANFUEXTP control block to the exits. • The exits cannot hold or delete output data sets.
Auditing	IP PrintWay extended mode writes different fields in the SMF type 6 record than IP PrintWay basic mode. IP PrintWay extended mode writes the IP address of the printer in a different field and in a different format. IP PrintWay extended mode also writes accounting information from the JOB JCL statement in the SMF type 6 record.
Customization	The major differences between IP PrintWay basic and extended modes are: <ul style="list-style-type: none"> • To start IP PrintWay extended mode daemons, you must edit the Infoprint Server configuration file (aopd.conf) . • To control which data sets IP PrintWay extended mode selects from the JES spool, you must define IP PrintWay job-selection rules in the Printer Inventory. • You specify environment variables for IP PrintWay extended mode in the aopstart EXEC instead of in the basic mode startup procedure. • You might need more space in the /var/Printsrv directory because IP PrintWay extended mode writes data to this directory instead of to hiperspace. • You must customize Infoprint Central so that operators can work with IP PrintWay extended mode print jobs and printers. • You do not need to define JES functional subsystems (FSSs and FSAs) for IP PrintWay extended mode.
Diagnosis	The major differences between IP PrintWay basic and extended modes are: <ul style="list-style-type: none"> • To trace IP PrintWay extended mode, you use the Infoprint Server trace facility. IP PrintWay basic mode uses GTF tracing. • IP PrintWay extended mode writes messages to the common message log. IP PrintWay basic mode writes messages to a separate data set.

Table 34. How IP PrintWay extended mode differs from basic mode (continued)

Area	How IP PrintWay basic and extended modes differ
Job submission	IP PrintWay extended mode prints all print jobs in the same JES output group to the same printer, using the same retry and retention values.
Operations	<p>The major differences between IP PrintWay basic and extended modes are:</p> <ul style="list-style-type: none"> • To start and stop IP PrintWay extended mode, you use the AOPSTART JCL procedure instead of the IP PrintWay basic mode startup procedure. • To start and stop individual printers, you use Infoprint Central instead of JES commands. • To work with IP PrintWay extended mode print jobs on the JES spool, you can use either JES commands or Infoprint Central instead of the ISPF panels for the IP PrintWay basic mode transmission-queue data set. Infoprint Central lets you work with print jobs that IP PrintWay is currently processing. • To view messages from IP PrintWay extended mode, you can use either Infoprint Central or the aoplogu command instead of the ISPF panel for IP PrintWay basic mode messages. Use Infoprint Central to see messages for selected print jobs and printers. Use the aoplogu command to copy messages in a specified time range to a file.

Identifying software requirements for IP PrintWay

This additional IBM software beyond the z/OS base elements is required:

- To print on VTAM-controlled printers, Infoprint Coaxial Printer Support for z/OS (5655-N62)

Editing the aopd.conf file for IP PrintWay (extended mode)

The **aopd.conf** file contains attributes that Infoprint Server uses when it starts. The default location of this file is **/etc/Printsrv/aopd.conf**.

To start the IP PrintWay extended mode daemons, you must specify the following attribute in the **aopd.conf** file:

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 any of the supported 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 when specifying printer addresses for the same physical printer. For example, if the administrator defines more than one printer definition for the same physical printer, in each printer definition, the administrator must specify exactly the same IP address or host name so that IP PrintWay recognizes that all the printer definitions refer to the same printer.

Tip: If administrators and job submitters always specify dotted-decimal IP addresses for printers, consider specifying **resolve-printway-printers = no**. This limits the number of DNS lookups that IP PrintWay does.

Rules:

- If you specify **resolve-printway-printers = no**, when the operator searches for a printer in Infoprint Central, the operator must enter the printer's dotted-decimal IP address or host name exactly as it is specified in the printer definition or in the DEST=IP: parameter of the OUTPUT JCL statement. Otherwise, Infoprint Central cannot find the printer. For example, if the printer definition contains the printer's IP address, the operator must search for the printer using the IP address instead of the printer's host name.
- If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: resolve-printway-printers = yes

Example: resolve-printway-printers = no

start-daemons = { outd }

To start the IP PrintWay extended mode daemons, **aopoutd** and **aopwsmd**, add the **outd** value to any existing values in this attribute. Enclose all values in braces. This attribute is required.

Rule: If you add **outd** while Infoprint Server is running, start Infoprint Server again. You do not need to stop any Infoprint Server daemons before restarting.

Default: start-daemons = {lpd}

Example: In this example, the Printer Inventory Manager daemons, the LPD, the IP PrintWay extended mode daemons, and the Infoprint Central daemon start: start-daemons = {lpd outd ssid}

If dynamic configuration is not enabled, you can also specify the following attribute in the **aopd.conf** configuration file:

smf-recording = yes | no

Indicates whether IP PrintWay extended mode writes SMF type 6 records for data sets that it sends to a printer or email destination. SMF type 6 records contain accounting information.

yes IP PrintWay extended mode writes SMF type 6 records.

no IP PrintWay extended mode does not write SMF type 6 records.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.

- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.
- This attribute does not apply to IP PrintWay basic mode.

Default: smf-recording = yes

Example: smf-recording = no

Related information: For information about:

- How to edit the **aopd.conf** file, see “Steps for editing the Infoprint Server configuration file (aopd.conf)” on page 42.
- All attributes in the **aopd.conf** file, see Appendix B, “Infoprint Server files,” on page 389.
- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 68.

Increasing the space available in /var/Printsrv directory (extended mode)

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 file system mounted at the **/var** or **/var/Printsrv** mount point. For space requirements, see “Calculating DASD space requirements for the **/var/Printsrv** directory” on page 32.

Updating the BPXPRMxx member of SYS1.PARMLIB (extended mode)

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 BPXPRMxx member of SYS1.PARMLIB:

Parameter	Meaning
MAXPROCUSER	The maximum number of processes that a single z/OS UNIX user ID can have concurrently active. Guideline: Add at least 70 for Infoprint Server processes.
MAXPROCSYS	The maximum number of processes that the z/OS system allows. Guideline: 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 path name of a z/OS UNIX file that contains a list of MVS data sets sanctioned by your installation for use as step libraries. Follow these guidelines:

- In the z/OS UNIX file, specify the name of each MVS data set on a separate line without quotation marks.

- Give the z/OS UNIX file execute permission for **other**. IBM suggests that UID 0 own the file and that the file's permissions be 755 or 555.

Tip: Also specify the names of the MVS data sets in the STEPLIB environment variable in the **aopstart** EXEC. See “Setting environment variables for IP PrintWay (extended mode)” on page 215.

Example: Specify these statements in the BPXPRMxx member:

```
MAXPROCUSER(70) /*70(Infoprint Server) */
MAXPROCSYS(450) /*50(PrintWay extended)+200(PrintInterface/NetSpool)+200(initial)*/
STEPLIBLIST('/etc/steplib')
```

Tip: You can use the SETOMVS or SET OMVS command to dynamically set the MAXPROCUSER, MAXPROCSYS, and STEPLIBLIST values. However, to make a permanent change, edit the BPXPRMxx member that is used for IPLs.

Related information:

- For information about the BPXPRMxx member, see *z/OS MVS Initialization and Tuning Reference*.
- For additional planning information and for information about how to create the file specified by the STEPLIBLIST statement, see *z/OS UNIX System Services Planning*.

Creating IP PrintWay job-selection rules (extended mode)

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.

In each job-selection rule in the Printer Inventory, you can specify the job-selection criteria shown in Table 35.

Table 35. IP PrintWay job-selection rules

Job-selection criteria	Description	Examples
DEST	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 (see note). Note: In JES2, R*, RM*, RMT*, U*, and N* do not match DEST parameters with a JES route code of remote, special local, local, anylocal, or NJE. Also, do not specify an asterisk for destinations that JES DESTID initialization statements define.	PRT001
CLASS	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.	D, N

Table 35. IP PrintWay job-selection rules (continued)

Job-selection criteria	Description	Examples
FORMS	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 (see note).	STD, ACCT*
Creator	The z/OS user ID associated with the print jobs. For print jobs submitted with JCL, the Ip 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 allocates on the JES spool, specify the user ID of the user who started the Infoprint Server daemons. Wildcards are allowed (see note).	USER01
Writer	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 (see note).	PRINTWAY
DEST IP address	Indicates whether print jobs can specify the DEST=IP: parameter on the OUTPUT JCL statement: <ul style="list-style-type: none"> • Include: Select only print jobs that specify the DEST=IP: parameter. • Exclude: Select only print jobs that do not specify the DEST=IP: parameter. • Ignore: Select print jobs that specify the DEST=IP: parameter and jobs that do not specify the DEST=IP: parameter. 	Include
Record limit	The lower and upper limits for the total number of input records in all non-AFP documents (data sets) in a print job (output group). JES calculates the number of input records. Note: One input record is not always equivalent to one printed line. For example, one record in a document that Print Interface creates on the JES spool can contain up to 32K bytes of data. Therefore, a large document can have a small number of records.	To select print jobs with 1200 records or less: Lower: 0 Upper: 1200
Page limit (AFP only)	The lower and upper limits for the total number of input pages in all AFP documents (data sets) in a print job (output group). JES calculates the number of input pages.	To select print jobs of 20 AFP pages or less: Lower: 0 Upper: 20
<p>Note: You can specify an asterisk (*) and a question mark (?) in any position in the DEST, CLASS, FORMS, Creator, and WRITER fields as wildcards:</p> <ul style="list-style-type: none"> • An * represents zero or more characters. • A ? represents one character. 		

Tips:

- 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.
- **JES3 sysplex:** In JES3 systems only, if the systems in the sysplex share the JES spool, you can 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:
 - 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.
- For best performance, specify job-selection criteria that are unique to IP PrintWay print jobs. For example, if you specify **CLASS P** as the job selection criterion, all print jobs in JES output class P should be IP PrintWay print jobs.
- If some performance degradation is acceptable, you can specify job-selection criteria that are not unique to IP PrintWay print jobs. For example, assume that you want both IP PrintWay and PSF to process print jobs in output class P:
 - IP PrintWay is to process print jobs that request an IP PrintWay printer in the **FSSDATA JCL** parameter; in a combination of the **CLASS, DEST** and **FORMS JCL** parameters; or in the **DEST=IP: JCL** parameter.
 - PSF is to process print jobs that request a PSF printer in the **DEST, FORMS, WRITER,** or other JCL parameters.

In this case, you can specify **CLASS P** as the IP PrintWay job-selection criterion. IP PrintWay selects any print job on the JES spool in output class P. However, if IP PrintWay does not find a printer definition in the Printer Inventory for the print job, and the **DEST=IP: JCL** parameter was not specified, it releases the print job to JES so that PSF (or any other program) can process it. IP PrintWay does not select the print job again unless the operator releases it.

If you want IP PrintWay to release print jobs back to JES without recording any error information, specify the **AOP_IGNORE_DCF_ROUTING_ERRORS=YES** environment variable in the **aopstart EXEC**. Specifying this environment variable improves performance and reduces the size of Infoprint Server databases. For information about this environment variable, see “Setting environment variables for IP PrintWay (extended mode)” on page 215.

- If you export the Printer Inventory from one system and import it to a second system, the IP PrintWay job-selection rules are exported to the second system along with the printer definitions and other definitions. Therefore, if you want IP PrintWay extended mode to use different job-selection rules on each system, you must customize the job-selection rules on the second system after you import the Printer Inventory. For example, you might want to create a script that runs the PIDU **modify** command to start and stop job-selection rules in the Printer Inventory on the second system. You could run this script each time you import the Printer Inventory to the second system. You can modify the **job-selection-status** attribute to start (enable) or stop (disable) a job-selection rule.

Tips for migrating from IP PrintWay basic mode:

- Job-selection rules for IP PrintWay extended mode are comparable to the JES work selection criteria you specify in IP PrintWay basic mode functional subsystem applications (FSAs). In your job-selection rules, specify the same work selection criteria that are specified in the WS parameter of the JES initialization statements for the IP PrintWay FSAs.

Examples:

- If you have two IP PrintWay FSAs, one that selects print jobs in class D and another that selects print jobs in class N, create either one or two job-selection rules:
 - If you want IP PrintWay to start and stop processing print jobs in the two classes at different times, create two job-selection rules. Specify class D in one rule and class N in the second rule.
 - If you want to start and stop processing print jobs in these classes at the same time, create one job-selection rule. Specify classes D and N in that rule.
- If you have two or more IP PrintWay FSAs that all select print jobs in class P, create one job-selection rule. Specify class P in that rule.
- 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. For example, both IP PrintWay basic mode and IP PrintWay extended mode should not select print jobs in the same class.

Creating job-selection rules in the Printer Inventory (extended mode)

You can use either the Infoprint Server ISPF panels or the Printer Inventory Definition Utility (PIDU) to create, modify, and delete job-selection rules in the Printer Inventory. If you do not create any job-selection rules, IP PrintWay does not select any print jobs from the JES spool to process.

Note: You must start Infoprint Server before you can create job selection rules.

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 workload 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.

You do not need to restart IP PrintWay after you create, modify, or delete job-selection rules in the Printer Inventory. When you create or modify a job-selection rule, IP PrintWay immediately starts using the new or changed rule unless you specify the PIDU **job-selection-status=disabled** attribute. When you delete a job-selection rule from the Printer Inventory, IP PrintWay immediately stops using the rule to select print jobs. However, IP PrintWay continues to process print jobs that it has already selected from the JES spool.

Example 1: This Infoprint Server ISPF panel shows a job-selection rule named *night* for processing print jobs in output class N.

```

Job Selection Rule

Rule name . . night
Description . Select overnight jobs in class N. Start at 12 AM. Stop at 7 AM. (extend)
Operator security profile
    . . . myprofile
-----
DEST. . . _____
CLASS . . N _____
-----
FORMS . . _____
-----
Creator . _____
WRITER. . _____

DEST IP address . 3 1. Include 2. Exclude 3. Ignore

Record limit . . . . . Lower _____ Upper _____
Page limit (AFP-only). Lower _____ Upper _____

```

Example 2: This Infoprint Server ISPF panel shows a job-selection rule named *small print jobs* for processing smaller print jobs in class P. The print jobs must contain less than (or equal to) 1200 records and 20 AFP pages.

```

Job Selection Rule

Rule name . . small
Description . Select smaller print jobs. (extend)
Operator security profile
    . . . myprofile
-----
DEST. . . _____
CLASS . . P _____
-----
FORMS . . _____
-----
Creator . _____
WRITER. . _____

DEST IP address . 3 1. Include 2. Exclude 3. Ignore

Creator . _____
WRITER. . _____

Record limit . . . . . Lower 0 Upper 1200
Page limit (AFP-only). Lower 0 Upper 20

```

Related information: For information about:

- How to use the Infoprint Server ISPF panels and the PIDU program, see *z/OS Infoprint Server Operation and Administration*.
- How to set up security for job-selection rules and specify the **Operator security profile** field, see “Setting up security for IP PrintWay job-selection rules” on page 354.

Starting and stopping job-selection rules (extended mode)

The operator can use Infoprint Central to start and stop IP PrintWay job-selection rules, or you can set up an automatic process to start and stop job-selection rules at different times of the day.

When a job-selection rule is stopped, IP PrintWay no longer uses that rule to select print jobs. However, any print jobs already selected using that job-selection rule continue to be processed. When the operator stops a job-selection rule and then restarts IP PrintWay, the job-selection rule remains stopped.

The security administrator can define a RACF resource profile in the PRINTSRV class to restrict who can display, start, and stop job-selection rules using Infoprint Central. You must specify the name of that RACF resource profile in the job-selection rule. If you do not specify a RACF resource profile in the job-selection rule, any Infoprint Central user authorized to read the Printer Inventory can display, start, and stop the job-selection rule.

You can set up a cron job to start and stop job-selection rules automatically.

Steps for starting and stopping job-selection rules at the same time each day: To start job-selection rule night at 12 AM each day and stop it at 7 AM:

1. Create two files to contain shell scripts. For example, create `/u/userid/do.dayrules` and `/u/userid/do.nightrules`. Your login user ID is `userid`.

Examples: Put these lines in file `/u/userid/do.dayrules`:

```
. /etc/profile
. /u/userid/.profile
pidu -c "modify job-selection-rule night job-selection-status=disabled;"
```

Put these lines in file `/u/userid/do.nightrules`:

```
. /etc/profile
. /u/userid/.profile
pidu -c "modify job-selection-rule night job-selection-status=enabled;"
```

Tips:

- If you do not set the `PATH`, `NLSPATH`, and optional `AOPCONF` environment variables in `/etc/profile` or `/u/userid/.profile`, replace the first two lines in the files with these **export** commands to export the environment variables. You can omit the `AOPCONF` variable if the Infoprint Server configuration file is `/etc/Printsrv/aopd.conf`.

```
export PATH=/usr/lpp/Printsrv/bin:/bin:$PATH
export NLSPATH=/usr/lpp/Printsrv/%L/%N:/usr/lpp/Printsrv/En_US/%N:$NLSPATH
export AOPCONF=/etc/Printsrv/aopd.conf
pidu ... (as shown above)
```

- When the **cron** daemon runs the commands in your **crontab** entry, it does not first run your **.profile** to set environment variables.

2. Set the execute permission bits in the files:

Examples:


```
chmod +x /u/userid/do.dayrules
chmod +x /u/userid/do.nightrules
```

3. Use the **crontab** command with the **-e** option to insert these lines into the **crontab** entry:

```
00 07 * * * /u/userid/do.dayrules
00 00 * * * /u/userid/do.nightrules
```

Related information: For information about:

- How to use the **crontab** command, see *z/OS UNIX System Services Command Reference*.
- How to use Infoprint Central to start and stop job-selection rules, see the online help system for Infoprint Central.

Authorizing IP PrintWay to select data sets from the JES spool (extended mode)

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 suggests that you give the AOOPER group access to profiles in the JESSPOOL class because users who start Infoprint Server must be members of the AOOPER group or have a UID of 0. (AOOPER is the default group name for Infoprint Server operators. However, your installation can use a different name for this group.) 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 profiles in the JESSPOOL class. If you start Infoprint Server in the **/etc/rc** file, give user ID ROOT access to the profiles.

Examples:

- 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 AOOPER group UPDATE access to the broad profile, enter these RACF commands:

```
PERMIT N1.*.*.D*.* CLASS(JESSPOOL) ID(AOOPER) ACCESS(UPDATE)
SETROPTS RACLIST(JESSPOOL) REFRESH
```

Tips:

- 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 so that your change takes effect. For example, enter these MVS commands to run the AOPSTOP and AOPSTART JCL procedures:

```
START AOPSTOP,OPTIONS='-d outd'
START AOPSTART
```

- Do not activate the JESSPOOL class while IP PrintWay extended mode is running.

Authorizing IP PrintWay to write SMF type 6 records (extended mode)

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.

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 use a different name for this group.) Therefore, IBM suggests 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.

Tip: 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.

Steps for authorizing IP PrintWay to write SMF type 6 records:

1. Define the BPX.SMF profile in the FACILITY class if the profile does not already exist. Enter this RACF command:

```
RDEFINE FACILITY (BPX.SMF) UACC(NONE)
```

2. Give the AOOPER group READ access to the BPX.SMF profile. Enter these RACF commands:

```
PERMIT BPX.SMF CLASS(FACILITY) ID(AOOPER) ACCESS(READ)
SETROPTS RACLIST(FACILITY) REFRESH
```

3. Stop and restart the IP PrintWay extended mode daemons so that your changes take effect. For example, enter these MVS commands to run the AOPSTOP and AOPSTART JCL procedures:

```
START AOPSTOP,OPTIONS='-d outd'
START AOPSTART
```

Setting environment variables for IP PrintWay (extended mode)

You must specify environment variables for IP PrintWay extended mode if any of these conditions apply:

- More than one TCP/IP stack exists in your installation (see `_BPXK_SETIBMOPT_TRANSPORT`).
- The library that contains IP PrintWay exits is not in the system LNKLIST (see STEPLIB).
- The SYS1.IMAGELIB library that contains FCBs is not in the system LNKLIST (see STEPLIB).

You can specify the following environment variables in the **aopstart EXEC**:

`_BPXK_SETIBMOPT_TRANSPORT`

The job name that is associated with the z/OS UNIX TCP/IP stack. For example, if the job name for the z/OS UNIX stack is TCPIPOE, specify `_BPXK_SETIBMOPT_TRANSPORT=TCPIPOE`. If you have only one TCP/IP stack in your installation, you do not need to specify this variable.

Tip for migrating from IP PrintWay basic mode: The name you specify in the `_BPXK_SETIBMOPT_TRANSPORT` environment variable might not match the TCP/IP job name that is specified in the IP PrintWay basic mode FSS definition in the Printer Inventory or in the TCPIP.DATA data set. This is because IP PrintWay basic mode can run on any TCP/IP stack, while IP PrintWay extended mode must use the z/OS UNIX stack.

STEPLIB

The names of step libraries that are not in the system LNKLIST. Separate library names with a colon. Specify:

- The library that contains optional IP PrintWay exits
- SYS1.IMAGELIB if you want IP PrintWay to use FCBs to format data with carriage controls
- Language Environment runtime libraries (*hlq.SCEERUN* and *hlq.SCEERUN2*) and C++ runtime library (*hlq.SCLBDLL*)

Rule: Specify these step libraries in the file specified by the STEPLIBLIST statement in the BPXPRMxx PARMLIB member. See “Updating the BPXPRMxx member of SYS1.PARMLIB (extended mode)” on page 207.

Default: None

Example: STEPLIB=SYS1.IMAGELIB:PW.EXITLIB

If dynamic configuration is not enabled, you must also specify these environment variables for IP PrintWay extended mode if any of these conditions apply:

- You print line data that contains unprintable characters (see `AOP_ALLOW_ALL_CHARACTERS_IN_LINE_DATA`).
- You print on VTAM-controlled printers (see `AOP_APPLID`).
- Line or text data created by batch applications print with extra blank characters at the end of lines (see `AOP_BLANK_TRUNCATION_CLASSES`).
- You want to suppress errors due to missing printer definitions (see `AOP_IGNORE_DCF_ROUTING_ERRORS`).
- You want to customize the IP PrintWay email function (see `AOP_MAIL_DO_NOT_ADD_SUFFIX` and `AOP_MAIL_PRESERVE_SUFFIXES`, and `AOP_MAIL_USE_FIRST_ADDRESS`).
- You send output to email destinations and the z/OS UNIX **sendmail** command was not installed in the default directory or you need to specify **sendmail** command options (see `AOPMAILER` and `AOPMAILER_OPTIONS`).

If dynamic configuration is not enabled, you can specify these variables in the **aopstart** EXEC:

AOP_ALLOW_ALL_CHARACTERS_IN_LINE_DATA

This environment variable determines how Infoprint Server detects the data format of documents that contain line data. Valid line data contains carriage control characters, table reference characters, binary zeroes, and printable characters. However some applications might create line data that contains unprintable characters.

If this environment variable is set to YES, if line data contains unprintable characters, Infoprint Server detects the data format as “line data”.

If this environment variable is not set, if line data contains unprintable characters, Infoprint Server detects the data format as “other”. Line data that Infoprint Server detects as “other” might not print correctly because Infoprint Server does not convert it to the printer's code page.

Tips:

- Job submitters can specify the data format in the **document-format** job attribute. For example, job submitters can specify **document-format=line** in the PRTATTRS parameter on the OUTPUT JCL statement. If the **document-format** job attribute is specified, Infoprint Server ignores this environment variable because it does not detect the data format.
- Set this environment variable to YES if Infoprint Server does not detect the line data format and job submitters cannot specify the **document-format=line** job attribute.
- This environment variable does not affect how NetSpool checks SCS and 3270 data streams for validity. The input SCS and 3270 data streams must contain only valid characters.

Rules:

- If dynamic configuration is enabled, specify the equivalent **allow-all-characters-in-line-data** attribute in the system configuration definition.
- If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

Default: The environment variable is not set.

Example: AOP_ALLOW_ALL_CHARACTERS_IN_LINE_DATA=YES

AOP_APPLID

The application program ID that IP PrintWay extended mode uses to establish a VTAM session with printers. This ID must match the name of the APPL statement defined to VTAM in the SYS1.VTAMLST data set. This environment variable is required to use IP PrintWay extended mode to print on VTAM-controlled printers.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **applid** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- This environment variable does not apply to IP PrintWay basic mode.

Default: The environment variable is not set.

Example: AOP_APPLID=PWAPPL

AOP_BLANK_TRUNCATION_CLASSES

The list of JES output classes for which IP PrintWay extended mode removes blanks from the end of records or lines. If a data set is allocated in one of the listed classes, and the data set contains either line data or text data, IP PrintWay removes any blank characters that occur at the end of each record or line. If this environment variable is not set, IP PrintWay does not remove any blank characters.

Values: One or more letters or numbers. Specify multiple letters and numbers with no separation. An asterisk (*) indicates all JES output classes. Lowercase letters are converted to uppercase.

Tips:

- Specify this environment variable only if line or text data prints with extra blank characters at the end of lines.

- This environment variable can slow performance and cause IP PrintWay to use more system resources.
- Before giving data to IP PrintWay extended mode, JES always restores blanks that it truncated from data sets to save space on the JES spool. JES restores blanks regardless of how you set the JES2 BLNKTRNC and JES3 TRUNC parameters. Therefore, if you specify either BLNKTRNC=YES (JES2) or TRUNC=YES (JES3), you might need to specify this environment variable if your printed output contains unwanted blank characters. For example, if you specify BLNKTRNC=YES for output classes C and D, you could specify AOP_BLANK_TRUNCATION_CLASSES=CD to request that IP PrintWay extended mode remove trailing blank characters for output classes C and D. (BLNKTRNC=YES and TRUNC=YES are the JES defaults.)
For information about specifying JES blank truncation, see one of these:
 - *z/OS JES2 Initialization and Tuning Guide*
 - *z/OS JES3 Initialization and Tuning Guide*

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **blank-truncation-classes** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- This environment variable does not apply to IP PrintWay basic mode.

Default: The environment variable is not set.

Example: AOP_BLANK_TRUNCATION_CLASSES=ABC123

AOP_IGNORE_DCF_ROUTING_ERRORS

If this environment variable is set to YES, IP PrintWay extended mode does not write an error message (AOP3201E) when it cannot find a printer definition in the Printer Inventory that matches the DEST, CLASS, and FORMS parameters specified on the OUTPUT (or DD) JCL statement for a print job. Also, it does not save information about the print job in the Printer Inventory.

If this environment variable is not set, IP PrintWay extended mode writes an error message (AOP3201I) when it cannot find a printer definition that matches the DEST, CLASS, and FORMS parameters. Also, it saves information about the print job in the Printer Inventory so that Infoprint Central can display the print job and the associated AOP3201E message.

Tips:

- Set this environment variable to YES if your IP PrintWay job-selection rules cause IP PrintWay to select a large number of jobs whose DEST, CLASS, and FORMS parameters do not match printer definitions in the Printer Inventory. (In this situation, message AOP3201E occurs numerous times in the common message log.) Setting this environment variable to YES can conserve storage and processing resources.
- If you set this environment variable to YES, it might be difficult to determine that IP PrintWay did not process a print job because its CLASS, DEST, and FORMS parameters do not match a printer definition.
- The text of message AOP3201E is:
AOP3201E The DEST, CLASS, and FORMS JCL keywords do not match a printer definition in the Printer Inventory.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **ignore-dcf-routing-errors** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- This environment variable does not apply to IP PrintWay basic mode.

Default: The environment variable is not set.

Example: AOP_IGNORE_DCF_ROUTING_ERRORS=YES

AOP_MAIL_DO_NOT_ADD_SUFFIX

If this environment variable is set to YES, IP PrintWay does not append a suffix to any file names specified in the MAILFILE JCL parameter or the **mail-file-name** job attribute.

If this environment variable is not set, IP PrintWay appends one of these suffixes for the data formats it recognizes: afp, jpg, pcl, pdf, ps, sap, tif, txt, or octet-stream (for unrecognized data formats).

Tip: If the AOP_MAIL_DO_NOT_ADD_SUFFIX environment variable is set to YES, IP PrintWay ignores the AOP_MAIL_PRESERVE_SUFFIXES environment variable.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **mail-do-not-add-suffixes** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- If you run IP PrintWay basic mode, specify this environment variable in the IP PrintWay basic mode startup procedure.

Default: The environment variable is not set.

Example: AOP_MAIL_DO_NOT_ADD_SUFFIX=YES

AOP_MAIL_PRESERVE_SUFFIXES

The list of suffixes that IP PrintWay preserves in file names specified in the MAILFILE JCL parameter or in the **mail-file-name** job attribute. If the file name ends with one of the suffixes in this list, IP PrintWay does not append another suffix to the file name.

If no file name is specified in MAILFILE or **mail-file-name**, or if the suffix of the file name does not match a suffix in this environment variable, IP PrintWay appends one of these suffixes for the data formats it recognizes: afp, jpg, pcl, pdf, ps, sap, tif, txt, or octet-stream (for unrecognized data formats).

Tips:

- Specify the suffixes that users might specify in file names. For example, you might specify the .doc suffix for Microsoft Word documents.
- If the AOP_MAIL_DO_NOT_ADD_SUFFIX environment variable is set to YES, IP PrintWay ignores the AOP_MAIL_PRESERVE_SUFFIXES environment variable.

Rules:

- Begin each suffix with a period.

- To specify more than one suffix, separate the suffixes with a blank. Do not enclose the list of suffixes in single or double quotation marks.
- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **mail-preserve-suffixes** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- If you run IP PrintWay basic mode, specify this environment variable in the IP PrintWay startup procedure.

Default: The environment variable is not set.

Example: c

AOP_MAIL_USE_FIRST_ADDRESS

Specifies the email addresses that IP PrintWay extended mode uses when a print job (output group) contains more than one document (data set).

NO IP PrintWay sends each document to the email address or addresses specified for the document.

YES IP PrintWay sends each document to the email address or addresses specified for the first document in the print job.

Tips:

- AOP_MAIL_USE_FIRST_ADDRESS has no effect if the **Concatenate job** option is selected in the **Dataset grouping** field in the printer definition, or if the **Automatic dataset grouping** field is selected in the printer definition. When either concatenation option is selected, IP PrintWay extended mode sends all documents in a print job in one email to the email address or addresses specified for the first document in the print job.
- IP PrintWay basic mode ignores this environment variable. IP PrintWay basic mode always sends each document to the email address or addresses specified for the document unless the **Concatenate job** option is selected in the **Dataset grouping** field in the printer definition.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **mail-use-first-address** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- This environment variable does not apply to IP PrintWay basic mode.

Default: YES

Example: AOP_MAIL_USE_FIRST_ADDRESS=NO

AOP_SELECT_WORK_FROM_HOLD_QUEUE

Indicates whether IP PrintWay extended mode selects new held output groups from the JES spool.

Values:

YES IP PrintWay extended mode selects new held output groups from the JES spool. This lets you use Infoprint Central to move the held output groups to different IP PrintWay printers before releasing them.

NO IP PrintWay extended mode does not select new held output groups from the JES spool. This lets you use JES or SDSF commands to release individual data sets in an output group for printing.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **select-work-from-hold-queue** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart EXEC**.
- This environment variable does not apply to IP PrintWay basic mode.

Default: YES

Example: AOP_SELECT_WORK_FROM_HOLD_QUEUE = NO

AOP_SUPPRESS_POST_UNIX_FILTER_FORMATTING

Indicates whether IP PrintWay extended mode formats line or text data after a UNIX filter processes it. If this environment variable is set to any value, IP PrintWay extended mode does not format data when all these conditions are met:

- The data set to be processed contains either line data or text data.
- A UNIX filter is specified in the printer definition for the data format.
- The PAGEDEF or FORMDEF parameter is specified on the OUTPUT JCL statement for the data set. (A page or form definition specified in the printer definition does not meet this criterion.)

If this environment variable is not set, IP PrintWay extended mode always formats line or text data after a UNIX filter processes it. For example, IP PrintWay processes carriage control characters and transparent data characters, and converts the data from the document code page to the printer code page.

Tips:

- Specify this environment variable if you use a UNIX filter that calls an AFP transform to convert line data to PCL, PDF, or PostScript format.
- This environment variable does not affect how IP PrintWay extended mode processes data when you specify a DLL filter in the printer definition. IP PrintWay extended mode does not modify data returned by DLL filters.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **suppress-post-unix-filter-formatting** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart EXEC**.
- This environment variable does not apply to IP PrintWay basic mode.

Default: IP PrintWay extended mode formats line or text data after a UNIX filter processes it.

Example:

AOP_SUPPRESS_POST_UNIX_FILTER_FORMATTING=YES

AOPMAILER

Specifies the full path name of the z/OS UNIX **sendmail** command that IP

PrintWay uses to send output to email destinations. This environment variable is optional. However, you must set it if you did not install the **sendmail** executable file in the default **/bin** directory.

Rules:

- Start the path name with a slash (/).
- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **mailer-path-name** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- If you run IP PrintWay basic mode, specify this environment variable in the IP PrintWay basic mode startup procedure.

Default: /bin/sendmail

Example: AOPMAILER=/mydirectory/sendmail

AOPMAILER_OPTIONS

Options for the z/OS UNIX **sendmail** command. IP PrintWay specifies these options in addition to the **-i** option when it sends emails. You can specify this environment variable regardless of whether you specify the AOPMAILER environment variable. This environment variable is optional.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **mailer-options** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- If you run IP PrintWay basic mode, specify this environment variable in the IP PrintWay basic mode startup procedure.

Default: The environment variable is not set.

Example: AOPMAILER_OPTIONS=-Am

IP PrintWay extended mode also uses the LANG, LIBPATH, NLSPATH, and PATH environment variables.

Example: This example shows how to specify an environment variable in the **aopstart** EXEC:

```
/*-----  
/* The following environment variables are examples of other variable  
/* which may need to be set for your installation.  
/*-----  
n=n+1;env.n='STEPLIB=SYS1.IMAGELIB:hlq.EXITLIB'
```

Tips for migrating from IP PrintWay basic mode:

- If the STEPLIB JCL statement in the IP PrintWay basic mode startup procedure specifies a library that contains IP PrintWay exits, specify the STEPLIB environment variable in the **aopstart** EXEC. Also, add the library to the file specified by the STEPLIBLIST statement in the BPXPRMxx PARMLIB member.
- If the STDENV data definition (DD) in the IP PrintWay basic mode startup procedure contains environment variables, specify the same environment variables in the **aopstart** EXEC or, if dynamic configuration is enabled, in the system configuration definition.

- If the **Applid** field in an IP PrintWay FSS definition in the Printer Inventory contains an application program ID, specify the AOP_APPLID environment variable in the **aopstart** EXEC or, if dynamic configuration is enabled, specify the **applid** attribute in the system configuration definition. You can use the same application program ID for both IP PrintWay basic mode and extended mode if you do not run both modes at the same time.

Related information: For information about:

- How to edit the **aopstart** EXEC, see “Editing the aopstart EXEC” on page 57
- The LANG, LIBPATH, NLSPATH, and PATH variables, see “Setting environment variables for the Printer Inventory Manager” on page 43
- For information about how to access the Language Environment runtime libraries and improve the performance of runtime libraries, see *z/OS UNIX System Services Planning*.

Editing the system configuration definition for IP PrintWay (extended mode)

The system configuration definition in the Printer Inventory contains attributes that let you customize IP PrintWay extended mode while it is running. If dynamic configuration is enabled, you can specify these attributes in the system configuration definition to customize IP PrintWay extended mode:

allow-all-characters-in-line-data = yes | no

This attribute determines how Infoprint Server detects the data format of documents that contain line data. Valid line data contains carriage control characters, table reference characters, binary zeroes, and printable characters. However some applications might create line data that contains unprintable characters.

yes Infoprint Server detects the data format as “line data” even if the input data contains unprintable characters.

no Infoprint Server detects the data format as “other” if the input data contains unprintable characters. Line data that Infoprint Server detects as “other” might not print correctly because Infoprint Server does not convert it to the printer's code page.

Tips:

- Job submitters can specify the data format in the **document-format** job attribute. For example, job submitters can specify **document-format=line** in the PRTATTRS parameter on the OUTPUT JCL statement. If the **document-format** job attribute is specified, Infoprint Server ignores this attribute because it does not detect the data format.
- Specify **allow-all-characters-in-line-data=yes** if Infoprint Server does not detect the line data format and job submitters cannot specify the **document-format=line** job attribute.
- This attribute does not affect how NetSpool checks SCS and 3270 data streams for validity. The input SCS and 3270 data streams must contain only valid characters.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.

- If dynamic configuration is not enabled, specify the equivalent `AOP_ALLOW_ALL_CHARACTERS_IN_LINE_DATA` environment variable in the `aopstart EXEC`.
- This attribute does not apply to IP PrintWay basic mode.

Default: `allow-all-characters-in-line-data = no`

Example: `allow-all-characters-in-line-data = yes`

applid = applid

The application program ID that IP PrintWay extended mode uses to establish a VTAM session with printers. This ID must match the name of the APPL statement defined to VTAM in the SYS1.VTAMLST data set. This attribute is required to use IP PrintWay extended mode to print on VTAM-controlled printers.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent `AOP_APPLID` environment variable in the `aopstart EXEC`.
- This attribute does not apply to IP PrintWay basic mode.

Default: None

Example: `applid = PWAPPL`

blank-truncation-classes = classes

The list of JES output classes for which IP PrintWay extended mode removes blanks from the end of records or lines. If a data set is allocated in one of the listed classes, and the data set contains either line data or text data, IP PrintWay removes any blank characters that occur at the end of each record or line. If this attribute is not specified, IP PrintWay extended mode does not remove any blank characters.

Values: One or more letters or numbers. Specify multiple letters and numbers with no separation. An asterisk (*) indicates all JES output classes. Lowercase letters are converted to uppercase.

Tips:

- Specify this attribute only if line or text data prints with extra blank characters at the end of lines.
- This attribute can slow performance and cause IP PrintWay to use more system resources.
- Before giving data to IP PrintWay extended mode, JES always restores blanks that it truncated from data sets to save space on the JES spool. JES restores blanks regardless of how you set the JES2 `BLNKTRNC` and JES3 `TRUNC` parameters. Therefore, if you specify either `BLNKTRNC=YES` (JES2) or `TRUNC=YES` (JES3), you might need to specify this attribute if your printed output contains unwanted blank characters. For example, if you specify `BLNKTRNC=YES` for output classes C and D, you could specify `AOP_BLANK_TRUNCATION_CLASSES=CD` to request that IP PrintWay extended mode remove trailing blank characters for output classes C and D. (`BLNKTRNC=YES` and `TRUNC=YES` are the JES defaults.)

For information about specifying JES blank truncation, see one of these:

- *z/OS JES2 Initialization and Tuning Guide*

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOP_BLANK_TRUNCATION_CLASSES environment variable in the **aopstart** EXEC.
- This attribute does not apply to IP PrintWay basic mode.

Default: None.

Example: blank-truncation-classes = ABC123

ignore-dcf-routing-errors = yes | no

Indicates whether IP PrintWay extended mode writes an error message when it cannot find a printer definition in the Printer Inventory that matches the specified DEST, CLASS, and FORMS JCL parameters:

- yes** IP PrintWay extended mode does not write an error message (AOP3201E) when it cannot find a printer definition in the Printer Inventory that matches the DEST, CLASS, and FORMS parameters specified on the OUTPUT (or DD) JCL statement for a print job. Also, it does not save information about the print job in the Printer Inventory.
- no** IP PrintWay extended mode writes an error message (AOP3201I) when it cannot find a printer definition that matches the DEST, CLASS, and FORMS parameters. Also, it saves information about the print job in the Printer Inventory so that Infoprint Central can display the print job and the associated AOP3201E message.

Tips:

- Set **ignore-dcf-routing-errors = yes** if your IP PrintWay job-selection rules cause IP PrintWay to select a large number of jobs whose DEST, CLASS, and FORMS parameters do not match printer definitions in the Printer Inventory. (In this situation, message AOP3201E occurs numerous times in the common message log.) Specifying **ignore-dcf-routing-errors = yes** can conserve storage and processing resources.
- If you set **ignore-dcf-routing-errors = yes**, it might be difficult to determine that IP PrintWay did not process a print job because its CLASS, DEST, and FORMS parameters do not match a printer definition.
- The text of message AOP3201E is:
AOP3201E The DEST, CLASS, and FORMS JCL keywords do not match a printer definition in the Printer Inventory.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOP_IGNORE_DCF_ROUTING_ERRORS environment variable in the **aopstart** EXEC.
- This attribute does not apply to IP PrintWay basic mode.

Default: ignore-dcf-routing-errors = no

Example: ignore-dcf-routing-errors = yes

mail-do-not-add-suffixes = yes | no

Indicates whether IP PrintWay extended mode appends a suffix to file names specified in the MAILFILE JCL parameter or the **mail-file-name** job attribute:

yes IP PrintWay does not append a suffix.

no IP PrintWay appends one of these suffixes for the data formats: afp, jpg, pcl, pdf, ps, sap, tif, txt, or octet-stream (for unrecognized data formats).

Tip: If **mail-do-not-add-suffixes = yes**, IP PrintWay ignores the **mail-preserve-suffixes** attribute.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOP_MAIL_DO_NOT_ADD_SUFFIX environment variable in the **aopstart** EXEC.
- If you run IP PrintWay basic mode, specify the equivalent AOP_MAIL_DO_NOT_ADD_SUFFIX environment variable in the IP PrintWay startup procedure.

Default: mail-do-not-add-suffixes = no

Example: mail-do-not-add-suffixes = yes

mail-preserve-suffixes = suffixes

The suffixes that IP PrintWay extended mode preserves in file names specified in the MAILFILE JCL parameter or in the **mail-file-name** job attribute. If the file name ends with one of the suffixes in this list, IP PrintWay does not append another suffix to the file name.

If no file name is specified in MAILFILE or **mail-file-name**, or if the suffix of the file name does not match a suffix in this attribute, IP PrintWay appends one of these suffixes for the data formats it recognizes: afp, jpg, pcl, pdf, ps, sap, tif, txt, or octet-stream (for unrecognized data formats).

Tips:

- Specify the suffixes that users might specify in file names. For example, you might specify the .doc suffix for Microsoft Word documents.
- If the **mail-do-not-add-suffixes = yes**, IP PrintWay ignores the **mail-preserve-suffixes** attribute.

Rules:

- Begin each suffix with a period.
- To specify more than one suffix, separate the suffixes with a blank and enclose the suffixes in single or double quotation marks.
- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOP_MAIL_PRESERVE_SUFFIXES environment variable in the **aopstart** EXEC.
- If you run IP PrintWay basic mode, specify the equivalent AOP_MAIL_PRESERVE_SUFFIXES environment variable in the IP PrintWay startup procedure.

Default: None.

Example: mail-preserve-suffixes = ".123 .xls .doc"

mail-use-first-address = yes | no

When the print job contains more than one document, indicates whether IP PrintWay extended mode sends each document to the email address or addresses specified for the first document in the print job.

yes IP PrintWay sends each document to the email address or addresses specified for the first document in the print job.

no IP PrintWay sends each document to the email address or addresses specified for that document.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOP_MAIL_USE_FIRST_ADDRESS environment variable in the **aopstart** EXEC.
- This attribute does not apply to IP PrintWay basic mode.

Tip: This attribute has no effect if the **Concatenate job** option is selected in the **Dataset grouping** field in the printer definition, or if the **Automatic dataset grouping** field is selected in the printer definition. When either concatenation option is selected, IP PrintWay extended mode sends all documents in a print job in one email to the email address or addresses specified for the first document in the print job.

Default: mail-use-first-address = yes

Example: mail-use-first-address = no

mailer-options = *options*

Options for the z/OS UNIX **sendmail** command. IP PrintWay extended mode specifies these options in addition to the **-i** option when it sends emails. You can specify this attribute regardless of whether you specify the **mailer-path-name** attribute. This attribute is optional.

If you specify more than one option, separate the options with a blank and enclose the entire value in single or double quotation marks.

Rules:

- To specify more than one option, separate the options with a blank and enclose the options in single or double quotation marks.
- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOPMAILER_OPTIONS environment variable in the **aopstart** EXEC.
- If you run IP PrintWay basic mode, specify the equivalent AOPMAILER_OPTIONS environment variable in the IP PrintWay startup procedure.

Default: None

Example: mailer-options = -Am

mailer-path-name = *pathname*

The full path name of the z/OS UNIX **sendmail** command that IP PrintWay extended mode uses to send output to email destinations. This attribute is optional. However, you must specify it if you did not install the **sendmail** executable file in the default directory, **/bin**.

Rules:

- Start the path name with a slash (/).
- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOPMAILER environment variable in the **aopstart EXEC**.
- If you run IP PrintWay basic mode, specify the equivalent AOPMAILER environment variable in the IP PrintWay startup procedure.

Default: mailer-path-name=/bin/sendmail

Example: mailer-path-name=/mydirectory/sendmail

select-work-from-hold-queue = **yes** | **no**

Indicates whether IP PrintWay extended mode selects new held output groups from the JES spool.

- yes** IP PrintWay extended mode selects new held output groups from the JES spool. This lets you use Infoprint Central to move the held output groups to different IP PrintWay printers before releasing them.
- no** IP PrintWay extended mode does not select new held output groups from the JES spool. This lets you use JES or SDSF commands to release individual data sets in an output group for printing.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute while IP PrintWay is running, stop the IP PrintWay daemons (**aopoutd** and **aopwsmd**) and restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOP_SELECT_WORK_FROM_HOLD_QUEUE environment variable in the **aopstart EXEC**.
- This attribute does not apply to IP PrintWay basic mode.

Default: select-work-from-hold-queue = yes

Example: select-work-from-hold-queue = no

smf-recording = **yes** | **no**

Indicates whether IP PrintWay extended mode writes SMF type 6 records for data sets that it sends to a printer or email destination. SMF type 6 records contain accounting information.

- yes** IP PrintWay extended mode writes SMF type 6 records.
- no** IP PrintWay extended mode does not write SMF type 6 records.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.
- This attribute does not apply to IP PrintWay basic mode.

Default: smf-recording = yes

Example: smf-recording = no

suppress-post-unix-filter-formatting = yes | no

Indicates whether IP PrintWay extended mode formats line or text data after a UNIX filter processes it:

- yes** IP PrintWay extended mode does not format data when all these conditions are met:
- The data set to be processed contains either line data or text data.
 - A UNIX filter is specified in the printer definition for the data format.
 - The PAGEDEF or FORMDEF parameter is specified on the OUTPUT JCL statement for the data set. (A page or form definition specified in the printer definition does not meet this criterion.)
- no** IP PrintWay extended mode always formats line or text data after a UNIX filter processes it. For example, IP PrintWay processes carriage control characters and transparent data characters, and converts the data from the document code page to the printer code page.

Tips:

- Specify **suppress-post-unix-filter-formatting=yes** if you use a UNIX filter that calls an AFP transform to convert line data to PCL, PDF, or PostScript format.
- This attribute does not affect how IP PrintWay extended mode processes data when you specify a DLL filter in the printer definition. IP PrintWay extended mode does not modify data returned by DLL filters.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent `AOP_SUPPRESS_POST_UNIX_FILTER_FORMATTING` environment variable in the **aopstart** EXEC.
- This attribute does not apply to IP PrintWay basic mode.

Default: suppress-post-unix-filter-formatting = no

Example: suppress-post-unix-filter-formatting = yes

Example: This screen shows a portion of the ISPF panel that you use to edit the system configuration definition for IP PrintWay extended mode:

Infoprint Server: System Configuration

```
General:
_ Allow all characters in line data

IP PrintWay (extended mode):
_ Ignore DEST, CLASS, FORMS routing errors
 / Select work from hold queue
_ Suppress formatting after UNIX filter
 / Write SMF records
APPL ID . . . . . _____
Blank truncation classes . . . . . _____
                                     -----
Email:
 / Send email to address of first document
_ Do not add suffixes to files
Preserve suffixes . . . . . _____ (more)
Application path . . . . . /bin/sendmail (extend)
Application options . . . . . _____ (extend)
```

Related information: For information about:

- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 68
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 80

APF-authorizing SYS1.IMAGELIB (extended mode)

You must APF-authorize the SYS1.IMAGELIB data set if IP PrintWay extended mode uses forms control buffers (FCBs) to process any documents. If SYS1.IMAGELIB is in the system LNKLIST, you can APF-authorize the system LNKLIST. If SYS1.IMAGLIB is not in the system LNKLIST, or if you do not APF-authorize the system LNKLIST, you must APF-authorize SYS1.IMAGELIB.

To temporarily APF-authorize SYS1.IMAGELIB, use the SETPROG APF system command. For more information, see *z/OS MVS System Commands*.

To permanently authorize SYS1.IMAGELIB, add the data set to the APF statement in the PROGxx member of SYS1.PARMLIB. For more information, see *z/OS MVS Initialization and Tuning Reference*.

Note: If SYS1.IMAGELIB is not in the system LNKLIST, you must specify SYS1.IMAGELIB in the STEPLIB environment variable in the **aopstart** EXEC and in the STEPLIBLIST sanction list in the BPXPRMxx PARMLIB member. For information, see:

- “Setting environment variables for IP PrintWay (extended mode)” on page 215
- “Updating the BPXPRMxx member of SYS1.PARMLIB (extended mode)” on page 207.

Customizing from AFP transforms for IP PrintWay (extended mode)

If the administrator selects one of the Printer Job Control (PJM) options in the printer definition (**Restart printing after last successful page** and **Record pages printed successfully**), that printer definition cannot be used to print documents that already contain PJL JOB commands. This is because a PJL JOB command in

the document would conflict with the PJJ JOB command that IP PrintWay adds to the document to track the number of pages that print successfully.

If you use an IBM transform that transforms AFP to PCL or PostScript, you can customize the transform to not add a PJJ JOB command to the output data stream. For information about how to customize the transform, see the documentation for the transform.

If a document contains the PJJ JOB command, IP PrintWay extended mode cannot track the number of pages that print. It writes an error message (AOP3859E) and does not print the document.

Related information: For information about:

- IBM transforms that add PJJ JOB commands, see *IBM Print Transforms from AFP for Infoprint Server for z/OS*.
- PJJ options in the printer definition, see *z/OS Infoprint Server Operation and Administration*.

Defining printer host names in the DNS

IP PrintWay uses the domain name server (DNS) to do these lookups:

- Look up the IP address for a printer's host name (basic and extended modes)
- Look up the host name for a printer's IP address (extended mode)

Follow these rules when you define printer host names in the DNS:

- 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.
- If you specify a printer's host name to IP PrintWay, map the printer's host name to an IP address.
- **IP PrintWay extended mode:** If you specify a printer's host name to IP PrintWay extended mode, map the printer's host name to the IP address and also map the printer's IP address to the host name.
- **IP PrintWay extended mode:** If you specify a printer's IP address to IP PrintWay extended mode, you do not need to map the IP address to a host name. However, if you do map the printer's IP address to a host name, also map the printer's host name to the IP address.
- **IP PrintWay extended mode:** Define a printer's host name in the DNS before the administrator creates a printer definition for the printer.

If the administrator specifies a host name in a printer definition before the host name is defined in the DNS, Infoprint Central cannot display the printer until someone submits a print job or the administrator changes a field in the printer definition.

Tip: If you run IP PrintWay extended mode and do not define printer host names for any printers in the DNS, consider specifying the **resolve-printway-printers=no** attribute in the Infoprint Server configuration file. This limits the number of DNS lookups that IP PrintWay does. For information, see “Editing the **aopd.conf** file for IP PrintWay (extended mode)” on page 205.

Customizing z/OS Communications Server

To print on TCP/IP-attached printers or to send output to email destinations, you must customize the TCP/IP component of z/OS Communications Server. For information about how to customize TCP/IP, see *z/OS Communications Server: IP Configuration Guide* and *z/OS Communications Server: IP Configuration Reference*.

This table describes tasks related to TCP/IP customization. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies.

Task	Condition	See page
Customizing the TCPIP.DATA data set	Optional: To specify the TCP/IP name or high-level qualifier	232
Customizing the PROFILE.TCPIP data set	Optional: To change TCP/IP configuration	233
Customizing z/OS UNIX sendmail	Optional: To send output to email destinations	234

Customizing the TCPIP.DATA data set

The TCP/IP TCPIP.DATA data set contains configuration information for TCP/IP. This section describes only those statements that have special consideration for IP PrintWay basic mode or IP PrintWay extended mode. For information about the TCPIP.DATA data set, see *z/OS Communications Server: IP Configuration Reference*.

This TCPIP.DATA statement primarily affects IP PrintWay extended mode:

Statement	Meaning
LOOKUP	Specifies the order in which TCP/IP uses the DNS and the local host files to resolve host names. Guideline: To significantly improve performance when you run IP PrintWay extended mode, specify: LOOKUP DNS

These TCPIP.DATA statements affect only IP PrintWay basic mode:

Statement	Meaning
TCPIPJOBNAME	Specifies the name of the TCP/IP program that IP PrintWay basic mode uses. You can specify the TCP/IP program name either in this statement or in the IP PrintWay 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. Limitation: IP PrintWay cannot interpret symbolic TCP/IP names. Symbolic names include an ampersand (&). Therefore, if the TCPIPJOBNAME statement specifies a symbolic TCP/IP name, do one of these:

- Specify the TCP/IP name in the **TCP/IP job name** field in the IP PrintWay FSS definition in the Printer Inventory. IP PrintWay basic mode uses the TCP/IP name in the **TCP/IP job name** field instead of the TCP/IP name in the TCPIPJOBNAME statement. However, each IP PrintWay FSS and FSA writes informational message ANFM016I to the console because the TCPIPJOBNAME statement contains a symbolic name:

```
ANFM016I Invalid TCP/IP jobname
```

Operators can ignore this message because it does not cause an error.

- Create a separate TCPIP.DATA data set for IP PrintWay basic mode to use. In this TCPIP.DATA data set, replace the symbolic TCP/IP job name with the actual TCP/IP job name. Specify this TCPIP.DATA data set in the SYSTCPD DD statement in the IP PrintWay startup procedures. In this case, IP PrintWay does not write message ANFM016I to the console.

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.

Tip: If you change any of the values in the TCPIP.DATA data set, you must restart TCP/IP so that your changes take effect.

IP PrintWay basic mode: You can create a separate TCPIP.DATA data set for use only 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 sequence:

1. The data set that the SYSTCPD DD statement refers to in the IP PrintWay basic mode startup procedure
2. *jobname*.TCPIP.DATA
jobname is the job name of the IP PrintWay basic mode startup procedure.
3. SYS1.TCPPARMS(TCPDATA)
4. *tcphlq*.TCPIP.DATA
tcphlq 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

When you customize TCP/IP, you can customize the PROFILE.TCPIP data set. For information about how to customize this data set for IP PrintWay and also for other components of Infoprint Server, see “Customizing z/OS Communications Server” on page 117.

Customizing z/OS UNIX sendmail

When the email protocol is selected in a printer definition, IP PrintWay uses z/OS UNIX sendmail to prepare and send emails 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 email 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 publication *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.
- The Sendmail Consortium website at <http://www.sendmail.org>.

Guidelines: Follow these guidelines when you 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 email could not be delivered to a remote system and how long sendmail waits before bouncing email. Bounced email is undeliverable email.
- To use sendmail aliases or to receive messages from sendmail about bounced emails, start sendmail as a daemon. 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.

Example:

```
su
/usr/sbin/sendmail -bd -q1m &
exit
```

The **su** command switches to an effective UID of 0. To use **su**, you must be permitted to the BPX.SUPERUSER profile in the FACILITY class in RACF.

Tip: To start sendmail 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 email 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 emails. If you do not specify the user ID that owns the alias name, sendmail sends notification of bounced emails 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 emails. Also if you want to receive replies from emails or sendmail notifications about undeliverable emails, configure your z/OS system so that it can receive emails from outside the firewall.
- If you do not install the **sendmail** command in the default directory, **/bin**, specify the full path name of the **sendmail** command in one of these ways:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the **mailer-path-name** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the AOPMAILER environment variable in the **aopstart EXEC**.
- If you run IP PrintWay basic mode, specify the AOPMAILER environment variable in the IP PrintWay startup procedure.
- If you need to specify **sendmail** command options, specify the options in one of these ways:
 - If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the **mailer-options** attribute in the system configuration definition.
 - If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the AOPMAILER_OPTIONS environment variable in the **aopstart EXEC**.
 - If you run IP PrintWay basic mode, specify the AOPMAILER_OPTIONS environment variable in the IP PrintWay startup procedure.

IP PrintWay specifies the options that you specify in addition to the **-i** option when it sends emails.

Tip: To verify that you have configured sendmail correctly, run the **sendmail** command directly from the z/OS UNIX command line. The **sendmail** command is described in *z/OS Communications Server: IP User's Guide and Commands*. Some tests that you can do are:

- **Send a file to a local user:** Run this command to send file `myfile` to yourself:

```
sendmail youruserID <myfile
```

Result: The email should be delivered to you and you should receive no error messages from sendmail. To view the mail sent to your user ID, run the z/OS UNIX **mail** command:

```
mail
```

- **Send a file to a remote address:** To verify that firewalls do not prevent sending emails, send a file to a remote user on a domain that is not in your installation:

```
sendmail username@domain <myfile
```

Result: The email should be delivered to the remote address and you should receive no error messages from sendmail.

- **Send a file to an alias name:** To verify that the sendmail aliases file is configured correctly, send a file to a sendmail alias name:
 1. To edit the aliases file, use the **su** command to switch to an effective UID of 0. Then edit the file using your favorite editor and save your changes:

```
su
oedit /etc/mail/aliases
```

To use the **su** command, you must be permitted to the BPX.SUPERUSER profile in the FACILITY class in RACF.

2. To create alias `testlist`, type these lines in the sendmail aliases file, **/etc/mail/aliases**:

```
testlist: youruserID
owner-testlist: youruserID
```

3. Run the sendmail **newaliases** command to update the sendmail aliases file:

```
/usr/sbin/newaliases
```

4. If you switched to an effective UID of 0 in step 1, switch back to your own UID:

```
exit
```

5. Run the **sendmail** command to send file myfile to alias name testlist:

```
sendmail testlist <myfile
```

Result: The email should be delivered to you and you should receive no error messages from sendmail.

Related information: For information about:

- How to edit the system configuration definition, see “Editing the system configuration definition for IP PrintWay (extended mode)” on page 223
- How to specify environment variables for IP PrintWay extended mode, see “Setting environment variables for IP PrintWay (extended mode)” on page 215
- How to specify environment variables for IP PrintWay basic mode, see “Customizing the IP PrintWay startup procedure (basic mode)” on page 320

Creating a VTAM APPL resource definition for IP PrintWay

When IP PrintWay prints on VTAM-controlled printers, it runs as a VTAM application program. Therefore, to print on VTAM-controlled printers, you must create a VTAM APPL statement in the SYS1.VTAMLST data set.

If you run IP PrintWay basic mode, create an APPL statement with a unique APPL ID for each IP PrintWay FSS defined to JES. If IP PrintWay runs on more than one system in a sysplex, the APPL IDs for all systems in the sysplex must be unique.

If you run IP PrintWay extended mode, create one APPL statement for each z/OS system. If IP PrintWay runs on more than one system in a sysplex, the APPL ID for each system must be unique. If you have already created an APPL statement for IP PrintWay basic mode, you can use the same APPL statement for IP PrintWay extended mode if you do not run IP PrintWay basic mode and IP PrintWay extended mode at the same time.

Create the APPL statement in a new or existing application major node definition. Figures Figure 21 and Figure 22 on page 237 show sample APPL statements created under a major node named PRINTWAY.

```
PRINTWAY VBUILD TYPE=APPL Application major node
*
PWAPPL APPL AUTH=ACQ APPL for IP PrintWay extended
```

Figure 21. Sample APPL statement for IP PrintWay extended mode

```

PRINTWAY VBUILD TYPE=APPL Application major node
*
PWAPPL01 APPL AUTH=ACQ APPL for an IP PrintWay basic mode FSA
PWAPPL02 APPL AUTH=ACQ APPL for an IP PrintWay basic mode FSA

```

Figure 22. Sample APPL statement for IP PrintWay basic mode

AUTH=ACQ

This parameter lets IP PrintWay initiate a session with a printer. AUTH=ACQ is required.

For complete information about the APPL statement and its parameters, see *z/OS Communications Server: SNA Resource Definition Reference*.

Note: After you create the APPL statement, specify the name of the statement in one of these locations:

- **IP PrintWay basic mode:** Specify the application program ID in the **Applid** field of the IP PrintWay FSS definition in the Printer Inventory. For information, see “Creating FSS and FSA definitions in the Printer Inventory (basic mode)” on page 312.
- **IP PrintWay extended mode:** If dynamic configuration is enabled, specify the application program ID in the **applid** attribute in the system configuration definition. If dynamic configuration is not enabled, specify the application program ID in the AOP_APPLID environment variable in the **aopstart** EXEC.

Writing IP PrintWay exit and filter programs

Note

This section contains programming-interface information.

These optional exits let you customize IP PrintWay processing:

- Routing exit

The Routing exit lets you inspect and change routing parameters for each data set before the data set is transmitted. The Routing exit lets you change the printer or email address and some options. You cannot change the transmission protocol that IP PrintWay uses (LPR, IPP, direct sockets, VTAM, or email). IP PrintWay basic mode also lets you hold the data set on the JES spool, or delete the data set from the JES spool. For more information, see “Routing exit (ANFUXRTG)” on page 247.

Note: The email addresses specified in an email header override email addresses specified in any other location, including any email addresses specified in the Routing exit.

- Begin Data Set exit

The Begin Data Set exit lets you inspect routing parameters and transmission options for a data set before the data set is transmitted. You can change most transmission options, add a document header and document trailer, and add one or more records to the beginning of the data set. For example, you could add a separator page. For more information, see “Begin Data Set exit” on page 258.

- Record exit

The Record exit lets you add, replace, or delete records in a data set before the data set is transmitted. For more information, see “Record exit” on page 269.

- End Data Set exit

The End Data Set exit lets you add records to the end of a data set before the data set is transmitted. For example, you could add a separator page. For more information, see “End Data Set exit” on page 276.

- SMF exit

The SMF exit lets you suppress or replace the System Management Facilities (SMF) type 6 record that IP PrintWay writes for each data set that has been transmitted. For more information, see “SMF exit (ANFUXSMF)” on page 285.

- Message exit (basic mode)

The Message exit lets you suppress or change the text of any message that IP PrintWay writes to the IP PrintWay message-log data set. The Message exit applies only to IP PrintWay basic mode. For more information, see “Message exit (ANFUXMSG) — basic mode” on page 290.

- Response Notification exit

The Response Notification exit lets you do an action based on the success or failure of the transmission as indicated by the Response Notification code. For more information, see “Response Notification exit (ANFUXRSP)” on page 292.

IP PrintWay calls exits in this order:

1. Routing exit
2. Begin Data Set exit
3. Record exit
4. End Data Set exit
5. Response Notification exit
6. SMF exit

IP PrintWay basic mode calls the Message exit before writing any messages to the IP PrintWay message-log data set.

In addition to writing IP PrintWay exits, you can 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 NetSpool also use it. However, a filter program is called only once for any data stream. If you write a filter program, and you run IP PrintWay basic mode, you must select the **Resubmit for filtering** option in the printer definition.

Tip: You can specify only one filter for each data format in the printer definition. Therefore, if you write your own filter, you cannot also specify a transform filter in the printer definition.

Related information: For information about how to write filter programs, see “Writing a filter program” on page 121.

Comparing IP PrintWay basic mode and extended mode exits

These differences exist between IP PrintWay basic mode and extended mode exits:

- If an IP PrintWay exit is in an APF-authorized step library instead of in the system LNKLST:

- IP PrintWay basic mode: Specify the step library in the STEPLIB DD statement in the IP PrintWay startup procedure. For information, see “Editing the ANFWPROC startup procedure (basic mode)” on page 321.
- IP PrintWay extended mode: Specify the step library in the STEPLIB environment variable in the **aopstart** EXEC. For information, see “Setting environment variables for IP PrintWay (extended mode)” on page 215. Also, specify the step library in the file specified in the STEPLIBLIST statement in the BPXPRMxx PARMLIB member. For information, see *z/OS UNIX System Services Planning*.
- IP PrintWay exits run in different states:
 - IP PrintWay basic mode: Exits run in supervisor state with a protection key of 1.
 - IP PrintWay extended mode: Exits run in problem state with a protection key of 8.
- If an exit sets flag XTPM2CON to write a message to the console:
 - IP PrintWay basic mode: The message is written only to the console.
 - IP PrintWay extended mode: The message is written to both the console and the common message log.
- If an exit sets flag XTPM2MDS to write a message to a message data set:
 - IP PrintWay basic mode: The message is written only to the IP PrintWay message-log data set.
 - IP PrintWay extended mode: The message is written to both the message-log data set and the common message log.
- IP PrintWay extended mode does not set these fields in the ANFUEXTP control block:
 - XTP_FILTERING: IP PrintWay extended mode always sets this flag to zero because the resubmit for filtering option applies only to IP PrintWay basic mode.
 - XTPDSJNW: IP PrintWay extended mode always sets this flag to zero. (IP PrintWay basic mode sets this flag to indicate the data set being processed is the JESNEWS data set.)
- IP PrintWay passes the name of the z/OS system where the data set originated to the exits in different ways:
 - IP PrintWay basic mode: The system name is in the JMR control block. The address of the JMR is in the IAZJSPA control block.
 - IP PrintWay extended mode: No JMR is provided. Instead, the ANFUEXTP control block contains these fields: XTP_ORIGINATING_MVS_SYSTEM and XTP_ORIGINATING_JES2_SYSTEM. Your exit can check the XTPVERS field to determine whether these fields are set. IP PrintWay extended mode sets the XTPVERS field to 1. IP PrintWay basic mode sets the XTPVERS field to 0.
- IP PrintWay passes job separator information to the exits in different ways:
 - IP PrintWay basic mode: Field XTPJSPAP in the ANFUEXTP control block contains a pointer to the JES job separator page data area (IAZJSPA) for the data set being processed.
 - IP PrintWay extended mode: Field XTPJSPAP contains a pointer to a data area that is the same as the JES data area except for some fields. For more information, see “JES job separator page data area (IAZJSPA)” on page 243.
- In IP PrintWay extended mode, the Routing exit cannot share work areas with other exits. For more information about work areas, see “Programming considerations” on page 242.

- **Begin Data Set exit:** IP PrintWay extended mode stores the maximum line length in field XTPMLINL in the ANFUEXTP control block. IP PrintWay basic mode always sets field XTPMLINL to zeroes.
- **Routing exit:** IP PrintWay supports different functions in this exit:
 - IP PrintWay basic mode: The Routing exit can hold or delete the print job.
 - IP PrintWay extended mode: The Routing exit cannot hold or delete the print job. IP PrintWay extended mode ignores flags XTPDSHLD and XTPDSTRM in the ANFUEXTP control block.
- **Routing exit:** IP PrintWay processes this exit in different ways for data sets in the same print job (JES output group):
 - IP PrintWay basic mode: The Routing exit can reroute each data set in the same output group to a separate printer. Also, the Routing exit can specify different printing options for each data set in the same output group.
 - IP PrintWay extended mode: The Routing exit cannot reroute each data set in the same output group to a separate printer. The Routing exit can reroute the first data set in an output group. IP PrintWay reroutes all data sets in the output group to the same printer. Also, the Routing exit cannot specify different retry and retention times for each data set in the same output group. However, the exit can specify a different number of copies (XTPCOPYS field in the ANFUEXTP control block) and different print options (XTPOPTNM field) for each data set in the output group.
- **SMF exit:** IP PrintWay extended mode writes information in new fields in the SMF type 6 record. Therefore, you must recompile your SMF exit when you migrate to IP PrintWay extended mode.

For example, IP PrintWay does not write the IP address of the printer in fields SMF6IP1 through SMF6IP4. These fields contain zeroes when you run IP PrintWay extended mode. For information about the SMF type 6 record that IP PrintWay extended mode writes, see *z/OS MVS System Management Facilities (SMF)*.
- **Message exit:** IP PrintWay extended mode does not support the Message exit.

Tip: If the Message exit writes IP PrintWay basic mode messages to the system console, you can instead use the Infoprint Server hardcopy log function. The hardcopy log function writes all or selected Infoprint Server messages, including IP PrintWay extended mode messages, to the z/OS hardcopy log. For information about how to use this function, see “Sending non-console messages to the hardcopy log” on page 84.

Resubmit for filtering considerations (basic mode)

For a detailed description of IP PrintWay basic mode and Print Interface processing when the **resubmit for filtering** option is selected in a printer definition, see *z/OS Infoprint Server Operation and Administration*. When the **resubmit for filtering** option is selected, IP PrintWay basic mode processes data two times, once when it selects the original data set from the JES spool and a second time when it selects the data set that Print Interface has allocated on the JES spool.

When the **resubmit for filtering** option is selected, IP PrintWay calls the Response Notification and SMF exits once for each data set. However, IP PrintWay calls the Routing, Begin Data Set, Record, and End Data Set exits only for the second data set. When the Routing, Begin Data Set, Record, and End Data Set exits are called, Print Interface has transformed data into the new data format.

When IP PrintWay calls the Routing, Begin Data Set, Record, and End Data Set exits, some fields in the ANFUEXTP control block no longer contain the values

originally specified by the job submitter in JCL parameters. Instead, these fields contain values specified in the printer definition or, if no value is specified, a default value assigned by JES. Table 36 summarizes the values in some of the ANFUEXTP fields when the **resubmit for filtering** option is selected.

*Table 36. Contents of ANFUEXTP fields when **resubmit for filtering** option is selected (IP PrintWay basic mode)*

ANFUEXTP field names	Contents of field
XTPKEY	Name of the printer definition.
XTPIPADR, XTPPRTQU, XTPIPORT	IP address, print queue name, and port number specified in the printer definition.
XTPRETNS, XTPRETNF	Retention times specified in the printer definition.
XTPRTRYL, XTPRTRYT	Retry values specified in the printer definition.
XTPUSRID	User ID of the person who submitted the original batch job.

To obtain the original JCL values specified by the job submitter, use the Scheduler Work Block Text Unit (SWBTU). (The address of the SWBTU is in field XTPSWBTP in the ANFUEXTP control block.) These are the JCL parameters in the SWBTU that contain the original values specified by the job submitter when the Routing, Begin Data Set, Record, and End Data Set exits are called:

- ADDRESS, BUILDING, DEPT, NAME, ROOM, TITLE
- DATAACK
- DUPLEX
- CHARS (If the CHARS parameter is not specified, the value in the UCS parameter is used.)
- COPIES
- FORMDEF
- FORMS
- INTRAY, OUTBIN
- OFFSETXB, OFFSETXF, OFFSETYB, OFFSETYF
- OVERLAYB, OVERLAYF
- PAGEDEF (If the PAGEDEF parameter is not specified, the value in the FCB parameter is used.)
- PRMODE
- TRC
- USERLIB

JCL parameters not in this list, such as the CLASS and DEST parameters, do not contain the original JCL value specified by the job submitter when the Routing, Begin Data Set, Record, and End Data Set exits are called. Instead, they contain values from the Allocation section of the printer definition.

The Begin Data Set, End Data Set, SMF, and Response Notification exits can check field XTP_FILTERING in the ANFUEXTP control block to determine whether the **resubmit for filtering** option is selected in the printer definition.

Programming considerations

Consider these points when coding an exit:

- You should be an experienced programmer who is aware of overall system implications.
- Test the exit before installing it on a production system. A programming exception in an exit causes IP PrintWay extended mode or the IP PrintWay basic mode FSA to abend. An ESTAI is in effect while the exit is running.
- The exit should restore all registers at exit and not set a return code in register 15.
- Do not do an explicit or implied WAIT in the exit because it puts IP PrintWay into a WAIT state.
- The exits run in 31-bit addressing mode.
- To reference fields in control blocks, use the name associated with each field in the DSECT control block. Do not use absolute displacements, because these displacements can change from one release to another.
- Code the exits to be reentrant.
- IP PrintWay basic mode exits run in supervisor state with a protection key of 1. IP PrintWay extended mode exits run in problem state with a protection key of 8.
- As it processes an output group, IP PrintWay passes the same Exit Parameter Area (ANFUEXTP) to the Begin Data Set exit, the Record exit, and the End Data Set exit.
- Exits, except for the Message exit in basic mode and the Routing exit in extended mode, use the same work areas. Fields XTPWORK1 and XTPWORK2 contain the addresses of the shared work areas. The sample exits check if these work area addresses are set and, if so, use the work areas. IP PrintWay frees the work area storage only when the IP PrintWay FSA ends (basic mode) or when IP PrintWay ends (extended mode).

You can code exits that allocate and free their own work areas. In this case, do not set the work area address in fields XTPWORK1 and XTPWORK2. These address fields should contain zeroes. In addition, be sure to free the work areas in the exits.

The Begin Data Set exit, Record exit, and End Data Set are called in sequence for a single data set. If these exits use the same work areas, they can pass information about a data set from one exit to another in the work areas. IP PrintWay never clears the information in the work areas.

The SMF exit is not called in sequence. Therefore, exits should not pass data set information to the SMF exit in the work areas.

Rule: If exits share work areas, all exits must declare the same work area size. The work area size must be the largest that any exit requires. The work area size should also contain additional space for future changes.

- A return code of 8 to the Response Notification exit indicates that an error was encountered in IP PrintWay's initial processing of a data set. In this case, no data set transmission processing exits are invoked. Therefore, no communication with these exits is possible.

Tip: Future IP PrintWay program maintenance might require that you reassemble the exits.

Common control blocks

All IP PrintWay exits, except for the Message exit, can access data set and job information that is contained in these control blocks:

- JES job separator page data area (IAZJSPA)
- Scheduler work block text unit (SWBTU)

All IP PrintWay exits can access data set and job information contained in this control block:

- IP PrintWay Exit Parameter Area (ANFUEXTP)

JES job separator page data area (IAZJSPA)

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
 - JSPJPRIO
 - JSPCEUID
 - JSPCEDSN
 - JSPCESEG
- These fields do not contain valid values:
 - JSPADEVN
 - JSPADEVA
 - JSPAJMR
 - JSPJGRP1
 - JSPJGRP2
 - JSPAUSR1
 - JSPAUSR2
 - JSPCELEN
 - JSPCEVSN
 - JSPCESEC.

Fields that do not contain valid values contain blanks (text fields) or zeros (numeric fields).

- The address of the JMR, in field JSPAJMR, is not available to IP PrintWay extended mode. However, IP PrintWay extended mode puts the system name in these fields of the ANFUEXTP control block:
XTP_ORIGINATING_MVS_SYSTEM and XTP_ORIGINATING_JES2_SYSTEM.

- No JSPA user extensions are present.

Tips:

- 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.
- For information about the contents of the JSPA, see *z/OS MVS Using the Functional Subsystem Interface*.

Scheduler work block text unit (SWBTU)

The SWBTU is a system control block that contains JCL parameters for the data set. These JCL values are merged from the DD statement, the OUTPUT statement, and JES defaults, according to a hierarchy determined by JES. If NetSpool or Print Interface allocated the data set on the JES spool, the JCL values come from the printer definition or Infoprint Server job attributes.

You can use the SWBTUREQ macro to obtain the JCL parameters specified for the data set from the SWBTU. For example, you could obtain distribution parameters specified on the OUTPUT statement and print them on a separator page created in your Begin Data Set exit. For information about how to specify distribution parameters on an OUTPUT statement, see *z/OS Infoprint Server User's Guide*.

If your installation selects the **resubmit for filtering** option in the printer definitions, see “Resubmit for filtering considerations (basic mode)” on page 240 for more information about the JCL parameters in this control block.

For information about the SWBTUREQ macro, see *z/OS MVS Programming: Authorized Assembler Services Reference SET-WTO*. For information about SWBs and how to use the SWB token to retrieve keywords, see *z/OS MVS Using the Subsystem Interface*.

IP PrintWay exit parameter area (ANFUEXTP)

The IP PrintWay Exit Parameter Area (ANFUEXTP) provides information to the exits about the data set being processed. Also, the exits can set flags in ANFUEXTP to specify the action IP PrintWay is to take at return and can modify values in ANFUEXTP to change the processing of IP PrintWay.

When you run IP PrintWay basic mode, the same ANFUEXTP control block is shared by all exits, except for the Message exit, that are called by the same IP PrintWay basic mode FSA. When you run IP PrintWay extended mode, the same ANFUEXTP control block is shared by all exits except for the Routing exit. Your exit programs do not need to enqueue on this control block because IP PrintWay calls the exits serially.

See the description of each exit for the fields that are valid at input and the fields that each exit can modify. Figure 23 on page 245 shows the format of ANFUEXTP. Macro ANFUEXTP provides a DSECT in assembler language for this control block.

Offsets	Hex	Type	Length	Name	Description
0	(0)	Structure	1280	ANFUEXTP	Exit parameter area
0	(0)	Character	4	XTPID	Eye catcher - 'XTP '
4	(4)	Unsigned	72	XTPSAVE	Save area for exit
76	(4C)	Address	4	XTPWORK1	Work area for exit
80	(50)	Address	4	XTPWORK2	Work area for exit
84	(54)	Character	53	XTPDSNAM	Data set name
137	(89)	Unsigned	1	XTPVERS	Version of IP PrintWay: 0 = basic mode 1 = extended mode
138	(8A)	Character	2	*	Reserved
140	(8C)	Character	8	XTPUSRID	User id of submitter
148	(94)	Bit string	1	XTPSRCFL	Flags showing where values came from. If the bit is on the value came from JCL on OUTPUT; otherwise it came from the printer definition or defaults:
		1... ..		XTPSOPTS	PRTOPTNS from JCL
		.1.. ..		XTPSIPAD	IPADDR from JCL
		..1.		XTPSPRTQ	PRTQUEUE from JCL
		...1		XTPSRTNS	RETAINS from JCL
	 1...		XTPSRTNF	RETAINF from JCL
	1..		XTPSRTYL	RETRYL from JCL
	1.		XTPSRTYT	RETRYT from JCL
	1		XTPSPORT	PORTNO from JCL
149	(95)	Bit string	1	XTPDSFLG	Data set information flags:
		1... ..		XTPDSCC	data set has carriage controls
		.1.. ..		XTPDSHLD	release data set to JES; data set is held on JES spool
		..1.		XTPDSTRM	release data set to JES; data set is deleted from JES spool
		...1		XTPDSRTG	reselect printer definition
	 1...		XTPDSJOB	data set is first in job
	1..		XTPDSJNW	data set is JESNEWS data set
	1.		XTPDSEJB	data set is last in job
	1		XTPDSCON	data set is in a concatenation
150	(96)	Bit string	1	XTPRCFLG	Record flags:
		1... ..		XTPRCTRN	translate record from exit
		.1.. ..		XTPRCCC	record from exit has carriage controls
		..1.		XTPRCORG	transmit original record
		...1		XTPRDFST	1st call to exit for record or data set
	 1...		XTPRLAST	last record from exit
	1..		XTPRCEXT	transmit record from exit
	1.		XTPRCDBL	record from exit is double-byte
	1		*	reserved
151	(97)	Bit string	1	XTPMSGFL	Message flags:
		1... ..		XTPM2CON	issue message to console
		.1.. ..		XTPM2MDS	issue message to message data set
		..11 1111		*	reserved
152	(98)	Address	4	XTPERPTR	Address of record from exit
156	(9C)	Unsigned	4	XTPERLEN	Length of record from exit
160	(A0)	Address	4	XTPPRPTR	Address of original record
164	(A4)	Unsigned	4	XTPPRLN	Length of original record
168	(A8)	Character	16	XTPOPTNM	Name of components
184	(B8)	Unsigned	2	XTPLLEN	Length of LPR options
186	(BA)	Unsigned	2	XTPPQLEN	Length of value in XTPPRTQU
188	(BC)	Unsigned	2	XTPIALEN	Length of value in XTPIPADR
190	(BE)	Unsigned	2	XTPMSGLN	Length of message
192	(C0)	Character	256	XTPPRTQU	Name of print queue or VTAM logmode
448	(1C0)	Character	256	XTPIPADR	IP address/host name, URL, LU name, or email addresses

Figure 23. Format of the IP PrintWay exit parameter area (ANFUEXTP) (Part 1 of 3)

704	(2C0)	Unsigned	4	XTPRETNS	Success retain time (see note)
708	(2C4)	Unsigned	4	XTPRETNF	Failure retain time (see note)
712	(2C8)	Unsigned	4	XTPRTRYL	Retry limit
716	(2CC)	Unsigned	4	XTPRTRYT	Retry time (see note)
720	(2D0)	Character	17	XTPKEY	Name of printer definition
720	(2D0)	Character	8	XTPDEST	DEST
728	(2D8)	Character	1	XTPCLASS	CLASS
729	(2D9)	Character	8	XTPFORMS	FORMS
737	(2E1)	Bit string	1	XTPRECFM	Record format
738	(2E2)	Unsigned	2	XTPMRECL	Maximum record length
740	(2E4)	Address	4	XTPSWBTP	Address of SWBTU area
744	(2E8)	Unsigned	4	XTPSWBTL	Length of SWBTU area
748	(2EC)	Address	4	XTPJSPAP	Address of JSPA
752	(2F0)	Unsigned	4	XTPCOPYS	Number of copies
756	(2F4)	Address	4	XTPMSGP	Address of message
760	(2F8)	Unsigned	4	XTPIPORT	PORTNO value
764	(2FC)	Bit String	1	XTP_MISC_FLAGS	Miscellaneous flags:
		1... ..		XTP_FIRST_REC	first record of a data set
		.1.. ..		XTP_MAIL_INCOM	email addresses exceed 256 chars
		..1.		XTP_TRANS_DOC_HEADER	translate document header
		...1		XTP_TRANS_DOC_TRAILER	translate document trailer
	 1111		*	reserved
765	(2FD)	Character	2	XTPMLINL	Maximum line length
767	(2FF)	Character	1	*	Reserved

=====

The following fields are used by the Block Letter program.

=====

768	(300)	Bit string	1	XTPBIFLG	Input flags for block program:
		1... ..		XTPBSLNT	slant the letters
		.1..		XTPBPRFM	performance (narrow) letters
		..1.		XTPBLJST	left justify letters
		...1		XTPBFRST	first call to block program for this string
	 1111		*	reserved
769	(301)	Bit string	1	XTPBOFLG	Output flags for block program:
		1... ..		XTPBLAST	this is the last line
		.111 1111		*	reserved
770	(302)	Unsigned	1	XTPBLENG	Length of output record
771	(303)	Character	1	*	Reserved
772	(304)	Character	8	XTPBSTRG	String to convert, blank padded
780	(30C)	Character	150	XTPBOUTP	Output string
930	(3A2)	Character	16	XTPBWRKA	Work area for block program
946	(3B2)	Character	2	*	Reserved
948	(3B4)	Signed	4	XTPRSPCD	Response code

Figure 23. Format of the IP PrintWay exit parameter area (ANFUEXTP) (Part 2 of 3)

```

=====
The following fields are used by all exits.
=====
952 (3B8) Unsigned      4  XTP_ADDR_TYPE  Protocol that IP PrintWay uses:
                        0 = LPR
                        1 = IPP
                        2 = direct sockets
                        3 = VTAM
                        4 = email
956 (3BC) Unsigned      4  XTP_ROUTING_KEY_TYPE
                        Value in XTPKEY field:
                        0 = DEST, CLASS, and FORMS
                        1 = Printer definition name
960 (3C0) Unsigned      4  XTP_FILTERING  Resubmit for filtering values:
                        0 = No resubmit for filtering
                        1 = This is first data set processed
                        2 = This is second data set processed
964 (3C4) Character     8  XTP_JCL_DEST   DEST from JCL
972 (3CC) Character     1  XTP_JCL_CLASS  CLASS from JCL
973 (3CD) Character     8  XTP_JCL_FORMS  FORMS from JCL
981 (3D5) Character     8  XTP_ORIGINATING_MVS_SYSTEM
                        System name
989 (3DD) Character     4  XTP_ORIGINATING_JES2_SYSTEM
                        System name (JES2 only)
993 (3E1) Character     7  *              Reserved
1000 (3E8) Character    1012 XTPOPTNS       Transmission options (See macro for field names)
2012 (7DC) Address      4  XTP_DOC_HEADER_PTR   Document header address
2016 (7E0) Address      4  XTP_DOC_TRAILER_PTR  Document trailer address
2020 (7E4) Signed       2  XTP_DOC_HEADER_LENGTH Document header length
2022 (7E6) Signed       2  XTP_DOC_TRAILER_LENGTH Document trailer length

```

Figure 23. Format of the IP PrintWay exit parameter area (ANFUEXTP) (Part 3 of 3)

Note: The time in the XTPRETNS, XTPRETNF, and XTPRTRYT fields is a hexadecimal number: $(hours \times 10\,000) + (minutes \times 100) + seconds$. For example, if the time period is 24 hours, 5 minutes, and 1 second, this field contains 03AB75, the hexadecimal representation of the decimal number 240501.

Routing exit (ANFUXRTG)

IP PrintWay calls the Routing exit one time for each data set. The Routing exit lets you:

- Inspect routing parameters for a data set before it is transmitted to the printer or email destination.

The Routing exit can inspect the address of the printer or email destination, retry values, retention values, the name of the components that contain printer options, and the number of copies.

When the exit gets control, the values from the DD and OUTPUT JCL statements for the data set and from the printer definition in the Printer Inventory have been merged. Parameters on the DD and OUTPUT override the values specified in the printer definition. The XTPSRCFL field in the ANFUEXTP control block indicates whether a parameter was specified on a JCL statement.

- Reroute a data set (basic mode) or output group (extended mode) to another printer or email destination.

The exit can specify a new printer or email address in the ANFUEXTP control block, or the exit can specify a new printer definition for IP PrintWay to use. The exit can identify the new printer definition by either the printer definition name or by the class, destination, and form name assigned to it.

- Change the retry time and retention values for a data set (basic mode) or output group (extended mode).
- Change the number of copies of the data set that are printed.
- Change the name of the components that contain default options.
- IP PrintWay basic mode: Hold the data set on the JES spool for operator intervention.
- IP PrintWay basic mode: Delete the data set from the JES spool.
- Specify a message for IP PrintWay to write. IP PrintWay assigns message ID ANFM155I (basic mode) or AOP3003I (extended mode) to each message created by the exit.

IP PrintWay calls the same Routing exit before processing each data set or output group. You can install different Routing exits for different IP PrintWay basic mode FSSs. For information, see “Installing IP PrintWay exits” on page 300.

You might want to write a Routing exit to do one of these functions:

- Enforce limits in JCL parameters. For example, your installation might want to limit the length of the retention time specified by job submitters on the OUTPUT JCL statement. To do this, the Routing exit could inspect the values specified in the RETAINS and RETAINF JCL parameters and change them if they exceed the maximum time defined for your installation.
- Restrict the use of certain printers to specific users. To do this, the Routing exit could inspect the user ID of the job submitter and the host name or IP address and the print queue name or port of the printer. If you run IP PrintWay basic mode, your exit could hold or delete a data set or output group if the user ID of the job submitter is not on your list of authorized user IDs.

Tip: IP PrintWay does not call the Routing exit under these conditions:

- IP PrintWay cannot find a printer definition for a data set and the DEST=IP: JCL parameter is not specified.
- The **resubmit for filtering** option is selected in the printer definition and IP PrintWay basic mode is transmitting the data set to Print Interface for filtering. For more information, see “Resubmit for filtering considerations (basic mode)” on page 240.

Input to the Routing exit

At entry to the Routing exit, the registers contain:

- | | |
|--------------------|--|
| Register 1 | Pointer to the address of the ANFUEXTP control block. |
| Register 13 | Address of an 18-word save area for saving the caller's registers. |
| Register 14 | Return address. |
| Register 15 | Entry-point address. |

These fields in the ANFUEXTP control block are valid when the Routing exit is called. The format of ANFUEXTP is shown in Figure 23 on page 245.

- | | |
|-----------------|---|
| XTPSAVE | A register save area for use by the exit. |
| XTPWORK1 | Address of a work area, or zeroes. For information about how to use work areas, see “Programming considerations” on page 242. |
| XTPWORK2 | Address of a work area, or zeroes. For information about how to use work areas, see “Programming considerations” on page 242. |

XTPDSNAM Fully qualified name of the data set being processed, padded to the right with blanks.

XTPVERS Indicates whether IP PrintWay basic mode or extended mode called the exit and constructed the ANFUEXTP control block.

Value Meaning

0 IP PrintWay basic mode called the exit.

1 IP PrintWay extended mode called the exit.

Tip: Your exit can use this field to determine whether fields XTP_ORIGINATING_MVS_SYSTEM and XTP_ORIGINATING_JES2_SYSTEM contain valid values. These fields contain valid values if XTPVERS=1 (IP PrintWay extended mode called the exit).

XTPUSRID User ID of the job submitter, padded to the right with blanks, with these exceptions:

- If NetSpool allocated the data set on the JES spool, this field contains the owner embedded in the print data or specified in the printer definition. If none is specified, this is the ID of the user who started the Infoprint Server daemons.
- If Print Interface allocated the data set on the JES spool and the print request was not submitted from a z/OS system, this field contains the user ID from the remote system of the person who submitted the print request.

XTPSRCFL Flags indicating whether a parameter was specified as a JCL parameter.

If the flag is set to 1, the indicated parameter was specified as a JCL parameter.

If the flag is set to 0, the JCL parameter was not specified. Instead, IP PrintWay obtained the value from a printer definition. If the DEST=IP: parameter was specified in JCL without the name of a printer definition, IP PrintWay obtained the value from the IP PrintWay default printer definition. If no PrintWay default printer definition exists, IP PrintWay sets a default value.

Flag Meaning when set to B'1'

XTPSOPTS The PRTOPTNS parameter specified the name of components in the Printer Inventory.

XTPSIPAD The DEST=IP: parameter specified the host name or IP address of the target printer or print server.

XTPSPRTQ The PRTQUEUE parameter specified the name of the target print queue.

XTPSRTNS The RETAINS parameter specified the retention time after a successful transmission.

XTPSRTNF The RETAINF parameter specified the retention time after a failed transmission.

XTPSRTYL The RETRYL parameter specified the retry limit.

XTPSRTYT The RETRYT parameter specified the retry time.

XTPSPORT The PORTNO parameter specified the port number.

XTPDSFLG	Flags containing data set information:										
	<table border="0"> <thead> <tr> <th>Flag</th> <th>Meaning when set to B'1'</th> </tr> </thead> <tbody> <tr> <td>XTPDSJOB</td> <td>This is the first data set in the print job (JES output group).</td> </tr> <tr> <td>XTPDSJNW</td> <td>This data set is the JESNEWS data set. This flag applies only to IP PrintWay basic mode.</td> </tr> <tr> <td>XTPDSEJB</td> <td>This is the last data set in the print job (JES output group).</td> </tr> <tr> <td>XTPDSCON</td> <td>This data set is part of an IP PrintWay concatenation. IP PrintWay sets this flag when the Concatenate job option is selected in the Dataset grouping field of the printer definition and IP PrintWay is transmitting this data set with other data sets in the same transmission.</td> </tr> </tbody> </table>	Flag	Meaning when set to B'1'	XTPDSJOB	This is the first data set in the print job (JES output group).	XTPDSJNW	This data set is the JESNEWS data set. This flag applies only to IP PrintWay basic mode.	XTPDSEJB	This is the last data set in the print job (JES output group).	XTPDSCON	This data set is part of an IP PrintWay concatenation. IP PrintWay sets this flag when the Concatenate job option is selected in the Dataset grouping field of the printer definition and IP PrintWay is transmitting this data set with other data sets in the same transmission.
Flag	Meaning when set to B'1'										
XTPDSJOB	This is the first data set in the print job (JES output group).										
XTPDSJNW	This data set is the JESNEWS data set. This flag applies only to IP PrintWay basic mode.										
XTPDSEJB	This is the last data set in the print job (JES output group).										
XTPDSCON	This data set is part of an IP PrintWay concatenation. IP PrintWay sets this flag when the Concatenate job option is selected in the Dataset grouping field of the printer definition and IP PrintWay is transmitting this data set with other data sets in the same transmission.										
XTPOPTNM	The component name that was specified in the PRTOPTNS JCL parameter. The name is case-sensitive and is padded to the right with blanks. This field contains blanks if no component name was specified in the PRTOPTNS parameter.										
XTPPQLEN	Length of the value in field XTPPRTQU.										
XTPIALEN	Length of the value in field XTPIPADR.										
XTPPRTQU	Name of the target print queue specified in the PRTQUEUE JCL parameter or in a printer definition; or the VTAM logmode specified in a printer definition. Field XTP_ADDR_TYPE identifies which value is in this field.										
XTPIPADR	Field XTP_ADDR_TYPE identifies which of these values this field contains: <ul style="list-style-type: none"> • Host name or IP address of the target printer or print server specified in the DEST=IP: JCL parameter, the printer-ip-address job attribute, or in the printer definition • URL of the target printer specified in the printer definition • LU name of the target printer specified in the printer definition • Email addresses of the primary recipients specified in the MAILTO JCL parameter, mail-to-addresses job attribute, or printer definition. Multiple email addresses are separated by a comma. If the email addresses are greater than 256 characters, field XTP_MAIL_INCOM is set and field XTPIPADR contains only the email addresses that fit into this field. If field XTP_MAIL_INCOM is set, IP PrintWay does not let the Routing exit update this field. <p>Tip: IP PrintWay does not pass the email addresses of the copy or blind copy recipients to the exit.</p>										
XTPRETNS	Period of time IP PrintWay is to retain the data set on the JES spool after a successful transmission. This time was specified in the RETAINS JCL parameter or in the printer definition. IP PrintWay might have set a default value of zeroes. For the format of this field, see the note on page 247.										
XTPRETNF	Period of time IP PrintWay is to retain the data set on the JES spool after all transmission retries have failed. This time was specified in the RETAINF JCL parameter or in the printer										

definition. IP PrintWay might have set a default value of zeroes. For the format of this field, see the note on page 247.

XTPRTRYL Number of times IP PrintWay is to retry transmission of the data set. This number was specified in the RETRYL JCL parameter or in the printer definition. IP PrintWay might have set a default value of zero.

XTPRTRYT Period of time IP PrintWay is to wait between retries of the transmission. This time was specified in the RETRYT JCL parameter or in the printer definition. IP PrintWay might have set a default value of zero. For the format of this field, see the note on page 247.

XTPKEY If XTP_ROUTING_KEY_TYPE is set to '1', this field contains the name of the printer definition that was specified in JCL or in the printer definition. The name is padded to the right with blanks. The printer definition name is case-sensitive.

Fields XTPDEST, XTPCLASS, and XTPFORMS are mapped to the same location as XTPKEY.

XTPDEST If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the DEST field of the printer definition that IP PrintWay uses to print the data set. The name is padded to the right with blanks. If the DEST field in the printer definition is not specified, this field contains blanks.

Tip: Use field XTP_JCL_DEST if you want to obtain the value specified in the JCL parameter.

XTPCLASS If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the CLASS field of the printer definition that IP PrintWay uses to print the data set. If the CLASS field in the printer definition is not specified, this field contains blanks.

Tip: Use field XTP_JCL_CLASS if you want to obtain the value specified in the JCL parameter.

XTPFORMS If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the FORMS field of the printer definition that IP PrintWay uses to print the data set. The name is padded to the right with blanks. If the FORMS field in the printer definition is not specified, this field contains blanks.

Tip: Use field XTP_JCL_FORMS if you want to obtain the value specified in the JCL parameter.

XTPRECFM **Basic mode:** The record format as indicated in the JFCRECFM field in the job file control block (JFCB).

Extended mode: Indicates the type of carriage control characters:

Value Meaning

- X'04' A - American national standard (ASA) control characters
- X'02' M - Machine code control characters
- X'00' No control characters

XTPMRECL Maximum record length specified on the LRECL parameter of the OUTPUT JCL statement. For data sets that NetSpool and Print Interface allocate on the JES spool, the value is 32756.

XTPSWBTP Address of the SWBTU for the data set. For more information, see "Scheduler work block text unit (SWBTU)" on page 244.

XTPSWBTL Length of the SWBTU.

XTPVERS Indicates whether IP PrintWay basic mode or extended mode called the exit and constructed the ANFUEXTP control block.

Value Meaning

- 0 IP PrintWay basic mode called the exit.
- 1 IP PrintWay extended mode called the exit.

Tip: Your exit can use this field to determine whether fields XTP_ORIGINATING_MVS_SYSTEM and XTP_ORIGINATING_JES2_SYSTEM contain valid values. These fields contain valid values if XTPVERS=1 (IP PrintWay extended mode called the exit).

XTPJSPAP The address of a full or partial JSPA. For more information about this field, see "JES job separator page data area (IAZJSPA)" on page 243.

- IP PrintWay basic mode: Address of the JSPA (IAZJSPA) for the data set being processed.
- IP PrintWay extended mode: Address of a partial JSPA (IAZJSPA) for the data set being processed.

XTPCOPYS Number of copies of the data set. This number was specified in JCL, in the printer definition, or JES provided a default value.

XTPIPORT Port number on the target system. This number was specified in the PORTNO JCL parameter or in the printer definition.

XTP_MISC_FLAGS

Flags containing miscellaneous information:

Flag	Meaning when set to B'1'
XTP_MAIL_INCOM	The list of email addresses for the primary recipients exceeds 256 characters. Therefore, field XTPIPADR does not contain the complete list of email addresses. If this flag is set, IP PrintWay does not let the Routing exit modify field XTPIPADR.

XTP_ADDR_TYPE

This field identifies the transmission protocol and the contents of fields XTPIPADR, XTPPRTQU and XTPIPORT:

Value	Protocol	XTPIPADR	XTPPRTQU	XTPIPORT
0	LPR	IP address	Print queue name	
1	IPP	URL		
2	Direct sockets	IP address		Port number
3	VTAM	LU name	Logmode	
4	Email	email addresses		

XTP_ROUTING_KEY_TYPE

An indicator of how IP PrintWay selected the printer definition it will use to obtain printing options:

Value Meaning

- 0** IP PrintWay selected the printer definition by matching the DEST, CLASS, and FORMS fields specified in JCL parameters with the values in the printer definition.
- 1** IP PrintWay selected the printer definition using the printer definition name specified in JCL parameters or specified by NetSpool or Print Interface.

XTP_JCL_DEST

The DEST value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the DEST value in the printer definition. Otherwise, it is the DEST value in the OUTPUT JCL statement unless the **resubmit for filtering** option is selected in the printer definition, in which case this field contains the DEST value specified in the printer definition. The value is padded to the right with blanks.

XTP_JCL_CLASS

The CLASS value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the CLASS value in the printer definition. Otherwise, it is the CLASS value specified in a JCL statement unless the **resubmit for filtering** option is selected in the printer definition, in which case this field contains the CLASS value specified in the printer definition. The value is padded to the right with blanks.

XTP_JCL_FORMS

The FORMS value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the FORMS value in the printer definition. Otherwise, it is the FORMS value in a JCL statement. The value is padded to the right with blanks.

XTP_ORIGINATING_MVS_SYSTEM

The system name (left-justified and padded with blanks) of the MVS image where the data set originated. This field contains the same value as the SSS2SYS field in the IAZSSS2 control block. For information about the SSS2SYS field, see *z/OS MVS Using the Subsystem Interface*. This field contains the system name only if you run IP PrintWay extended mode (XTPVERS=1). IP PrintWay basic mode does not set this field.

Tip: In a JES2 environment, the XTP_ORIGINATING_JES2_SYSTEM field contains the member name of the individual JES2 system where the data set originated.

XTP_ORIGINATING_JES2_SYSTEM

Contains the member name (left-justified and padded with blanks) of the individual JES2 system where the data set originated. This field contains the same value as the SSS2MBR field in the IAZSSS2 control block. For information about the SSS2MBR field, see *z/OS MVS Using the Subsystem Interface*. This field contains the member name only if you run IP PrintWay extended mode (XTPVERS=1). IP PrintWay basic mode does not set this field. On a JES3 system, this field contains blanks.

Output from the Routing exit

At return, restore the contents of all registers. Do not set a return code in register 15.

The exit can modify these fields in the ANFUEXTP control block:

- XTPWORK1** Address of a work area, or zeroes. For information about how to use work areas, see "Programming considerations" on page 242.
- XTPWORK2** Address of a work area, or zeroes. For information about how to use work areas, see "Programming considerations" on page 242.
- XTPDSFLG** Flags indicating the action you want IP PrintWay to take upon return. IP PrintWay sets these flags to B'0' before calling the exit. To transmit the data set, do *not* set any of these flags.

Note: Set only one of these flags:

Flag	Meaning when set to B'1'
-------------	---------------------------------

XTPDSHLD	Hold the data set on the JES spool for operator intervention. This flag applies only to IP PrintWay basic mode. IP PrintWay extended mode ignores this flag.
-----------------	--

XTPDSTRM	Delete the data set from the JES spool. This flag applies only to IP PrintWay basic mode. IP PrintWay extended mode ignores this flag.
-----------------	--

XTPDSRTG	Reselect the printer definition using the value in the XTPKEY field.
-----------------	--

- XTPMSGFL** Flags indicating the destination of a message created by this exit. IP PrintWay sets these flags to B'0' before calling the exit. Put the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN.

Flag	Meaning when set to B'1'
-------------	---------------------------------

XTPM2CON	IP PrintWay basic mode writes the message to the console. IP PrintWay extended mode writes the message to both the console and the Infoprint Server common message log.
-----------------	---

XTPM2MDS	IP PrintWay basic mode writes the message to the IP PrintWay message-log data set. IP PrintWay
-----------------	--

extended mode writes the message to both the Infoprint Server common message log and the console.

- XTPMSGLN** Length of the message whose address is in field XTPMSGP.
- XTPMSGP** Address of message created by this exit. IP PrintWay adds message ID ANFM155I (IP PrintWay basic mode) or AOP3003I (IP PrintWay extended mode) to the beginning of this message.

If you set the XTPDSRTG flag, you can also change these fields in the ANFUEXTP control block. (IP PrintWay ignores changes you make to fields containing routing parameters, such as fields XTPOPTNM, XTPPRTQU, XTPIPADR, XTPRETNS, XTPRETNF, XTPRTRYL, and XTPRTRYT.)

- XTPKEY** Name of the printer definition you want IP PrintWay to use. This field is case-sensitive. Also set XTP_ROUTING_KEY_TYPE to '1'. Fields XTPDEST, XTPCLASS, and XTPFORMS are mapped to the same location as XTPKEY.
- IP PrintWay extended mode uses the value in this field only if this is the first data set in the output group. Otherwise, it ignores the value and uses the value for the first data set.
- XTPCLASS** Class specified in the CLASS field of the printer definition you want IP PrintWay to use. Do not use lowercase characters. Also set XTP_ROUTING_KEY_TYPE to '0'.
- XTPDEST** Destination name specified in the DEST field of the printer definition you want IP PrintWay to use, padded to the right with blanks. Do not use lowercase characters. Also set XTP_ROUTING_KEY_TYPE to 0.
- Note:** Do not specify DFLTENTRY as the destination name.
- XTPFORMS** Form name specified in the FORMS field of the printer definition you want IP PrintWay to use, padded to the right with blanks. Do not use lowercase characters. Also set XTP_ROUTING_KEY_TYPE to '0'.
- XTPCOPYS** Number of copies of the data set. Specify a number from 1-255.

XTP_ROUTING_KEY_TYPE

Indicates how you want IP PrintWay to select the new printer definition:

Value Meaning

- 0** IP PrintWay is to select the printer definition by matching the values in fields XTPCLASS, XTPDEST, and XTPFORMS with the values in the printer definition.
- 1** IP PrintWay is to select the printer definition using the printer definition name in field XTPKEY.

If you do not set any of the flags in field XTPDSFLG, you can change the values in these fields in the ANFUEXTP control block. IP PrintWay uses the changed values

when it transmits the data set. (IP PrintWay ignores changes you make to fields XTPKEY, XTPCLASS, XTPFORMS, and XTPDEST.)

XTPOPTNM The name of components in the Printer Inventory that contain default options. The component name is case-sensitive and must be padded to the right with blanks.

XTPPQLEN Length of the value in field XTPPRTQU.

XTPPRTQU Name of the printer or print server's queue, or the VTAM logmode.

IP PrintWay extended mode uses the value in this field only if this is the first data set in the output group. Otherwise, it ignores the value and uses the value for the first data set.

XTPIALEN Length of the value in field XTPIPADR.

XTPIPADR Specify one of these values:

- Host name or IP address of the printer or print server.
- URL of the printer or print server.
- LU name of the printer.
- Email addresses of the primary recipients. Separate each email address or z/OS UNIX sendmail alias name with a comma. If field XTP_MAIL_INCOM is set at entry to the exit, which indicates that the original email addresses are greater than 256 characters, IP PrintWay does not let the exit modify this field.

Tips:

- The Routing exit cannot change the email addresses of the copy or blind copy recipients.
- If an email address is specified in a header in the document, the email address in the document overrides the email address specified in this field.

IP PrintWay extended mode uses the value in this field only if this is the first data set in the output group. Otherwise, it ignores the value and uses the value for the first data set.

XTPRETNS Period of time IP PrintWay is to retain the data set on the JES spool after a successful transmission. For the format of this field, see the note on page 247.

IP PrintWay extended mode uses the value in this field only if this is the first data set in the output group. Otherwise, it ignores the value and uses the value for the first data set.

XTPRETNF Period of time IP PrintWay is to retain the data set on the JES spool after all transmission retries have failed. For the format of this field, see the note on page 247.

IP PrintWay extended mode uses the value in this field only if this is the first data set in the output group. Otherwise, it ignores the value and uses the value for the first data set.

XTPRTRYL Number of times IP PrintWay is to retry transmission of the data set. Specify a number from 0-32767.

IP PrintWay extended mode uses the value in this field only if this is the first data set in the output group. Otherwise, it ignores the value and uses the value for the first data set.

- XTPRTRYT** Period of time IP PrintWay is to wait between retries of the transmission. For the format of this field, see the note on page 247. IP PrintWay extended mode uses the value in this field only if this is the first data set in the output group. Otherwise, it ignores the value and uses the value for the first data set.
- XTPCOPYS** Number of copies of the data set. Specify a number from 1-255.
- XTPIPORT** Port number of the printer or print server. IP PrintWay extended mode uses the value in this field only if this is the first data set in the output group. Otherwise, it ignores the value and uses the value for the first data set.

Examples -- Routing exit

1. To request that IP PrintWay obtain routing parameters from a specific printer definition:
 - Set XTPDSRTG=1 in the ANFUEXTP control block.
 - Specify the printer definition name in field XTPKEY; or specify the class, destination name, and forms name in fields XTPCLASS, XTPDEST, and XTPFORMS.
 - Set XTP_ROUTING_KEY_TYPE, depending on whether you specified the printer definition name or the class, destination, and forms names.
2. To transmit the data set without changing any routing information, simply return control to IP PrintWay.
3. To transmit the data set to:
 - A different port number, overriding the previous port number:
 - Put the port number in XTPIPORT
 - A different print queue, overriding the previous print queue:
 - Put the name of the printer's or print server's queue in field XTPPRTQU.
 - Put the length of the queue name in field XTPPQLEN.
 - A different VTAM LU name, overriding the previous LU name:
 - Put the LU name in field XTPIPADR.
 - Put the length of the LU name in field XTPIALEN.
 - A different email address, overriding the previous email addresses:
 - Put the email addresses and z/OS UNIX sendmail alias names in field XTPIPADR. Separate each address and alias name with a comma.
 - Put the length of the email addresses in field XTPIALEN.

Tip: You *cannot* change the protocol that IP PrintWay uses in the XTP_ADDR_TYPE field.
4. To request that IP PrintWay write a message to the console:
 - Set XTPM2CON=1 in the ANFUEXTP control block.
 - Put the address of the message in field XTPMSGP.
 - Put the length of the message in field XTPMSGLN.

Also see the sample exits distributed with IP PrintWay and described in “IP PrintWay sample exits” on page 299.

Begin Data Set exit

The Begin Data Set exit lets you:

- Inspect the routing parameters and transmission options for a data set.
- Change most of the transmission options for a data set.
- Add one or more records to the beginning of a data set. For example, the exit can add records to create a separator page.
 - The exit can add more than one record by requesting that IP PrintWay call the exit multiple times for the same data set.
 - On a separator page, the exit can print job and data set information contained in the scheduler work block text unit (SWBTU). See “Scheduler work block text unit (SWBTU)” on page 244.
 - The exit can create a job-level separator page by adding a separator page only for the first data set in each print job (JES output group). The XTPDSJOB flag in the ANFUEXTP control block indicates whether the data set is the first data set in a print job.
 - The exit can use the Block Letter program (ANFUBLK) to build block letters suitable for printing on a separator page. For more information, see “Using the block letter program (ANFUBLK)” on page 298.
 - When IP PrintWay sends multiple data sets to a printer in a concatenation (**dataset-grouping=concatenate-job** attribute is specified in the printer definition), it sends the records that the Begin Data Set exit adds before the first data set in the concatenation. It does not send the records again before each subsequent data set in the concatenation.

Tip: If you run IP PrintWay basic mode, the administrator can set the **concatenation-separators=document** printer attribute in the IP PrintWay FSS definition to send the records before each data set in the concatenation. However, this can cause extra an separator page to print before each copy. For information, see *z/OS Infoprint Server Operation and Administration*.

- Specify a document header and a document trailer for a data set (see the XTP_DOC_HEADER_PTR and XTP_DOC_TRAILER_PTR fields). Typically, the document header and document trailer contain printer commands. For example, a document header might specify printer commands for landscape printing, and a document trailer might specify a printer command to reset the printer to its original state.
 - The exit can specify a different document header and trailer for each data set. This capability exists even for data sets that IP PrintWay sends to the printer in the same job or the same concatenation.
 - The document header and trailer override the document header and trailer specified in the printer definition.
 - IP PrintWay sends the document header and trailer to the printer before and after each data set, and before and after each subsequent copy of the data set in this order:
 1. Records (Begin Data Set)
 2. Document header
 3. Data set
 4. Document trailer
 5. Document header
 6. Data set (copy)
 7. Document trailer
 8. Records (End Data Set)

Tip: If you add printer commands, in some cases you might need to redefine the print driver on your workstation. For example, when printing data on an HP Laserjet III_{Si} printer on a Windows system, define the HP print driver as a "generic/text only" driver. This prevents the print driver from changing your setup strings to be incorrect.

- Inspect the routing parameters of a data set. However, the exit cannot change them.
- Specify a message for IP PrintWay to write. IP PrintWay assigns message ID ANFM155I (basic mode) or AOP3003I (extended mode) to the message.

Tips:

- The administrator specifies the name of the Begin Data Set exit in the printer definition. Therefore, you can write different Begin Data Set exits for different data sets if the data sets use different printer definitions.
- IP PrintWay does not call the Begin Data Set exit when the **resubmit for filtering** option is selected in the printer definition and IP PrintWay basic mode transmits the data set to Print Interface for filtering. For more information, see "Resubmit for filtering considerations (basic mode)" on page 240.

Input to the Begin Data Set exit

At entry to the Begin Data Set exit, the registers contain:

- Register 1** Pointer to the address of the ANFUEXTP control block.
- Register 13** Address of an 18-word save area for saving the caller's registers.
- Register 14** Return address.
- Register 15** Entry-point address.

These fields in ANFUEXTP are valid when the Begin Data Set exit is called. The format of the ANFUEXTP control block is shown in Figure 23 on page 245.

- XTPSAVE** A register save area for use by the exit.
- XTPWORK1** Address of a work area, or zeroes. For information about how to use work areas, see "Programming considerations" on page 242.
- XTPWORK2** Address of a work area, or zeroes. For information about how to use work areas, see "Programming considerations" on page 242.
- XTPDSNAM** Fully qualified name of the data set being processed, padded to the right with blanks.
- XTPVERS** Indicates whether IP PrintWay basic mode or extended mode called the exit and constructed the ANFUEXTP control block.

Value Meaning

- 0** IP PrintWay basic mode called the exit.
- 1** IP PrintWay extended mode called the exit.

Tip: Your exit can use this field to determine whether fields XTP_ORIGINATING_MVS_SYSTEM and XTP_ORIGINATING_JES2_SYSTEM contain valid values. These fields contain valid values if XTPVERS=1 (IP PrintWay extended mode called the exit).

- XTPUSRID** User ID of the job submitter, padded to the right with blanks, with these exceptions:

- If NetSpool allocated the data set on the JES spool, this field contains the owner embedded in the print data or specified in the printer definition. If none is specified, this is the ID of the user who started the Infoprint Server daemons.
- If Print Interface allocated the data set on the JES spool and the print request was not submitted from a z/OS system, this field contains the user ID from the remote system of the person who submitted the print request.

XTPDSFLG Flags containing data set information:

Flag	Meaning when set to B'1'
XTPDSCC	This data set contains carriage control characters.
XTPDSJOB	This is the first data set in the print job (JES output group).
XTPDSJNW	This data set is the JESNEWS data set. This flag applies only to IP PrintWay basic mode.
XTPDSEJB	This is the last data set in the print job (JES output group).
XTPDSCON	This data set is part of an IP PrintWay concatenation. IP PrintWay sets this flag when the Concatenate job option is selected in the Dataset grouping field of the printer definition and IP PrintWay is transmitting this data set with other data sets in the same transmission.

XTPRCFLG Flags containing data set information:

Flag	Meaning when set to B'1'
XTPRDFST	This is the first call to the exit for this data set.

XTPPQLEN Length of the value in field XTPPRTQU.

XTPIALEN Length of the value in field XTPIPADR.

XTPPRTQU Name of the printer's or print server's queue specified in the PRTQUEUE JCL parameter, in a printer definition, or by the Routing exit; or the VTAM logmode specified in a printer definition or by the Routing exit. Field XTP_ADDR_TYPE identifies which value is in this field.

XTPIPADR Field XTP_ADDR_TYPE identifies which of these values this field contains:

- Host name or IP address of the printer or print server specified in the DEST=IP: JCL parameter, in the **printer-ip-addresses** job attribute, in the printer definition, or by the Routing exit
- URL of the printer or print server specified in the printer definition or by the Routing exit
- LU name of the printer specified in the printer definition or by the Routing exit
- Email addresses of the primary recipients specified in the MAILTO JCL parameter, **mail-to-addresses** job attribute, printer definition, or Routing exit. Multiple email addresses are separated by a comma. If the email addresses are greater than

256 characters, field XTP_MAIL_INCOM is set and field XTPIPADR contains only the email addresses that fit into this field.

Tip: IP PrintWay does not pass the email addresses of the copy or blind copy recipients to the exit.

- XTPRETNS** Period of time IP PrintWay is to retain the data set on the JES spool after a successful transmission. This time was specified in the RETAINS JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might have set a default value of zeroes. For the format of this field, see the note on page 247.
- XTPRETNF** Period of time IP PrintWay is to retain the data set on the JES spool after all transmission retries have failed. This time was specified in the RETAINF JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might have set a default value of zeroes. For the format of this field, see the note on page 247.
- XTPRTRYL** Number of times IP PrintWay is to retry transmission of the data set. This number was specified in the RETRYL JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might have set a default value of zero.
- XTPRTRYT** Period of time IP PrintWay is to wait between retries of the transmission. This time was specified in the RETRYT JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might have set a default value of zero. For the format of this field, see the note on page 247.
- XTPKEY** If XTP_ROUTING_KEY_TYPE is set to '1', contains the name of the printer definition. This name was specified in JCL, in the printer definition, or by the Routing exit. The name is padded to the right with blanks. This field is case-sensitive.
- Fields XTPDEST, XTPCLASS, and XTPFORMS are mapped to the same location as XTPKEY.
- XTPDEST** If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the DEST field of the printer definition that IP PrintWay uses to print the data set. The name is padded to the right with blanks. If the DEST field in the printer definition is not specified, this field contains blanks.
- Tip:** Use field XTP_JCL_DEST if you want to obtain the value specified in a JCL parameter.
- XTPCLASS** If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the CLASS field of the printer definition that IP PrintWay uses to print the data set. If the CLASS field in the printer definition is not specified, this field contains blanks.
- Tip:** Use field XTP_JCL_CLASS if you want to obtain the value specified in a JCL parameter.
- XTPFORMS** If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the FORMS field of the printer definition that IP PrintWay uses to print the data set. The name is padded to the right with

blanks. If the FORMS field in the printer definition is not specified, this field contains blanks.

Tip: Use field XTP_JCL_FORMS if you want to obtain the value specified in a JCL parameter.

XTPRECFM	<p>Basic mode: The record format as indicated in the JFCRECFM field in the job file control block (JFCB).</p> <p>Extended mode: Indicates the type of carriage control characters:</p> <table border="0"> <thead> <tr> <th style="text-align: left;">Value</th> <th style="text-align: left;">Meaning</th> </tr> </thead> <tbody> <tr> <td>X'04'</td> <td>A - American national standard (ASA) control characters</td> </tr> <tr> <td>X'02'</td> <td>M - Machine code control characters</td> </tr> <tr> <td>X'00'</td> <td>No control characters</td> </tr> </tbody> </table>	Value	Meaning	X'04'	A - American national standard (ASA) control characters	X'02'	M - Machine code control characters	X'00'	No control characters
Value	Meaning								
X'04'	A - American national standard (ASA) control characters								
X'02'	M - Machine code control characters								
X'00'	No control characters								
XTPMRECL	Maximum record length specified on the LRECL parameter of the OUTPUT JCL statement. For data sets that NetSpool and Print Interface allocate on the JES spool, the value is 32756.								
XTPSWBTP	Address of the SWBTU for the data set. For more information, see "Scheduler work block text unit (SWBTU)" on page 244.								
XTPSWBTL	Length of the SWBTU.								
XTPJSPAP	<p>The address of a full or partial JSPA. For more information about this field, see "JES job separator page data area (IAZJSPA)" on page 243.</p> <ul style="list-style-type: none"> • IP PrintWay basic mode: Address of the JSPA (IAZJSPA) for the data set being processed. • IP PrintWay extended mode: Address of a partial JSPA (IAZJSPA) for the data set being processed. 								
XTPCOPYS	Number of copies of the data set. This number was specified in JCL, by the Routing exit, or JES provided a default value.								
XTPIPORT	Port number of the printer or print server. This number was specified in the PORTNO JCL parameter or in the printer definition.								
XTP_MISC_FLAGS	<p>Flags containing miscellaneous information:</p> <table border="0"> <thead> <tr> <th style="text-align: left;">Flag</th> <th style="text-align: left;">Meaning when set to B'1'</th> </tr> </thead> <tbody> <tr> <td>XTP_MAIL_INCOM</td> <td>The list of email addresses for the primary recipients exceeds 256 characters. Therefore, field XTPIPADR does not contain the complete list of email addresses.</td> </tr> <tr> <td>XTP_TRANS_DOC_HEADER</td> <td>Indicates that IP PrintWay translates the document header to the printer code page specified in the printer definition.</td> </tr> <tr> <td>XTP_TRANS_DOC_TRAILER</td> <td>Indicates that IP PrintWay translates the document trailer to the printer code page specified in the printer definition.</td> </tr> </tbody> </table>	Flag	Meaning when set to B'1'	XTP_MAIL_INCOM	The list of email addresses for the primary recipients exceeds 256 characters. Therefore, field XTPIPADR does not contain the complete list of email addresses.	XTP_TRANS_DOC_HEADER	Indicates that IP PrintWay translates the document header to the printer code page specified in the printer definition.	XTP_TRANS_DOC_TRAILER	Indicates that IP PrintWay translates the document trailer to the printer code page specified in the printer definition.
Flag	Meaning when set to B'1'								
XTP_MAIL_INCOM	The list of email addresses for the primary recipients exceeds 256 characters. Therefore, field XTPIPADR does not contain the complete list of email addresses.								
XTP_TRANS_DOC_HEADER	Indicates that IP PrintWay translates the document header to the printer code page specified in the printer definition.								
XTP_TRANS_DOC_TRAILER	Indicates that IP PrintWay translates the document trailer to the printer code page specified in the printer definition.								

XTP_ADDR_TYPE

This field identifies the transmission protocol and the contents of fields XTPIPADR, XTPPRTQU and XTPIPORT:

Value	Protocol	XTPIPADR	XTPPRTQU	XTPIPORT
0	LPR	IP address	Print queue name	
1	IPP	URL		
2	Direct sockets	IP address		Port number
3	VTAM	LU name	Logmode	
4	Email	email addresses		

XTP_ROUTING_KEY_TYPE

An indicator of how IP PrintWay selected the printer definition it will use to obtain printing options:

Value	Meaning
-------	---------

- | | |
|---|--|
| 0 | IP PrintWay selected the printer definition by matching the DEST, CLASS, and FORMS fields specified in JCL statements with the values in the printer definition. |
| 1 | IP PrintWay selected the printer definition using the printer definition name specified on the OUTPUT JCL statement or specified by NetSpool or Print Interface. |

If field XTP_FILTERING = 2, this field contains 1.

XTP_FILTERING

This field identifies when IP PrintWay processes data two times (as two separate data sets) because the **resubmit for filtering** option is selected in the printer definition. This field applies only to IP PrintWay basic mode. IP PrintWay extended mode always sets this field to 0 because the **resubmit for filtering** option does not apply to extended mode. For more information, see “Resubmit for filtering considerations (basic mode)” on page 240.

Value	Meaning
-------	---------

- | | |
|---|---|
| 0 | The resubmit for filtering option is not selected in the printer definition. |
| 2 | The resubmit for filtering option is selected and either this is the second data set IP PrintWay processes or the job was originally submitted directly to Print Interface and IP PrintWay processes only one data set. Print Interface has already transformed the input data into the format required by the printer. For example, if the afpxpcl filter is specified in the printer definition, the input data is now in PCL format. |

XTP_JCL_DEST

The DEST value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the DEST value in the printer definition. Otherwise, it is the DEST value in the OUTPUT JCL statement unless the **resubmit for filtering** option is selected in the printer definition, in which case this field contains the DEST value specified in the printer definition. The value is padded to the right with blanks.

XTP_JCL_CLASS

The CLASS value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the CLASS value in the printer definition. Otherwise, it is the CLASS value specified in a JCL statement unless the **resubmit for filtering** option is selected in the printer definition, in which case this field contains the CLASS value specified in the printer definition. The value is padded to the right with blanks.

XTP_JCL_FORMS

The FORMS value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the FORMS value in the printer definition. Otherwise, it is the FORMS value in a JCL statement. The value is padded to the right with blanks.

XTP_ORIGINATING_MVS_SYSTEM

The system name (left-justified and padded with blanks) of the MVS image where the data set originated. This field contains the same value as the SSS2SYS field in the IAZSSS2 control block. For information about the SSS2SYS field, see *z/OS MVS Using the Subsystem Interface*. This field contains the system name only if you run IP PrintWay extended mode (XTPVERS=1). IP PrintWay basic mode does not set this field.

Tip: In a JES2 environment, the XTP_ORIGINATING_JES2_SYSTEM field contains the member name of the individual JES2 system where the data set originated.

XTP_ORIGINATING_JES2_SYSTEM

Contains the member name (left-justified and padded with blanks) of the individual JES2 system where the data set originated. This field contains the same value as the SSS2MBR field in the IAZSSS2 control block. For information about the SSS2MBR field, see *z/OS MVS Using the Subsystem Interface*. This field contains the member name only if you run IP PrintWay extended mode (XTPVERS=1). IP PrintWay basic mode does not set this field. On a JES3 system, this field contains blanks.

XTPOPTNS

Transmission options. These options were specified in the printer definition or in the components named in the PRTOPTNS JCL parameter. For the field names, see the ANFUEXTP macro. For each transmission option, there are two fields:

- The first field contains the length of the value. A number greater than zero indicates that the second field contains a valid value.
- The second field contains the value of the option.

XTP_DOC_HEADER_PTR

The address of a 292-byte area. The area contains the document header, if one exists. This header was specified in the printer definition or in the Begin Data Set exit in a previous call to the exit for the data set. IP PrintWay sends the document header to the printer at the beginning of the data set and at the beginning of each subsequent copy of the data set.

Tips:

- XTP_DOC_HEADER_LENGTH contains the length of the document header.

- XTP_TRANS_DOC_HEADER indicates whether IP PrintWay translates the document header to the printer code page before sending the header to the printer.

XTP_DOC_TRAILER_PTR

The address of a 292-byte area. The area contains the document trailer, if one exists. This trailer was specified in the printer definition or in the Begin Data Set exit in a previous call to the exit for the data set. IP PrintWay sends the document trailer to the printer at the end of the data set and at the end of each subsequent copy of the data set.

Tips:

- XTP_DOC_TRAILER_LENGTH contains the length of the document trailer.
- XTP_TRANS_DOC_TRAILER indicates whether IP PrintWay translates the document trailer to the printer code page before sending the trailer to the printer.

XTP_DOC_HEADER_LENGTH

The length of the document header. XTP_DOC_HEADER_PTR contains the address of the document header. -1 indicates that no document header exists.

XTP_DOC_TRAILER_LENGTH

The length of the document trailer. XTP_DOC_TRAILER_PTR contains the address of the document trailer. -1 indicates that no document trailer exists.

XTPMLINL (extended mode)

The length of the longest line in the data set including unprintable characters, such as all control characters and transparent data. This field contains a value only for data sets that NetSpool converts to line data format and allocates on the JES spool. For all other data sets, including data sets that NetSpool converts to PCL format, this field contains zeroes. The Begin Data Set exit can use the line length to add PCL commands that select an appropriate font and page orientation (portrait or landscape) for the data.

Note: IP PrintWay basic mode always sets this field to zeroes.

Output from the Begin Data Set exit

At return, restore the contents of all registers. Do not set a return code in register 15.

The exit can set these fields in the ANFUEXTP control block:

- | | |
|-----------------|---|
| XTPWORK1 | Address of a work area, or zeroes. For information about how to use work areas, see “Programming considerations” on page 242. |
| XTPWORK2 | Address of a work area, or zeroes. For information about how to use work areas, see “Programming considerations” on page 242. |
| XTPRCFLG | Flags to request that IP PrintWay add a new record and specify characteristics of the new record. |

Tips:

- IP PrintWay sets all flags to B'0' before calling the exit.
- If the exit does not add a record on the first call to the exit (XTPRCEXT=0), IP PrintWay does not call the exit again

regardless of whether flag XTPRLAST is set. That is, the exit must add at least one record for the exit to be called more than once.

- If the exit adds a record (XTPRCEXT=1) on any call to the exit, set the XTPRLAST flag if it is the last record. This tells IP PrintWay to not call the exit again.

Flag Meaning when set to B'1'

XTPRCTRN Translate data in the new record to the printer code page specified in the printer definition. Also, IP PrintWay adds a CRLF instruction (X'0D0A') to the end of the data to start a new line. This can cause a blank line to be printed if the new record does not contain any data.

Tip: To avoid a blank line when the exit adds a printer instruction, code the printer instruction in hexadecimal and do not set this flag.

XTPRCCC The new record contains a carriage control character in column 1.

XTPRLAST Do not call this exit again to add another record to the beginning of this data set.

XTPRCEXT Add the record whose address is in field XTPERPTR.

XTPRCDBL The new record contains double-byte characters.

XTPMSGFL Flags indicating the destination of a message created by this exit. IP PrintWay sets these flags to B'0' before calling the exit. Put the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN.

Flag Meaning when set to B'1'

XTPM2CON IP PrintWay basic mode writes the message to the console. IP PrintWay extended mode writes the message to both the console and the Infoprint Server common message log.

XTPM2MDS IP PrintWay basic mode writes the message to the IP PrintWay message-log data set. IP PrintWay extended mode writes the message to both the Infoprint Server common message log and the console.

XTPMSGLN Length of the message whose address is in field XTPMSGP.

XTPMSGP Address of message created by this exit. IP PrintWay adds message ID ANFM155I (basic mode) or AOP3003I (extended mode) to the beginning of this message.

XTPERPTR Address of the record to be added.

XTPERLEN Length of the record to be added.

XTP_MISC_FLAGS

Flags containing miscellaneous information:

Flag Meaning when set to B'1'

XTP_TRANS_DOC_HEADER Indicates that IP PrintWay

translates the document header to the printer code page that is specified in the printer definition before sending the header to the printer.

XTP_TRANS_DOC_TRAILER

Indicates that IP PrintWay translates the document trailer to the printer code page that is specified in the printer definition before sending the trailer to the printer.

XTP_DOC_HEADER_PTR

The address of a 292-byte area that contains the document header. IP PrintWay sends the document header to the printer at the beginning of the data set and at the beginning of each subsequent copy of the data set. Specify the length of the document header in the XTP_DOC_HEADER_LENGTH field.

Rules:

1. Do not change the address in this field. You can change the data in the 292-byte area that this field points to.
2. The document header cannot exceed 292 bytes.
3. Set the XTP_TRANS_DOC_HEADER flag to indicate whether IP PrintWay should translate the document header to the printer code page specified in the printer definition.
4. Do not specify these values, which are allowed only in a printer definition:
<ESC> <CR> <FF> <LF> <SP>

XTP_DOC_TRAILER_PTR

The address of a 292-byte area that contains the document trailer. IP PrintWay sends the document trailer to the printer at the end of the data set and at the end of each subsequent copy of the data set.

Rules:

1. Do not change the address in this field. You can change the data in the 292-byte area that this field points to.
2. Specify the length of the document trailer in XTP_DOC_TRAILER_LENGTH.
3. The document trailer cannot exceed 292 bytes.
4. Set XTP_TRANS_DOC_TRAILER to indicate whether IP PrintWay must translate the document trailer to the printer code page specified in the printer definition.
5. Do not specify these values, which are allowed only in a printer definition:
<ESC> <CR> <FF> <LF> <SP>

XTP_DOC_HEADER_LENGTH

The length of the document header. XTP_DOC_HEADER_PTR contains the address of the document header. If you do not want to change the document header specified in the printer definition, do not change the value in this field. If you do not want a document header at all, specify -1.

XTP_DOC_TRAILER_LENGTH

The length of the document trailer. XTP_DOC_TRAILER_PTR contains the address of the document trailer. If you do not want to change the document trailer specified in the printer definition, do not change the value in this field. If you do not want a document trailer at all, specify -1.

XTPOPTNS

The transmission options. For the field names, see the ANFUEXTP macro. For each option, there are two fields:

- The first field contains the length of the value. A number greater than zero indicates that the next field contains a valid value.
- The second field contains the value for the option.

If you add or change a value for one of the options, also set the length in the length field.

Examples -- Begin Data Set exit

1. To add one record to the beginning of the data set, set these fields in ANFUEXTP:
 - Set flag XTPRCEXT=1 to request that IP PrintWay add a record.
 - Put the address of the new record in field XTPERPTR.
 - Put the length of the new record in field XTPERLEN.
 - Set flags to indicate the characteristics of the record you are adding:
 - Set flag XTPRCTRN=1 if you want IP PrintWay to translate the record to the printer code page specified in the printer definition.
 - Set flag XTPRCCC=1 if the record contains a carriage-control character.
 - Set flag XTPRCDBL=1 if the record contains double-byte characters.
 - Set flag XTPRLAST=1 to indicate this is the last record you are adding.
2. To add or change a value for a transmission option, specify these values in the XTPOPTNS field of the ANFUEXTP control block:
 - Specify the value in the field for that transmission option. For the field names and valid values, see the ANFUEXTP macro.
 - Specify the length of the value in the corresponding length field. A zero length indicates that the value is not specified.

For example, to request that the printer or print server's LPD print a banner (separator) page, set these fields:

- Set XTPO_LPR_PRINT_BANNER to 1.
- Set XTPO_LPR_PRINT_BANNER to the length of XTPO_LPR_PRINT_BANNER.

If you specify a character string, for example in the XTPO_LPR_BANNER_CLASS field, set the length to the actual length of the character string.

3. To transmit the data set with no changes, simply return control to IP PrintWay.
4. To request that IP PrintWay write a message to the console:
 - Set XTPM2CON=1 in the ANFUEXTP control block.
 - Put the address of the message in field XTPMSGP.
 - Put the length of the message in field XTPMSGLN.
5. To specify a document header, use these instructions to move the document header from area DOC_H into the area that XTP_DOC_HEADER_PTR points

to, and set the length of the document header in field XTP_DOC_HEADER_LENGTH. These instructions assume that registers 2, 3, and 5 are available.

- Copy some ASCII PCL commands into the document header area and set the length:

```
L    R2,XTP_DOC_HEADER_PTR
LA   R3,DOC_H
MVC  0(L'DOC_H,R2),0(R3)
LA   R5,L'DOC_H
STH  R5,XTP_DOC_HEADER_LENGTH
```

- Ensure that the XTP_TRANS_DOC_HEADER flag is 0 because the document header contains ASCII PCL commands that should not be translated to the code page specified in the **Printer codepage** field in the printer definition:

```
OI   XTP_MISC_FLAGS,XTP_TRANS_DOC_HEADER
XI   XTP_MISC_FLAGS,XTP_TRANS_DOC_HEADER
```

```
DOC_H DC X'1B451B266C31301B2873313748'
```

Tips:

- Use similar instructions to specify a document trailer.
- If the document header or trailer is longer than 256 bytes, change the MVC instruction to MVCL. Do not specify a header or trailer that is larger than 292 bytes.

Tip: See also the sample exits distributed with IP PrintWay and described in “IP PrintWay sample exits” on page 299.

Record exit

The Record exit lets you:

- Inspect each record in a data set before the data set is transmitted.
- Add one or more records, replace a record, or delete a record. The exit can add more than one record at the same location in the data set by requesting that IP PrintWay call the exit multiple times.

If the exit adds or replaces a record, the exit can specify whether the record:

- Should be translated to ASCII or EBCDIC
 - Contains carriage control characters
 - Contains double-byte characters
- Inspect the routing parameters and the transmission options for a data set. However, the exit cannot change them.
 - Specify a message for IP PrintWay to write. IP PrintWay assigns message ID ANFM155I (basic mode) or AOP3003I (extended mode) to the message.

Tips:

- The administrator specifies the name of the Record exit in the printer definition. Therefore, you can write different Record exits for different data sets if the data sets use different printer definitions.
- IP PrintWay does not call the Record exit in these situations:
 - When a record was added by the Begin Data Set exit or the End Data Set exit.
 - When the **resubmit for filtering** option is selected in the printer definition and IP PrintWay is transmitting the data set to Print Interface for filtering. For more information, see “Resubmit for filtering considerations (basic mode)” on page 240.

Input to the Record exit

At entry to the Record exit, the registers contain:

Register 1	Pointer to the address of the ANFUEXTP control block.
Register 13	Address of an 18-word save area for saving the caller's registers.
Register 14	Return address.
Register 15	Entry-point address.

These fields in ANFUEXTP are valid when the Record exit is called. The format of the ANFUEXTP control block is shown in Figure 23 on page 245.

XTPSAVE	A register save area for use by the exit.
XTPWORK1	Address of a work area, or zeroes. For information about how to use work areas, see "Programming considerations" on page 242.
XTPWORK2	Address of a work area, or zeroes. For information about how to use work areas, see "Programming considerations" on page 242.
XTPDSNAM	Fully qualified name of the data set being processed, padded to the right with blanks.
XTPVERS	Indicates whether IP PrintWay basic mode or extended mode called the exit and constructed the ANFUEXTP control block.

Value Meaning

0	IP PrintWay basic mode called the exit.
1	IP PrintWay extended mode called the exit.

Tip: Your exit can use this field to determine whether fields XTP_ORIGINATING_MVS_SYSTEM and XTP_ORIGINATING_JES2_SYSTEM contain valid values. These fields contain valid values if XTPVERS=1 (IP PrintWay extended mode called the exit).

XTPUSRID	User ID of the job submitter, padded to the right with blanks, with these exceptions: <ul style="list-style-type: none">• If NetSpool allocated the data set on the JES spool, this field contains the owner embedded in the print data or specified in the printer definition. If none is specified, this is the ID of the user who started the Infoprint Server daemons.• If Print Interface allocated the data set on the JES spool and the print request was not submitted from a z/OS system, this field contains the user ID from the remote system of the person who submitted the print request.
-----------------	---

XTPDSFLG Flags containing data set information:

Flag Meaning when set to B'1'

XTPDSCC	This data set contains carriage control characters.
XTPDSJNW	This data set is the JESNEWS data set. This flag applies only to IP PrintWay basic mode.

XTPRCFLG Flags containing record information:

Flag Meaning when set to B'1'

XTPRDFST	This is the first call to the exit for this record.
-----------------	---

XTPPRPTR	Address of the record currently in the data set (original record).
XTPPRLN	Length of record whose address is in field XTPPRPTR.
XTPPQLN	Length of the value in field XTPPRTQU.
XTPIALEN	Length of the value in field XTPIPADR.
XTPPRTQU	Name of the printer's or print server's queue specified in the PRTQUEUE JCL parameter, in a printer definition, or by the Routing exit; or the VTAM logmode specified in a printer definition or by the Routing exit. Field XTP_ADDR_TYPE identifies which value is in this field.
XTPIPADR	Field XTP_ADDR_TYPE identifies which of these values this field contains: <ul style="list-style-type: none"> • Host name or IP address of the printer or print server specified in the DEST=IP: JCL parameter, in the printer-ip-address job attribute, in the printer definition, or by the Routing exit • URL of the printer or print server specified in the printer definition or by the Routing exit • LU name of the printer specified in the printer definition or by the Routing exit • Email addresses of the primary recipients specified in the MAILTO JCL parameter, mail-to-addresses job attribute, printer definition, or Routing exit. Multiple email addresses are separated by a comma. If the email addresses are greater than 256 characters, field XTP_MAIL_INCOM is set and field XTPIPADR contains only the email addresses that fit into this field. <p>Tip: IP PrintWay does not pass the email addresses of the copy or blind copy recipients to the exit.</p>
XTPRETNS	Period of time IP PrintWay is to retain the data set on the JES spool after a successful transmission. This time was specified in the RETAINS JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might have set a default value of zeroes. For the format of this field, see the note on page 247.
XTPRETNF	Period of time IP PrintWay is to retain the data set on the JES spool after all transmission retries have failed. This time was specified in the RETAINF JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might have set a default value of zeroes. For the format of this field, see the note on page 247.
XTPRTRYL	Number of times IP PrintWay is to retry transmission of the data set. This number was specified in the RETRYL JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might have set a default value of zero.
XTPRTRYT	Period of time IP PrintWay is to wait between retries of the transmission. This time was specified in the RETRYT JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might have set a default value of zero. For the format of this field, see the note on page 247.
XTPKEY	If XTP_ROUTING_KEY_TYPE is set to '1', contains the name of the printer definition. This name was specified in JCL, in the printer definition, or by the Routing exit. The name is padded to the right with blanks.

Fields XTPDEST, XTPCLASS, and XTPFORMS are mapped to the same location as XTPKEY.

XTPDEST If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the DEST field of the printer definition that IP PrintWay uses to print the data set. The name is padded to the right with blanks. If the DEST field in the printer definition is not specified, this field contains blanks.

Tip: Use field XTP_JCL_DEST if you want to obtain the value specified in JCL parameters.

XTPCLASS If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the CLASS field of the printer definition that IP PrintWay uses to print the data set. If the CLASS field in the printer definition is not specified, this field contains blanks.

Tip: Use field XTP_JCL_CLASS if you want to obtain the value specified in JCL parameters.

XTPFORMS If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the FORMS field of the printer definition that IP PrintWay uses to print the data set. The name is padded to the right with blanks. If the FORMS field in the printer definition is not specified, this field contains blanks.

Tip: Use field XTP_JCL_FORMS if you want to obtain the value specified in JCL parameters.

XTPSWBTP Address of the SWBTU for the data set. For more information, see "Scheduler work block text unit (SWBTU)" on page 244.

XTPSWBTL Length of the SWBTU.

XTPJSPAP The address of a full or partial JSPA. For more information about this field, see "JES job separator page data area (IAZJSPA)" on page 243.

- IP PrintWay basic mode: Address of the JSPA (IAZJSPA) for the data set being processed.
- IP PrintWay extended mode: Address of a partial JSPA (IAZJSPA) for the data set being processed.

XTPIPORT Port number of the printer or print server. This number was specified in the PORTNO JCL parameter or in the printer definition.

XTP_MISC_FLAGS

Flags containing miscellaneous information:

Flag	Meaning when set to B'1'
XTP_FIRST_REC	This is the first record in the exit.
XTP_MAIL_INCOM	The list of email addresses for the primary recipients exceeds 256 characters. Therefore, field XTPIPADR does not contain the complete list of email addresses.

XTP_ADDR_TYPE

This field identifies the transmission protocol and the contents of fields XTPIPADR, XTPPRTQU and XTPIPORT:

Value	Protocol	XTPIPADR	XTPPRTQU	XTPIPORT
0	LPR	IP address	Print queue name	
1	IPP	URL		
2	Direct sockets	IP address		Port number
3	VTAM	LU name	Logmode	
4	Email	email addresses		

XTP_ROUTING_KEY_TYPE

An indicator of how IP PrintWay selected the printer definition it will use to obtain printing options:

Value	Meaning
-------	---------

- | | |
|---|--|
| 0 | IP PrintWay selected the printer definition by matching the DEST, CLASS, and FORMS fields specified in JCL statements with the values in the printer definition. |
| 1 | IP PrintWay selected the printer definition using the printer definition name specified on the OUTPUT JCL statement or specified by NetSpool or Print Interface. |

XTP_JCL_DEST

The DEST value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the DEST value in the printer definition. Otherwise, it is the DEST value in the OUTPUT JCL statement unless the **resubmit for filtering** option is selected in the printer definition, in which case this field contains the DEST value specified in the printer definition. The value is padded to the right with blanks.

XTP_JCL_CLASS

The CLASS value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the CLASS value in the printer definition. Otherwise, it is the CLASS value specified in a JCL statement unless the **resubmit for filtering** option is selected in the printer definition, in which case this field contains the CLASS value specified in the printer definition. The value is padded to the right with blanks.

XTP_JCL_FORMS

The FORMS value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the FORMS value in the printer definition. Otherwise, it is the FORMS value in a JCL statement. The value is padded to the right with blanks.

XTP_ORIGINATING_MVS_SYSTEM

The system name (left-justified and padded with blanks) of the MVS image where the data set originated. This field contains the same value as the SSS2SYS field in the IAZSSS2 control block. For

information about the SSS2SYS field, see *z/OS MVS Using the Subsystem Interface*. This field contains the system name only if you run IP PrintWay extended mode (XTPVERS=1). IP PrintWay basic mode does not set this field.

Tip: In a JES2 environment, the XTP_ORIGINATING_JES2_SYSTEM field contains the member name of the individual JES2 system where the data set originated.

XTP_ORIGINATING_JES2_SYSTEM

Contains the member name (left-justified and padded with blanks) of the individual JES2 system where the data set originated. This field contains the same value as the SSS2MBR field in the IAZSSS2 control block. For information about the SSS2MBR field, see *z/OS MVS Using the Subsystem Interface*. This field contains the member name only if you run IP PrintWay extended mode (XTPVERS=1). IP PrintWay basic mode does not set this field. On a JES3 system, this field contains blanks.

XTPOPTNS

Transmission options. These options were specified in the printer definition or in the components named in the PRTOPTNS JCL parameter. For the field names, see the ANFUEXTP macro. For each transmission option, there are two fields:

- The first field contains the length of the value. A number greater than zero indicates that the second field contains a valid value.
- The second field contains the value of the option.

Output from the Record exit

At return, restore the contents of all registers. Do not set a return code in register 15.

The exit can set these fields in the ANFUEXTP control block:

XTPWORK1 Address of a work area, or zeroes. For information about how to use work areas, see “Programming considerations” on page 242.

XTPWORK2 Address of a work area, or zeroes. For information about how to use work areas, see “Programming considerations” on page 242.

XTPRCFLG Flags to request that IP PrintWay add the original record or a new record and to specify characteristics of the record.

Tips:

- IP PrintWay sets all flags to B'0' before calling the exit.
- If the exit does not add a record on the first call to the exit (XTPRCORG=0 and XTPRCEXT=0), IP PrintWay does not call the exit again regardless of whether flag XTPRLAST is set. That is, the exit must add at least one record for the exit to be called more than once.
- If the exit adds a record (XTPRCORG=1 or XTPRCEXT=1) on any call to the exit, set the XTPRLAST flag if this is the last record to be added. This tells IP PrintWay to not call the exit again for this record.

Flag Meaning when set to B'1'

XTPRCTRN Translate data in the new record to the printer code page specified in the printer definition. Also, IP PrintWay adds a CRLF instruction (X'0D0A') to the

end of the data to start a new line. This can cause a blank line to be printed if the new record does not contain any data.

Tips:

- To avoid a blank line when the exit adds a printer instruction, code the printer instruction in hexadecimal and do not set this flag.
- This flag does not apply to the original record.

XTPRCCC	The new record contains a carriage control character in column 1. This flag does <i>not</i> apply to the original record.
XTPRCORG	Add the original record to the data set.
XTPRLAST	Do not call this exit again for this record.
XTPRCEXT	Add the new record whose address is in field XTPERPTR. Do not set this flag and flag XTPRCORG at the same time. If flag XTPRCORG is set, IP PrintWay ignores this flag.
XTPRCDBL	The new record contains double-byte characters. This flag does <i>not</i> apply to the original record.
XTPMSGFL	Flags indicating the destination of a message created by this exit. Put the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN.
Flag	Meaning when set to B'1'
XTPM2CON	IP PrintWay basic mode writes the message to the console. IP PrintWay extended mode writes the message to both the console and the Infoprint Server common message log.
XTPM2MDS	IP PrintWay basic mode writes the message to the IP PrintWay message-log data set. IP PrintWay extended mode writes the message to both the Infoprint Server common message log and the console.
XTPERPTR	Address of the record to be added.
XTPERLEN	Length of the record to be added.
XTPPRPTR	Address of the original record in the data set. You can change the contents of the original record.
XTPPRLEN	Length of the original record whose address is in field XTPPRPTR.
XTPMSGLN	Length of the message whose address is in field XTPMSGP.
XTPMSGP	Address of message created by this exit. IP PrintWay adds message ID ANFM155I (basic mode) or AOP3003I (extended mode) to the beginning of this message.

Examples -- Record exit

1. To add a new record after the original record in the data set, set these fields in ANFUEXTP:
 - Set flag XTPRCORG=1 to request that IP PrintWay transmit the original record.

- Set flag XTPRLAST=0 to indicate that IP PrintWay is to call the exit again to add another record.

On the next call to the Record exit, set these fields in ANFUEXTP:

- Set flag XTPRCORG=0 to request that IP PrintWay not add the original record.
 - Set flag XTPRCEXT=1 to request that IP PrintWay add a new record.
 - Put the address of the new record in field XTPERPTR.
 - Put the length of the new record in field XTPERLEN.
 - Set flags to indicate the characteristics of the new record:
 - Set flag XTPRCTR=1 if you want IP PrintWay to translate the record to the printer code page specified in the printer definition.
 - Set flag XTPRCCC=1 if the record contains a carriage-control character.
 - Set flag XTPRCDBL=1 if the record contains double-byte characters.
 - Set flag XTPRLAST=1 to indicate this is the last record you are adding.
2. To replace the original record, set these fields in ANFUEXTP:
 - Set flag XTPRCORG=1 to request that IP PrintWay add the original record.
 - Modify the record whose address is in field XTPPRPTR. However, do not increase the length of the record in the original location. To make the record longer, create the record in another location and put the address of the record in field XTPPRPTR.
 - If the length of the record has changed, put the new record length in XTPPRLN.
 - Set flag XTPRLAST=1 to indicate this is the last record to be added.
 3. To make no changes to the original record:
 - Set flag XTPRCORG=1 to request that IP PrintWay add the original record.
 - Set flag XTPRLAST=1 to indicate this is the last record to be added.
 4. To delete the original record, simply return control to IP PrintWay.
 5. To request that IP PrintWay write a message to the console:
 - Set XTPM2CON=1 in the ANFUEXTP control block.
 - Put the address of the message in field XTPMSGP.
 - Put the length of the message in field XTPMSGLN.

Tip: See also the sample exits distributed with IP PrintWay and described in “IP PrintWay sample exits” on page 299.

End Data Set exit

The End Data Set exit lets you:

- Add one or more records to the end of a data set. You might want to create a separator page or to append fonts to send to the printer.
 - The exit can add more than one record by requesting that IP PrintWay call the exit multiple times for the same data set.
 - On a separator page, the exit can print job and data set information contained in the scheduler work block text unit (SWBTU). See “Scheduler work block text unit (SWBTU)” on page 244.
 - The exit can create a job-level separator page by adding a separator page only for the last data set in each print job (JES output group). The XTPDSEJB flag in the ANFUEXTP control block indicates whether the data set is the last data set in a print job.

- The exit can use the Block Letter program (ANFUBLK) to build block letters suitable for printing on a separator page. For more information, see “Using the block letter program (ANFUBLK)” on page 298.
- When IP PrintWay sends multiple data sets to a printer in a concatenation (**dataset-grouping=concatenate-job** attribute is specified in the printer definition), it sends records that the End Data Set exit adds only after the last data set in the concatenation. It does not send the records after each data set in the concatenation.

Tip: If you run IP PrintWay basic mode, the administrator can set the **concatenation-separators=document** printer attribute in the IP PrintWay FSS definition to send the records after each data set in the concatenation. However, this can cause extra an separator page to print after each copy. For information, see *z/OS Infoprint Server Operation and Administration*.

- Inspect the transmission options for a data set. However, the exit cannot change them.
- Inspect the routing parameters for a data set. However, the exit cannot change them.
- Specify a message for IP PrintWay to write. IP PrintWay assigns message ID ANFM155I (basic mode) or AOP3003I (extended mode) to the message.

Tips:

- The administrator specifies the name of the End Data Set exit in the printer definition. Therefore, you can write different End Data Set exits for different data sets if the data sets use different printer definitions.
- IP PrintWay basic mode does not call the End Data Set exit when the **resubmit for filtering** option is selected in the printer definition and IP PrintWay transmits the data set to Print Interface for filtering. For more information, see “Resubmit for filtering considerations (basic mode)” on page 240.
- To specify a document trailer that contains printer commands, use the Begin Data Set exit.
- IP PrintWay sends the document trailer to the printer before any records that the End Data Set exit adds.

Input to the End Data Set exit

At entry to the End Data Set exit, the registers contain:

- Register 1** Pointer to the address of the ANFUEXTP control block.
- Register 13** Address of an 18-word save area for saving the caller's registers.
- Register 14** Return address.
- Register 15** Entry-point address.

These fields in ANFUEXTP are valid when the End Data Set exit is called. The format of the ANFUEXTP control block is shown in Figure 23 on page 245.

- XTPSAVE** A register save area for use by the exit.
- XTPWORK1** Address of a work area, or zeroes. For information about how to use work areas, see “Programming considerations” on page 242.
- XTPWORK2** Address of a work area, or zeroes. For information about how to use work areas, see “Programming considerations” on page 242.
- XTPDSNAM** Fully qualified name of the data set being processed, padded to the right with blanks.

XTPVERS	Indicates whether IP PrintWay basic mode or extended mode called the exit and constructed the ANFUEXTP control block.												
	Value Meaning												
	0 IP PrintWay basic mode called the exit.												
	1 IP PrintWay extended mode called the exit.												
	Tip: Your exit can use this field to determine whether fields XTP_ORIGINATING_MVS_SYSTEM and XTP_ORIGINATING_JES2_SYSTEM contain valid values. These fields contain valid values if XTPVERS=1 (IP PrintWay extended mode called the exit).												
XTPUSRID	User ID of the job submitter, padded to the right with blanks, with these exceptions: <ul style="list-style-type: none"> • If NetSpool allocated the data set on the JES spool, this field contains the owner embedded in the print data or specified in the printer definition. If none is specified, this is the ID of the user who started the Infoprint Server daemons. • If Print Interface allocated the data set on the JES spool and the print request was not submitted from a z/OS system, this field contains the user ID from the remote system of the person who submitted the print request. 												
XTPDSFLG	Flags containing data set information: <table border="0" style="margin-left: 20px;"> <thead> <tr> <th style="text-align: left; padding-right: 20px;">Flag</th> <th>Meaning when set to B'1'</th> </tr> </thead> <tbody> <tr> <td style="padding-right: 20px;">XTPDSCC</td> <td>This data set contains carriage control characters.</td> </tr> <tr> <td style="padding-right: 20px;">XTPDSJOB</td> <td>This is the first data set in the print job (JES output group).</td> </tr> <tr> <td style="padding-right: 20px;">XTPDSJNW</td> <td>This data set is the JESNEWS data set. This flag applies only to IP PrintWay basic mode.</td> </tr> <tr> <td style="padding-right: 20px;">XTPDSEJB</td> <td>This is the last data set in the print job (JES output group).</td> </tr> <tr> <td style="padding-right: 20px;">XTPDSCON</td> <td>This data set is part of an IP PrintWay concatenation. IP PrintWay sets this flag when the Concatenate job option is selected in the Dataset grouping field of the printer definition and IP PrintWay is transmitting this data set with other data sets in the same transmission.</td> </tr> </tbody> </table>	Flag	Meaning when set to B'1'	XTPDSCC	This data set contains carriage control characters.	XTPDSJOB	This is the first data set in the print job (JES output group).	XTPDSJNW	This data set is the JESNEWS data set. This flag applies only to IP PrintWay basic mode.	XTPDSEJB	This is the last data set in the print job (JES output group).	XTPDSCON	This data set is part of an IP PrintWay concatenation. IP PrintWay sets this flag when the Concatenate job option is selected in the Dataset grouping field of the printer definition and IP PrintWay is transmitting this data set with other data sets in the same transmission.
Flag	Meaning when set to B'1'												
XTPDSCC	This data set contains carriage control characters.												
XTPDSJOB	This is the first data set in the print job (JES output group).												
XTPDSJNW	This data set is the JESNEWS data set. This flag applies only to IP PrintWay basic mode.												
XTPDSEJB	This is the last data set in the print job (JES output group).												
XTPDSCON	This data set is part of an IP PrintWay concatenation. IP PrintWay sets this flag when the Concatenate job option is selected in the Dataset grouping field of the printer definition and IP PrintWay is transmitting this data set with other data sets in the same transmission.												
XTPRCFLG	Flags containing data set information: <table border="0" style="margin-left: 20px;"> <thead> <tr> <th style="text-align: left; padding-right: 20px;">Flag</th> <th>Meaning when set to B'1'</th> </tr> </thead> <tbody> <tr> <td style="padding-right: 20px;">XTPRDFST</td> <td>This is the first call to the exit for this data set.</td> </tr> </tbody> </table>	Flag	Meaning when set to B'1'	XTPRDFST	This is the first call to the exit for this data set.								
Flag	Meaning when set to B'1'												
XTPRDFST	This is the first call to the exit for this data set.												
XTPPQLEN	Length of the value in field XTPPRTQU.												
XTPIALEN	Length of the value in field XTPIPADR.												
XTPPRTQU	Name of the printer's or print server's queue specified in the PRTQUEUE JCL parameter, in a printer definition, or by the Routing exit; or the VTAM logmode specified in a printer definition or by the Routing exit. Field XTP_ADDR_TYPE identifies which value is in this field.												

XTPIPADR	<p>Field XTP_ADDR_TYPE identifies which of these values this field contains:</p> <ul style="list-style-type: none"> • Host name or IP address of the printer or print server specified in the DEST=IP: JCL parameter, printer-ip-address job attribute, in a printer definition, or by the Routing exit • URL of the printer or print server specified in the printer definition or by the Routing exit • LU name of the printer specified in the printer definition or by the Routing exit • Email addresses of the primary recipients specified in the MAILTO JCL parameter, mail-to-addresses job attribute, printer definition, or Routing exit. Multiple email addresses are separated by a comma. If the email addresses are greater than 256 characters, field XTP_MAIL_INCOM is set and field XTPIPADR contains only the email addresses that fit into this field. <p>Tip: IP PrintWay does not pass the email addresses of the copy or blind copy recipients to the exit.</p>
XTPRETNS	<p>Period of time IP PrintWay is to retain the data set on the JES spool after a successful transmission. This time was specified in the RETAINS JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might have set a default value of zeroes. For the format of this field, see the note on page 247.</p>
XTPRETNF	<p>Period of time IP PrintWay is to retain the data set on the JES spool after all transmission retries have failed. This time was specified in the RETAINF JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might have set a default value of zeroes. For the format of this field, see the note on page 247.</p>
XTPRTRYL	<p>Number of times IP PrintWay is to retry transmission of the data set. This time was specified in the RETRYL JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might have set a default value of zero.</p>
XTPRTRYT	<p>Period of time IP PrintWay is to wait between retries of the transmission. This time was specified in the RETRYT JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might have set a default value of zero. For the format of this field, see the note on page 247.</p>
XTPKEY	<p>If XTP_ROUTING_KEY_TYPE is set to '1', this field contains the name of the printer definition. This name was specified in JCL, in the printer definition, or by the Routing exit. The name is padded to the right with blanks.</p> <p>Fields XTPDEST, XTPCLASS, and XTPFORMS are mapped to the same location as XTPKEY.</p>
XTPDEST	<p>If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the DEST field of the printer definition that IP PrintWay uses to print the data set. The name is padded to the right with blanks. If the DEST field in the printer definition is not specified, this field contains blanks.</p>

Tip: Use field XTP_JCL_DEST if you want to obtain the value specified on the OUTPUT JCL statement.

XTPCLASS If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the CLASS field of the printer definition that IP PrintWay uses to print the data set. If the CLASS field in the printer definition is not specified, this field contains blanks.

Tip: Use field XTP_JCL_CLASS if you want to obtain the value specified in JCL statements.

XTPFORMS If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the FORMS field of the printer definition that IP PrintWay uses to print the data set. The name is padded to the right with blanks. If the FORMS field in the printer definition is not specified, this field contains blanks.

Tip: Use field XTP_JCL_FORMS if you want to obtain the value specified in JCL statements.

XTPRECFM **Basic mode:** The record format as indicated in the JFCRECFM field in the job file control block (JFCB).

Extended mode: Indicates the type of carriage control characters:

Value Meaning

X'04' A - American national standard (ASA) control characters

X'02' M - Machine code control characters

X'00' No control characters

XTPMRECL Maximum record length specified on the LRECL parameter of the OUTPUT JCL statement. For data sets that NetSpool and Print Interface allocate on the JES spool, the value is 32756.

XTPSWBTP Address of the SWBTU for the data set. For more information, see "Scheduler work block text unit (SWBTU)" on page 244.

XTPSWBTL Length of the SWBTU.

XTPJSPAP The address of a full or partial JSPA. For more information about this field, see "JES job separator page data area (IAZJSPA)" on page 243.

- IP PrintWay basic mode: Address of the JSPA (IAZJSPA) for the data set being processed.
- IP PrintWay extended mode: Address of a partial JSPA (IAZJSPA) for the data set being processed.

XTPCOPYS Number of copies of the data set. This number was specified in JCL, by the Routing exit, or JES provided a default value.

XTPIPORT Port number on the target system. This number was specified in the PORTNO JCL parameter or in the printer definition.

XTP_MISC_FLAGS

Flags containing miscellaneous information:

Flag	Meaning when set to B'1'
------	--------------------------

XTP_TRANS_DOC_HEADER Indicates that IP PrintWay translates the document header to the printer code page specified in the printer definition.

XTP_TRANS_DOC_TRAILER Indicates that IP PrintWay translates the document trailer to the printer code page specified in the printer definition.

XTP_FILTERING

This field identifies when IP PrintWay basic mode processes data two times (as two separate data sets) because the **resubmit for filtering** option is selected in the printer definition. This field applies only to IP PrintWay basic mode. IP PrintWay extended mode always sets this field to 0 because the **resubmit for filtering** option does not apply to extended mode. For more information, see “Resubmit for filtering considerations (basic mode)” on page 240.

Value Meaning

- | | |
|----------|---|
| 0 | The resubmit for filtering option is not selected in the printer definition. |
| 2 | The resubmit for filtering option is selected and either this is the second data set IP PrintWay basic mode processes or the job was originally submitted directly to Print Interface and IP PrintWay basic mode processes only one data set. Print Interface has already transformed the input data into the format required by the printer. For example, if the afpxpcl filter is specified in the printer definition, the input data is now in PCL format. |

XTP_JCL_DEST

The DEST value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the DEST value in the printer definition. Otherwise, it is the DEST value in the OUTPUT JCL statement unless the **resubmit for filtering** option is selected in the printer definition, in which case this field contains the DEST value specified in the printer definition. The value is padded to the right with blanks.

XTP_JCL_CLASS

The CLASS value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the CLASS value in the printer definition. Otherwise, it is the CLASS value specified in a JCL statement unless the **resubmit for filtering** option is selected in the printer definition, in which case this field contains the CLASS value specified in the printer definition. The value is padded to the right with blanks.

XTP_JCL_FORMS

The FORMS value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the FORMS value in the printer definition. Otherwise, it is the FORMS value in a JCL statement. The value is padded to the right with blanks.

XTP_ORIGINATING_MVS_SYSTEM

The system name (left-justified and padded with blanks) of the MVS image where the data set originated. This field contains the same value as the SSS2SYS field in the IAZSSS2 control block. For information about the SSS2SYS field, see *z/OS MVS Using the Subsystem Interface*. This field contains the system name only if you run IP PrintWay extended mode (XTPVERS=1). IP PrintWay basic mode does not set this field.

Tip: In a JES2 environment, the XTP_ORIGINATING_JES2_SYSTEM field contains the member name of the individual JES2 system where the data set originated.

XTP_ORIGINATING_JES2_SYSTEM

Contains the member name (left-justified and padded with blanks) of the individual JES2 system where the data set originated. This field contains the same value as the SSS2MBR field in the IAZSSS2 control block. For information about the SSS2MBR field, see *z/OS MVS Using the Subsystem Interface*. This field contains the member name only if you run IP PrintWay extended mode (XTPVERS=1). IP PrintWay basic mode does not set this field. On a JES3 system, this field contains blanks.

XTPOPTNS

Transmission options. These options were specified in the printer definition or in the components named in the PRTOPTNS JCL parameter. For the field names, see the ANFUEXTP macro. For each transmission option, there are two fields:

- The first field contains the length of the value. A number greater than zero indicates that the second field contains a valid value.
- The second field contains the value of the option.

If you change a value for one of the options, also set the length of the value in the length field that precedes that field.

XTP_DOC_HEADER_PTR

The address of a 292-byte area. The area contains the document header, if one exists. This header was specified in the printer definition or in the Begin Data Set exit. IP PrintWay sends the document header to the printer at the beginning of the data set and at the beginning of each subsequent copy of the data set.

Tips:

- XTP_DOC_HEADER_LENGTH contains the length of the document header.
- XTP_TRANS_DOC_HEADER indicates whether IP PrintWay translates the document header to the printer code page specified in the printer definition.

XTP_DOC_TRAILER_PTR

The address of a 292-byte area. The area contains the document trailer, if one exists. This trailer was specified in the printer definition or in the Begin Data Set exit. IP PrintWay sends the document trailer to the printer at the end of the data set and at the end of each subsequent copy of the data set.

Tips:

- XTP_DOC_TRAILER_LENGTH contains the length of the document trailer.
- XTP_TRANS_DOC_TRAILER indicates whether IP PrintWay translates the document trailer to the printer code page specified in the printer definition.

XTP_DOC_HEADER_LENGTH

The length of the document header. XTP_DOC_HEADER_PTR contains the address of the document header. -1 indicates that no document header exists.

XTP_DOC_TRAILER_LENGTH

The length of the document trailer. XTP_DOC_TRAILER_PTR contains the address of the document trailer. -1 indicates that no document trailer exists.

Output from the End Data Set exit

At return, restore the contents of all registers. Do not set a return code in register 15.

The exit can set these fields in the ANFUEXTP control block:

- XTPWORK1** Address of a work area, or zeroes. For information about how to use work areas, see “Programming considerations” on page 242.
- XTPWORK2** Address of a work area, or zeroes. For information about how to use work areas, see “Programming considerations” on page 242.
- XTPRCFLG** Flags to request that IP PrintWay add a record and to specify characteristics of the new record.

Tips:

- IP PrintWay sets all flags to B'0' before calling the exit.
- If the exit does not add a record on the first call to the exit (XTPRCEXT=0), IP PrintWay does not call the exit again regardless of whether flag XTPRLAST is set. That is, the exit must add at least one record for the exit to be called more than once.
- If the exit adds a record (XTPRCEXT=1) on any call to the exit, set the XTPRLAST flag if it is the last record. This tells IP PrintWay to not call the exit again.

Flag

Meaning when set to B'1'

XTPRCTRN Translate data in the new record to the printer code page specified in the printer definition. Also, IP PrintWay adds a CRLF instruction (X'0D0A') to the end of the data to start a new line. This can cause a blank line to be printed if the new record does not contain any data.

Tip: To avoid a blank line when the exit adds a printer instruction, code the printer instruction in hexadecimal and do not set this flag.

XTPRCCC The new record contains a carriage control character in column 1.

	XTPRLAST	Do not call this exit again to add another record to the end of this data set.
	XTPRCEXT	Add the record whose address is in field XTPERPTR.
	XTPRCDBL	The new record contains double-byte characters.
XTPMSGFL		Flags indicating the destination of a message created by this exit. IP PrintWay sets these flags to B'0' before calling the exit. Put the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN.
	Flag	Meaning when set to B'1'
	XTPM2CON	IP PrintWay basic mode writes the message to the console. IP PrintWay extended mode writes the message to both the console and the Infoprint Server common message log.
	XTPM2MDS	IP PrintWay basic mode writes the message to the IP PrintWay message-log data set. IP PrintWay extended mode writes the message to both the Infoprint Server common message log and the console.
XTPERPTR		Address of the record to be added.
XTPERLEN		Length of the record to be added.
XTPMSGLN		Length of the message whose address is in field XTPMSGP.
XTPMSGP		Address of message created by this exit. IP PrintWay adds message ID ANFM155I (basic mode) or AOP3003I (extended mode) to the beginning of this message.

Examples -- End Data Set exit

1. To add one record to the end of the data set, set these fields in ANFUEXTP:
 - Set flag XTPRCEXT=1 to request that IP PrintWay add a record.
 - Put the address of the new record in field XTPERPTR.
 - Put the length of the new record in field XTPERLEN.
 - Set flags to indicate the characteristics of the record you are adding:
 - Set flag XTPRCTRN=1 if you want IP PrintWay to translate the record to ASCII or EBCDIC.
 - Set flag XTPRCCC=1 if the record contains a carriage-control character.
 - Set flag XTPRCDBL=1 if the record contains double-byte characters.
 - Set flag XTPRLAST=1 to indicate this is the last record you are adding.
2. To transmit the data set with no changes, simply return control to IP PrintWay.
3. To request that IP PrintWay write a message to the console:
 - Set XTPM2CON=1 in the ANFUEXTP control block.
 - Put the address of the message in field XTPMSGP.
 - Put the length of the message in field XTPMSGLN.

Also see the sample exits distributed with IP PrintWay and described in "IP PrintWay sample exits" on page 299.

SMF exit (ANFUXSMF)

The SMF exit lets you:

- Replace the SMF type 6 records written by IP PrintWay.
- Suppress the SMF type 6 records written by IP PrintWay.
- Specify a message for IP PrintWay to write. IP PrintWay assigns message ID ANFM155I (basic mode) or AOP3003I (extended mode) to the message.

System management facilities (SMF) is an optional control program that collects information to assist you in evaluating system use. If you do not write an SMF exit, IP PrintWay writes a standard SMF type 6 record for each data set sent to the printer.

For more information about the SMF type 6 record that IP PrintWay writes, see *z/OS Infoprint Server Operation and Administration*. For more information about SMF and about macro IFASMFR, which maps the SMF type 6-record, see *z/OS MVS System Management Facilities (SMF)*.

IP PrintWay calls the same SMF exit for all data sets. You can install different SMF exits for different IP PrintWay basic mode FSSs. For information, see “Installing IP PrintWay exits” on page 300.

You might want to write an SMF exit for these reasons:

- To include additional information in the SMF type 6 record.
- To suppress the SMF record for JESNEWS data sets. Flag XTPDSJNW indicates whether this is a JESNEWS data set. This flag applies only to IP PrintWay basic mode.
- To suppress the first SMF record that IP PrintWay basic mode writes for a data set when the **resubmit for filtering** option is selected in the printer definition. IP PrintWay writes one SMF record when it transmits the data set to Print Interface for filtering (that is, to transform data). This record contains the IP address of the z/OS system. IP PrintWay writes a second SMF record when it transmits the data set to the target printer. This record contains the IP address of the target printer.

For more information, see the description of field XTP_FILTERING in “Input to the SMF exit” and see “Resubmit for filtering considerations (basic mode)” on page 240. For an example of how to suppress the first SMF record, see “Examples -- SMF exit” on page 290.

Tip: Fields in macro IFASMFR that SVC 83 provides are not available to this exit.

Input to the SMF exit

At entry to the SMF exit, the registers contain:

- | | |
|--------------------|--|
| Register 1 | Pointer to the address of the ANFUEXTP control block. |
| Register 13 | Address of an 18-word save area for saving the caller's registers. |
| Register 14 | Return address. |
| Register 15 | Entry-point address. |

These fields in ANFUEXTP are valid when the SMF exit is called. The format of the ANFUEXTP control block is shown in Figure 23 on page 245.

- | | |
|----------------|---|
| XTPSAVE | A register save area for use by the exit. |
|----------------|---|

XTPWORK1	Address of a work area, or zeroes. For information about how to use work areas, see “Programming considerations” on page 242.										
XTPWORK2	Address of a work area, or zeroes. For information about how to use work areas, see “Programming considerations” on page 242.										
XTPDSNAM	Fully qualified name of the data set being processed, padded to the right with blanks.										
XTPVERS	Indicates whether IP PrintWay basic mode or extended mode called the exit and constructed the ANFUEXTP control block. Value Meaning 0 IP PrintWay basic mode called the exit. 1 IP PrintWay extended mode called the exit. Tip: Your exit can use this field to determine whether fields XTP_ORIGINATING_MVS_SYSTEM and XTP_ORIGINATING_JES2_SYSTEM contain valid values. These fields contain valid values if XTPVERS=1 (IP PrintWay extended mode called the exit).										
XTPUSRID	User ID of the job submitter, padded to the right with blanks, with these exceptions: <ul style="list-style-type: none"> • If NetSpool allocated the data set on the JES spool, this field contains the owner embedded in the print data or specified in the printer definition. If none is specified, this is the ID of the user who started the Infoprint Server daemons. • If Print Interface allocated the data set on the JES spool and the print request was not submitted from a z/OS system, this field contains the user ID from the remote system of the person who submitted the print request. 										
XTPDSFLG	Flags containing data set information: <table border="0" style="width: 100%;"> <thead> <tr> <th style="text-align: left;">Flag</th> <th style="text-align: left;">Meaning when set to B'1'</th> </tr> </thead> <tbody> <tr> <td>XTPDSJOB</td> <td>This is the first data set in the print job (JES output group).</td> </tr> <tr> <td>XTPDSJNW</td> <td>This data set is the JESNEWS data set. This flag applies only to IP PrintWay basic mode.</td> </tr> <tr> <td>XTPDSEJB</td> <td>This is the last data set in the print job (JES output group).</td> </tr> <tr> <td>XTPDSCON</td> <td>This data set is part of an IP PrintWay concatenation. IP PrintWay sets this flag when the Concatenate job option is selected in the Dataset grouping field of the printer definition and IP PrintWay is transmitting this data set with other data sets in the same transmission.</td> </tr> </tbody> </table>	Flag	Meaning when set to B'1'	XTPDSJOB	This is the first data set in the print job (JES output group).	XTPDSJNW	This data set is the JESNEWS data set. This flag applies only to IP PrintWay basic mode.	XTPDSEJB	This is the last data set in the print job (JES output group).	XTPDSCON	This data set is part of an IP PrintWay concatenation. IP PrintWay sets this flag when the Concatenate job option is selected in the Dataset grouping field of the printer definition and IP PrintWay is transmitting this data set with other data sets in the same transmission.
Flag	Meaning when set to B'1'										
XTPDSJOB	This is the first data set in the print job (JES output group).										
XTPDSJNW	This data set is the JESNEWS data set. This flag applies only to IP PrintWay basic mode.										
XTPDSEJB	This is the last data set in the print job (JES output group).										
XTPDSCON	This data set is part of an IP PrintWay concatenation. IP PrintWay sets this flag when the Concatenate job option is selected in the Dataset grouping field of the printer definition and IP PrintWay is transmitting this data set with other data sets in the same transmission.										
XTPPRPTR	Address of the SMF record for the data set.										
XTPPRLEN	Length of the SMF record for the data set.										
XTPKEY	If XTP_ROUTING_KEY_TYPE is set to '1', contains the name of the printer definition. This name was specified in JCL, in the printer definition, or by the Routing exit. The name is padded to the right with blanks.										

Fields XTPDEST, XTPCLASS, and XTPFORMS are mapped to the same location as XTPKEY.

XTPDEST If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the DEST field of the printer definition that IP PrintWay uses to print the data set. The name is padded to the right with blanks. If the DEST field in the printer definition is not specified, this field contains blanks.

Tip: Use field XTP_JCL_DEST if you want to obtain the value specified on the OUTPUT JCL statement.

XTPCLASS If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the CLASS field of the printer definition that IP PrintWay uses to print the data set. If the CLASS field in the printer definition is not specified, this field contains blanks.

Tip: Use field XTP_JCL_CLASS if you want to obtain the value specified in a JCL statements.

XTPFORMS If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the FORMS field of the printer definition that IP PrintWay uses to print the data set. The name is padded to the right with blanks. If the FORMS field in the printer definition is not specified, this field contains blanks.

Tip: Use field XTP_JCL_FORMS if you want to obtain the value specified in a JCL statement.

XTPSWBTP Address of the SWBTU for the data set. For more information, see “Scheduler work block text unit (SWBTU)” on page 244.

XTPSWBTL Length of the SWBTU.

XTPJSPAP The address of a full or partial JSPA. For more information about this field, see “JES job separator page data area (IAZJSPA)” on page 243.

- IP PrintWay basic mode: Address of the JSPA (IAZJSPA) for the data set being processed.
- IP PrintWay extended mode: Address of a partial JSPA (IAZJSPA) for the data set being processed.

XTPCOPYS Number of copies of the data set. This number was specified in JCL, by the Routing exit, or JES provided a default value.

XTP_ROUTING_KEY_TYPE

An indicator of how IP PrintWay selected the printer definition it will use to obtain printing options:

Value	Meaning
--------------	----------------

- | | |
|----------|--|
| 0 | IP PrintWay selected the printer definition by matching the DEST, CLASS, and FORMS fields specified on JCL statements with the values in the printer definition. |
| 1 | IP PrintWay selected the printer definition using the printer definition name specified on the OUTPUT JCL statement or specified by NetSpool or Print Interface. |

If field XTP_FILTERING = 2, this field contains 1.

XTP_FILTERING

This field identifies when IP PrintWay basic mode processes data two times (as two separate data sets) because the **resubmit for filtering** option is selected in the printer definition. This field applies only to IP PrintWay basic mode. IP PrintWay extended mode always sets this field to 0 because the **resubmit for filtering** option does not apply to extended mode. For more information, see "Resubmit for filtering considerations (basic mode)" on page 240.

Value Meaning

- | | |
|---|--|
| 0 | The resubmit for filtering option is not selected in the printer definition. |
| 1 | The resubmit for filtering option is selected and this is the first data set. IP PrintWay basic mode is transmitting this data set to Print Interface for filtering. The SMF record contains the IP address of the z/OS system. |
| 2 | The resubmit for filtering option is selected and either this is the second data set IP PrintWay basic mode processed or the job was originally submitted directly to Print Interface and IP PrintWay processed the data set only one time. IP PrintWay basic mode is transmitting this data set to the target printer. The SMF record contains the IP address of the target printer. |

XTP_JCL_DEST

The DEST value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the DEST value in the printer definition. Otherwise, it is the DEST value in the OUTPUT JCL statement unless the **resubmit for filtering** option is selected in the printer definition, in which case this field contains the DEST value specified in the printer definition. The value is padded to the right with blanks.

XTP_JCL_CLASS

The CLASS value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the CLASS value in the printer definition. Otherwise, it is the CLASS value specified in a JCL statement unless the **resubmit for filtering** option is selected in the printer definition, in which case this field contains the CLASS value specified in the printer definition. The value is padded to the right with blanks.

XTP_JCL_FORMS

The FORMS value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the FORMS value in the printer definition. Otherwise, it is the FORMS value in a JCL statement. The value is padded to the right with blanks.

XTP_ORIGINATING_MVS_SYSTEM

The system name (left-justified and padded with blanks) of the MVS image where the data set originated. This field contains the same value as the SSS2SYS field in the IAZSSS2 control block. For information about the SSS2SYS field, see *z/OS MVS Using the Subsystem Interface*. This field contains the system name only if you run IP PrintWay extended mode (XTPVERS=1). IP PrintWay basic mode does not set this field.

Tip: In a JES2 environment, the XTP_ORIGINATING_JES2_SYSTEM field contains the member name of the individual JES2 system where the data set originated.

XTP_ORIGINATING_JES2_SYSTEM

Contains the member name (left-justified and padded with blanks) of the individual JES2 system where the data set originated. This field contains the same value as the SSS2MBR field in the IAZSSS2 control block. For information about the SSS2MBR field, see *z/OS MVS Using the Subsystem Interface*. This field contains the member name only if you run IP PrintWay extended mode (XTPVERS=1). IP PrintWay basic mode does not set this field. On a JES3 system, this field contains blanks.

Output from the SMF exit

At return, restore the contents of all registers. Do not set a return code in register 15.

The exit can set these fields in the ANFUEXTP control block:

XTPWORK1 Address of a work area, or zeroes. For information about how to use work areas, see “Programming considerations” on page 242.

XTPWORK2 Address of a work area, or zeroes. For information about how to use work areas, see “Programming considerations” on page 242.

XTPRCFLG Flag to request that IP PrintWay write or suppress the SMF record. IP PrintWay sets this flag to B'0' before calling the exit.

Flag **Meaning when set to B'1'**

XTPRCORG Write the SMF record. Otherwise, IP PrintWay suppresses the record.

XTPMSGFL Flags indicating the destination of a message created by this exit. IP PrintWay sets these flags to B'0' before calling the exit. Put the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN.

Flag **Meaning when set to B'1'**

XTPM2CON IP PrintWay basic mode writes the message to the console. IP PrintWay extended mode writes the message to both the console and the Infoprint Server common message log.

XTPM2MDS IP PrintWay basic mode writes the message to the IP PrintWay message-log data set. IP PrintWay extended mode writes the message to both the Infoprint Server common message log and the console.

XTPPRPTR	Address of the SMF type 6 record to be written for the data set. You can change the contents and length of the original record.
XTPPRLN	Length of the SMF type 6 record whose address is in field XTPPRPTR.
XTPMSGLN	Length of the message whose address is in field XTPMSGP.
XTPMSGP	Address of message created by this exit. IP PrintWay adds message ID ANFM155I (basic mode) or AOP3003I (extended mode) to the beginning of this message.

Examples -- SMF exit

- To replace the SMF record, set these fields in ANFUEXTP:
 - Set flag XTPRCORG=1 to request that IP PrintWay transmit the original record.
 - Modify the SMF record whose address is in field XTPPRPTR.
 - If the length of the SMF record has changed, put the new length in field XTPPRLN.
- To write the original SMF record, with no changes, set this field in ANFUEXTP:
 - Set flag XTPRCORG=1 to request that IP PrintWay write the original SMF record.
- To suppress the SMF record, simply return control to IP PrintWay.
- To suppress the first of two SMF records IP PrintWay basic mode writes when the **resubmit for filtering** option is selected in the printer definition:
 - If XTP_FILTERING=1, return control to IP PrintWay. This action causes IP PrintWay to suppress the first SMF record.
 - If XTP_FILTERING=2, set XTPRCORG=1. This action causes IP PrintWay to write the second SMF record.
- To request that IP PrintWay write a message to the console:
 - Set XTPM2CON=1.
 - Put the address of the message in field XTPMSGP.
 - Put the length of the message in field XTPMSGLN.

Also see the sample exits distributed with IP PrintWay and described in “IP PrintWay sample exits” on page 299.

Message exit (ANFUXMSG) — basic mode

The Message exit applies only to IP PrintWay basic mode.

The Message exit lets you suppress a message or modify the ID or text of a message that IP PrintWay sends to the IP PrintWay message-log data set, including messages created by another IP PrintWay exit.

IP PrintWay calls the same Message exit for all data sets. You can install different Message exits for different IP PrintWay FSSs. For information, see “Installing IP PrintWay exits” on page 300.

Tip: In the Message exit, you can request that IP PrintWay send a message to the IP PrintWay message-log data set. However, you cannot request that IP PrintWay send a message to the console. To send a message to the console, code the WTO macro in the Message exit.

Input to the Message exit

At entry to the Message exit, the registers contain:

- Register 1** Pointer to the address of the ANFUEXTP control block.
- Register 13** Address of an 18-word save area for saving the caller's registers.
- Register 14** Return address.
- Register 15** Entry-point address.

These fields in ANFUEXTP are valid when the Message exit is called. The format of the ANFUEXTP control block is shown in Figure 23 on page 245.

- XTPSAVE** A register save area for use by the exit.
- XTPMSGLN** Length of the message whose address is in field XTPMSGP.
- XTPMSGP** Address of the message to be sent. This message has a message ID, followed by one blank and the message text.

XTP_JCL_DEST

The DEST value with which the output data set was allocated on the JES spool. This field contains binary zeroes if the message is not associated with a data set.

If NetSpool or Print Interface allocated the data set, this is the DEST value in the printer definition. Otherwise, it is the DEST value in the OUTPUT JCL statement unless the **resubmit for filtering** option is selected in the printer definition, in which case this field contains the DEST value specified in the printer definition. The value is padded to the right with blanks.

XTP_JCL_CLASS

The CLASS value with which the output data set was allocated on the JES spool. This field contains blanks if the message is not associated with a data set.

If NetSpool or Print Interface allocated the data set, this is the CLASS value in the printer definition. Otherwise, it is the CLASS value specified in a JCL statement unless the **resubmit for filtering** option is selected in the printer definition, in which case this field contains the CLASS value specified in the printer definition. The value is padded to the right with blanks.

XTP_JCL_FORMS

The FORMS value with which the output data set was allocated on the JES spool. This field contains blanks if the message is not associated with a data set.

If NetSpool or Print Interface allocated the data set, this is the FORMS value in the printer definition. Otherwise, it is the FORMS value in a JCL statement. The value is padded to the right with blanks.

Output from the Message exit

At return, restore the contents of all registers. Do not set a return code in register 15.

The exit can set these fields in the ANFUEXTP control block:

- XTPMSGFL** Flag indicating whether IP PrintWay is to write the message. IP PrintWay sets this flag to B'0' before calling the exit.

Flag	Meaning when set to B'1'
XTPM2MDS	Write the message to the IP PrintWay message-log data set.
XTPMSGLN	Length of the message whose address is in field XTPMSGP.
XTPMSGP	Address of the message to be sent. This message should have a message ID, followed by the message text.

Examples -- Message exit

- To write the message unchanged to the IP PrintWay message-log data set, set this field in ANFUEXTP:
 - Set flag XTPM2MDS=1 to request that IP PrintWay write the message to the IP PrintWay message-log data set.
- To suppress the message, simply return control to IP PrintWay.
- To replace the message, set these fields in ANFUEXTP:
 - Set XTPM2MDS=1 in the ANFUEXTP control block to write the message to the IP PrintWay message-log data set.
 - Modify the message whose address is in field XTPMSGP. Alternatively, put the address of another message in field XTPMSGP.
 - If the length of the message has changed, put the new length in field XTPMSGLN.

Also see the sample exits distributed with IP PrintWay and described in "IP PrintWay sample exits" on page 299.

Response Notification exit (ANFUXRSP)

The Response Notification exit lets you take an action based on the success or failure of the transmission, as indicated by the Response Notification code. For example, the exit can send a message to either the operator's console or IP PrintWay message-log data set, or both. The exit can also initiate actions outside of IP PrintWay by calling modules that are external to IP PrintWay.

IP PrintWay calls the same response exit for all data set transmissions. You can install different response exits for different IP PrintWay FSSs. For information, see "Installing IP PrintWay exits" on page 300.

Input to the Response Notification exit

Register 1	Pointer to the address of the ANFUEXTP control block.
Register 13	Address of an 18-word save area for saving the caller's registers.
Register 14	Return address.
Register 15	Entry point address.

These input fields in ANFUEXTP are valid when the Response Notification exit is called. The format of the ANFUEXTP control block is shown in Figure 23 on page 245.

XTPSAVE	A register save area for use by this exit.
XTPWORK1	Address of a work area, or zeroes. For information about how to use work areas, see "Programming considerations" on page 242.
XTPWORK2	Address of a work area, or zeroes. For information about how to use work areas, see "Programming considerations" on page 242.

XTPDSNAM	Fully qualified name of the data set being processed, padded to the right with blanks.																
XTPUSRID	<p>User ID of the job submitter, padded to the right with blanks, with these exceptions:</p> <ul style="list-style-type: none"> • If NetSpool allocated the data set on the JES spool, this field contains the owner embedded in the print data or specified in the printer definition. If none is specified, this is the ID of the user who started the Infoprint Server daemons. • If Print Interface allocated the data set on the JES spool and the print request was not submitted from a z/OS system, this field contains the user ID from the remote system of the person who submitted the print request. 																
XTPSRCFL	<p>Flags indicating whether a parameter was specified in a JCL parameter.</p> <p>If the flag is set to '1', the indicated parameter was specified in a JCL parameter.</p> <p>If the flag is set to '0', the JCL parameter was not specified. Instead, IP PrintWay obtained the value from a printer definition. If the DEST=IP: parameter was specified in JCL, IP PrintWay obtained the value from the PrintWay default printer definition. If no PrintWay default printer definition exists, IP PrintWay set a default value.</p> <table border="0" style="width: 100%;"> <thead> <tr> <th style="text-align: left; padding-right: 20px;">Flag</th> <th>Meaning when set to B'1'</th> </tr> </thead> <tbody> <tr> <td style="padding-top: 10px;">XTPOPTS</td> <td style="padding-top: 10px;">The PRTOPTNS parameter specified the name of components in the Printer Inventory.</td> </tr> <tr> <td style="padding-top: 10px;">XTPSIPAD</td> <td style="padding-top: 10px;">The DEST=IP: parameter specified the host name or IP address of the target printer or print server.</td> </tr> <tr> <td style="padding-top: 10px;">XTPSPRTQ</td> <td style="padding-top: 10px;">The PRTQUEUE parameter specified the name of the target print queue.</td> </tr> <tr> <td style="padding-top: 10px;">XTPSRTNS</td> <td style="padding-top: 10px;">The RETAINS parameter specified the retention time after a successful transmission.</td> </tr> <tr> <td style="padding-top: 10px;">XTPSRTNF</td> <td style="padding-top: 10px;">The RETAINF parameter specified the retention time after a failed transmission.</td> </tr> <tr> <td style="padding-top: 10px;">XTPSRTYL</td> <td style="padding-top: 10px;">The RETYLF parameter specified the retry limit.</td> </tr> <tr> <td style="padding-top: 10px;">XTPSRTYT</td> <td style="padding-top: 10px;">The RETYLT parameter specified the retry time.</td> </tr> </tbody> </table>	Flag	Meaning when set to B'1'	XTPOPTS	The PRTOPTNS parameter specified the name of components in the Printer Inventory.	XTPSIPAD	The DEST=IP: parameter specified the host name or IP address of the target printer or print server.	XTPSPRTQ	The PRTQUEUE parameter specified the name of the target print queue.	XTPSRTNS	The RETAINS parameter specified the retention time after a successful transmission.	XTPSRTNF	The RETAINF parameter specified the retention time after a failed transmission.	XTPSRTYL	The RETYLF parameter specified the retry limit.	XTPSRTYT	The RETYLT parameter specified the retry time.
Flag	Meaning when set to B'1'																
XTPOPTS	The PRTOPTNS parameter specified the name of components in the Printer Inventory.																
XTPSIPAD	The DEST=IP: parameter specified the host name or IP address of the target printer or print server.																
XTPSPRTQ	The PRTQUEUE parameter specified the name of the target print queue.																
XTPSRTNS	The RETAINS parameter specified the retention time after a successful transmission.																
XTPSRTNF	The RETAINF parameter specified the retention time after a failed transmission.																
XTPSRTYL	The RETYLF parameter specified the retry limit.																
XTPSRTYT	The RETYLT parameter specified the retry time.																
XTPDSFLG	<p>Flags containing data set information:</p> <table border="0" style="width: 100%;"> <thead> <tr> <th style="text-align: left; padding-right: 20px;">Flag</th> <th>Meaning when set to B'1'</th> </tr> </thead> <tbody> <tr> <td style="padding-top: 10px;">XTPMDSCC</td> <td style="padding-top: 10px;">This data set contains carriage control characters.</td> </tr> <tr> <td style="padding-top: 10px;">XTPDSJOB</td> <td style="padding-top: 10px;">This is the first data set in the print job (JES output group).</td> </tr> <tr> <td style="padding-top: 10px;">XTPDSJNW</td> <td style="padding-top: 10px;">This data set is the JESNEWS data set.</td> </tr> <tr> <td style="padding-top: 10px;">XTPDSEJB</td> <td style="padding-top: 10px;">This is the last data set in the print job (JES output group).</td> </tr> <tr> <td style="padding-top: 10px;">XTPDSCON</td> <td style="padding-top: 10px;">This data set is part of an IP PrintWay concatenation. IP PrintWay sets this flag when the Concatenate job option is selected in the Dataset</td> </tr> </tbody> </table>	Flag	Meaning when set to B'1'	XTPMDSCC	This data set contains carriage control characters.	XTPDSJOB	This is the first data set in the print job (JES output group).	XTPDSJNW	This data set is the JESNEWS data set.	XTPDSEJB	This is the last data set in the print job (JES output group).	XTPDSCON	This data set is part of an IP PrintWay concatenation. IP PrintWay sets this flag when the Concatenate job option is selected in the Dataset				
Flag	Meaning when set to B'1'																
XTPMDSCC	This data set contains carriage control characters.																
XTPDSJOB	This is the first data set in the print job (JES output group).																
XTPDSJNW	This data set is the JESNEWS data set.																
XTPDSEJB	This is the last data set in the print job (JES output group).																
XTPDSCON	This data set is part of an IP PrintWay concatenation. IP PrintWay sets this flag when the Concatenate job option is selected in the Dataset																

grouping field of the printer definition and IP PrintWay is transmitting this data set with other data sets in the same transmission.

- XTPPQLEN** Length of the value in XTPPRTQU.
- XTPIALEN** Length of the value in XTPIPADR.
- XTPPRTQU** Field XTP_ADDR_TYPE identifies which value is in this field:
- Name of the print queue specified in the PRTQUEUE JCL parameter, in the printer definition, or by the Routing exit.
 - Name of the VTAM logmode specified in a printer definition or by the Routing exit.
- Or, the XTPPRTQU might not be available to the Response Notification exit if the Response code is X'08'.
- XTP_MISC_FLAGS** Flags containing miscellaneous information:
- | Flag | Meaning when set to B'1' |
|-----------------------|---|
| XTP_MAIL_INCOM | The list of email addresses for the primary recipients exceeds 256 characters. Therefore, field XTPIPADR does not contain the complete list of email addresses. |
- XTPIPADR** Field XTP_ADDR_TYPE identifies which of these values this field contains:
- Host name or IP address of the target printer or print server specified in the DEST=IP: JCL parameter, in the printer definition, or by the Routing exit
 - URL of the target printer specified in the printer definition or by the Routing exit
 - LU name of the target printer specified in the printer definition or by the Routing exit
 - Email addresses of the primary recipients specified in the MAILTO JCL parameter, **mail-to-addresses** job attribute, printer definition, or Routing exit. Multiple email addresses are separated by a comma. If the email addresses are greater than 256 characters, field XTP_MAIL_INCOM is set and field XTPIPADR contains only the email addresses that fit into this field.
- Tip:** IP PrintWay does not pass the email addresses of the copy or blind copy recipients to the exit.
- XTPRETNS** Period of time IP PrintWay is to retain the data set on the JES spool after a successful transmission. This time was specified in the RETAINS JCL parameter or in the printer definition. IP PrintWay might have set a default value of zeros. For the format of this field, see the note on page 247. XTPRETNS might not be available to the Response Notification exit if the Response code is X'08'.
- XTPRETNF** Period of time IP PrintWay is to retain the data set on the JES spool after an **all transmission retries** has failed. This time was specified in the RETAINF JCL parameter or in the printer definition. IP PrintWay might have set a default value of zeros. For

the format of this field, see the note on page 247. XTPRETNF might not be available to the Response Notification exit if the Response code is X'08'.

- XTPRTRYL** The number of times IP PrintWay is to retry transmission of the data set. This time was specified in the RETRYL JCL parameter or in the printer definition. IP PrintWay might have set a default value of zeros. XTPRTRYL might not be available to the Response Notification exit if the Response code is X'08'.
- XTPRTRYT** The period of time IP PrintWay is to wait between retries of the transmission. This time was specified in the RETRYT JCL parameter or in the printer definition. IP PrintWay might have set a default value of zeros. For the format of this field, see the note on page 247. XTPRTRYT might not be available to the Response Notification exit if the Response code is X'08'.
- XTPKEY** If XTP_ROUTING_KEY_TYPE is set to '1', contains the name of the printer definition. This name was specified in JCL, in the printer definition, or by the Routing exit. The name is padded to the right with blanks.
- Fields XTPDEST, XTPCLASS, and XTPFORMS are mapped to the same location as XTPKEY.
- XTPDEST** If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the DEST field of the printer definition that IP PrintWay uses to print the data set. The name is padded to the right with blanks. If the DEST field in the printer definition is not specified, this field contains blanks.
- Tip:** Use field XTP_JCL_DEST if you want to obtain the value specified on the OUTPUT JCL statement.
- XTPCLASS** If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the CLASS field of the printer definition that IP PrintWay uses to print the data set. If the CLASS field in the printer definition is not specified, this field contains blanks.
- Tip:** Use field XTP_JCL_CLASS if you want to obtain the value specified in a JCL statement.
- XTPFORMS** If XTP_ROUTING_KEY_TYPE is set to '0', this field contains the value in the FORMS field of the printer definition that IP PrintWay uses to print the data set. The name is padded to the right with blanks. If the FORMS field in the printer definition is not specified, this field contains blanks.
- Tip:** Use field XTP_JCL_FORMS if you want to obtain the value specified in a JCL statement.
- XTPRECFM** The record format as indicated in the JFCRECFM field in the job file control block (JFCB).
- XTPMRECL** Maximum record length specified on the LRECL parameter of the

OUTPUT JCL statement. For data sets that NetSpool and Print Interface allocate on the JES spool, the value is 32756.

XTPSWBTP Address of the SWBTU for the data set. For more information, see “Scheduler work block text unit (SWBTU)” on page 244.

XTPSWBTL Length of the SWBTU area.

XTPJSPAP Address of the JSPA (IAZJSPA) for the data set being processed. For more information about this field, see “JES job separator page data area (IAZJSPA)” on page 243.

XTPCOPYS Number of copies of the data set. This number was specified in JCL or JES provided a default value.

XTPRSPCD The Response Notification code.

The Response Notification code is stored in XTPRSPCD. It can contain any of these values:

Value Meaning

00 PrintWay completed a successful transmission. After the successful retain time expires, the output data set is removed from the JES spool.

04 The transmission was unsuccessful.

08 The initial processing of the data set by PrintWay was unsuccessful.

12 The transmission was unsuccessful. The retry limit is now 0. After the failure retain time expires, the output data set is removed from the JES spool.

16 Because a spool I/O error occurred, the data set is unprintable.

XTPIPORT Port number on the target system. This number was specified in the PORTNO JCL parameter or in the printer definition.

XTP_FILTERING

This field identifies when IP PrintWay basic mode processes data two times (as two separate data sets) because the **resubmit for filtering** option is selected in the printer definition. For more information, see “Resubmit for filtering considerations (basic mode)” on page 240.

Value Meaning

0 The **resubmit for filtering** option is not selected in the printer definition.

1 The **resubmit for filtering** option is selected in the printer definition and this is the first data set IP PrintWay basic mode processed. IP PrintWay basic mode transmitted this data set to Print Interface for filtering.

2 The **resubmit for filtering** option is selected in the printer definition and either this is the second data set IP PrintWay basic mode processed or the data set was originally submitted directly to Print Interface and IP PrintWay basic mode processed only one data set. IP PrintWay basic mode transmitted this data set to the target printer.

XTP_JCL_DEST

The DEST value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the DEST value in the printer definition. Otherwise, it is the DEST value in the OUTPUT JCL statement unless the **resubmit for filtering** option is selected in the printer definition, in which case this field contains the DEST value specified in the printer definition. The value is padded to the right with blanks.

XTP_JCL_CLASS

The CLASS value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the CLASS value in the printer definition. Otherwise, it is the CLASS value specified in a JCL statement unless the **resubmit for filtering** option is selected in the printer definition, in which case this field contains the CLASS value specified in the printer definition. The value is padded to the right with blanks.

XTP_JCL_FORMS

The FORMS value with which the output data set was allocated on the JES spool.

If NetSpool or Print Interface allocated the data set, this is the FORMS value in the printer definition. Otherwise, it is the FORMS value in a JCL statement. The value is padded to the right with blanks.

XTPOPTNS

Transmission options. These options were specified in the printer definition or in the components named in the PRTOPTNS JCL parameter. For the field names, see the ANFUEXTP macro. For each transmission option, there are two fields:

- The first field contains the length of the value. A number greater than zero indicates that the second field contains a valid value.
- The second field contains the value of the option.

IP PrintWay calls the same response exit for all data set transmissions. You can install different response exits for different PrintWay FSSs. For information, see “Installing IP PrintWay exits” on page 300.

Output from the Response Notification exit

At return, restore the contents of all registers. Do not set a return code in register 15.

XTPMSGFL Flags indicating the destination of a message created by this exit. IP PrintWay sets these flags to B'0' before calling the exit. Put the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN.

Flag	Meaning when set to B'1'
------	--------------------------

XTPM2CON	Write the message to the console.
----------	-----------------------------------

XTPM2MDS	Write the message to the IP PrintWay message-log data set.
----------	--

XTPMSGLN Length of message whose address is in XTPMSGP.

XTPMSGP Address of message created by this exit. IP PrintWay adds message ID ANFM155I to the beginning of this message.

Examples

1. To write a message to the IP PrintWay message log when IP PrintWay has exhausted the transmission retry limit, code the Response user exit to:
 - Check that XTPRSPCD = 12.
 - Assemble the transmission failure message.
 - Put the address of the message into XTPMSGP in the ANFUEXTP control block to point to the message.
 - Put the length of the message in field XTPMSGLN.
 - Set XTPM2MDS to B'1' to indicate that the message should be printed to the log.
2. To write a message to the operator's console when IP PrintWay has found a data set that failed pre-transmission processing:
 - Check that XTPRSPCD = 08.
 - Assemble the processing failure message.
 - Put the address of the message into XTPMSGP in the ANFUEXTP control block to point to the message.
 - Put the length of the message in field XTPMSGLN.
 - Set XTPM2CON to B'1' to indicate that the message should be printed to the console.

Using the block letter program (ANFUBLK)

The Begin Data Set exit and the End Data Set exit can use the Block Letter program (ANFUBLK) to build block letters suitable for printing on separator pages. The block letters can be straight or slanted, and centered or left-justified. They are 12 characters high, using 12 printed lines, and can be either 8 or 12 characters wide.

ANFUBLK creates one line at a time for the exit to add to the data set. The exit must call ANFUBLK multiple times to receive all lines, until the ANFUBLK program indicates that this is the last line.

ANFUBLK produces block characters using tables in ANFUBTBL. The tables produce block characters from this input. All other characters are translated to blanks.

- Uppercase letters: A-Z
- Symbols: \$, #, and @
- Numerals: 0-9
- Lowercase letters, a-z, which are translated to uppercase
- Superscript EBCDIC numerals, which are translated to numerals, 0-9

IP PrintWay provides the assembler language source code for ANFUBTBL in SYS1.SAMPLIB(ANFUBTBS). You can modify the source to create your own block letters. The comments in ANFUBTBS describe how to change the tables. After changing the tables, assemble ANFUBTBS, rename it to ANFUBTBL, and use the sample link job in SYS1.SAMPLIB(ANFULINK) to create your exit module.

Input to ANFUBLK

When calling the Block Letter program, set the registers:

Register 1 Pointer to the address of the ANFUEXTP control block.

- Register 13** Address of an 18-word save area for saving the caller's registers.
- Register 14** Return address.
- Register 15** Entry-point address.

Also, set these fields in ANFUEXTP. The format of the ANFUEXTP control block is shown in Figure 23 on page 245.

- XTPBIFLG** Flags indicating the type of block letters you want:
- | Flag | Meaning when set to B'1' |
|-----------------|---|
| XTPBSLNT | Slant letters to the right. |
| XTPBPRFM | Create narrow letters for faster printing. Narrow letters are 8 characters wide, centered on an 80-character line. Otherwise, letters are 12 characters wide, centered on a 132-character line. |
| XTPBLJST | Left justify the letters. Otherwise, letters are centered. |
| | Tip: During centering calculations, the Block Letter program ignores blanks to the right of the last character in field XTPBSTRG. |
| XTPBFRST | This is the first call to the Block Letter program to build block letters for the string whose address is in XTPBSTRG. |
- XTPBSTRG** String to convert to block letters, from 1 to 8 characters, padded to the right with blanks.

Output from ANFUBLK

ANFUBLK sets register 15 to 0, and also sets these fields in ANFUEXTP:

- XTPBLAST** Flag set to 1 if this is the last line. If this flag is set to 0, call the Block Letter program again.
- XTPBLENG** Length of the line in field XTPBOUTP.
- XTPBOUTP** The output line containing part of the block letter. The first character or each line contains a carriage control character.

IP PrintWay sample exits

IP PrintWay provides these sample exits written in assembler language in SYS1.SAMPLIB:

- ANFUXRTG changes the IP address of the target printer or print server.
- ANFUXRT1 changes the name of the components that contain transmission options.
- ANFUXRT2 changes the URL.
- ANFUXRT3 changes the IP address and print queue name.
- ANFUXRT4 changes the printer definition name to DEST, CLASS, and FORMS values.
- ANFUXBD1 adds a PCL5 printer setup string to the beginning of a data set.
- ANFUXBD2 adds a separator page containing line data to the beginning of a data set. ANFUXBD2 uses the Block Letter program (ANFUBLK) to create block letters.

- ANFUXBD3 requests printing of a banner page before the first data set in a job but not before subsequent data sets in the job.

Note: This sample exit is to be used only with the LPR printing protocol.

- ANFUXED1 adds a separator page containing line data to the end of a data set. ANFUXED1 uses the Block Letter program (ANFUBLK) to create block letters.
- ANFUXRC1 writes lines in a boustrophedon manner. That is, it writes alternate lines in opposite directions, from left to right and from right to left.
- ANFUXSMF suppresses the SMF record for the JESNEWS data set.
- ANFUXMSG suppresses messages ANFM700I and ANFM604I.
- ANFUXRSP takes action according to the Response Notification code.

You can either view the samples online or print them. You can modify a sample exit, assemble, and install it to provide your own version of the exit. Object code for the sample exits is not provided.

Installing IP PrintWay exits

To install an exit, compile and link-edit the exit:

- Name the exit:
 - For the Routing exit, use the name ANFUXRTG.
 - For the SMF exit, use the name ANFUXSMF.
 - For the Message exit, use the name ANFUXMSG.
 - For the Response Notification exit, use the name ANFUXRSP.
 - For the Begin Data Set exit, the Record exit, and the End Data Set exit, use any program name allowed by your system. Specify the exit name in one or more printer definitions.
- Link-edit the exit with the RENT and AMODE 31 specifications. For a more secure environment, do not specify AC=1. AC=1, which makes a program APF-authorized, is not required because the IP PrintWay program is APF-authorized. When you omit AC=1, the exit runs APF-authorized only when IP PrintWay calls it.
- If the Begin Data Set or End Data Set exit program calls the Block Letter program (ANFUBLK), use the sample JCL in SYS1.SAMPLIB(ANFULINK) to link-edit the exit with the Block Letter program (ANFUBLK) and the tables (ANFUBTBL).

Put your exits in a library that is in the system LNKLIST or in an APF-authorized step library. If you put the exits in a step library, also:

- **IP PrintWay extended mode:** Specify the step library in the STEPLIB environment variable in the **aopstart** EXEC. For information, see “Setting environment variables for IP PrintWay (extended mode)” on page 215.
Also, add the library to the file specified by the STEPLIBLIST statement in the BPXPRMxx PARMLIB member. For information, see *z/OS UNIX System Services Planning*.
- **IP PrintWay basic mode:** Specify the library in the STEPLIB DD statement in the IP PrintWay basic mode startup procedure. For information, see “Editing the ANFWPROC startup procedure (basic mode)” on page 321.

Modifying IP PrintWay exits

To modify an exit, replace the exit. If you run IP PrintWay basic mode, you must restart the IP PrintWay FSS. If you run IP PrintWay extended mode, you do not need to restart IP PrintWay daemons.

Updating the BPXPRMxx member of SYS1.PARMLIB (basic mode)

When you start an IP PrintWay FSS, a z/OS UNIX process is started. If you start multiple IP PrintWay FSSs, you might need to 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:

Parameter	Meaning
MAXPROCUSER	The maximum number of processes that a single z/OS UNIX user ID can have concurrently active. Guideline: Add at least 70 for Infoprint Server processes.
MAXPROCSYS	The maximum number of processes that the z/OS system allows. Guideline: Add 70 to the current value for each IP PrintWay FSS that you create. For example, if you create 10 IP PrintWay FSSs, add 700 (70 * 10) to the current value. If you also run Print Interface or NetSpool, add an additional 200 for processes used by both NetSpool and Print Interface.

Example: If you create 10 IP PrintWay FSSs, and also use Print Interface or NetSpool, specify these statements in the BPXPRMxx member:

```
MAXPROCUSER(70) /*70(Infoprint Server) */
MAXPROCSYS(1100) /*700(PrintWay basic)+200(Print Interface/NetSpool)+200(initial)*/
```

Tip: You can use the SETOMVS or SET OMVS command to dynamically increase the MAXPROCUSER and MAXPROCSYS values. However, to make a permanent change, edit the BPXPRMxx member that is used for IPLs. For more information about the BPXPRMxx member, see *z/OS MVS Initialization and Tuning Reference*.

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 system command. To permanently authorize this data set:

- Update the PROGxx member of SYS1.PARMLIB to include the data sets in the LNKLST statement.
- If the LNKAUTH parameter that is put in the IEASYSxx member of SYS1.PARMLIB does not specify LNKLST, 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 LNKLSTxx and IEAAPFxx members of SYS1.PARMLIB. However, IBM suggests updating the PROGxx member. For more information, see *z/OS MVS Initialization and Tuning Reference*.

Defining the IP PrintWay functional subsystem (basic mode)

IP PrintWay basic mode operates as a JES functional subsystem (FSS). A functional subsystem is an extension of JES that runs in its own address space. For more information about JES functional subsystems, see one of these:

- *z/OS JES2 Initialization and Tuning Guide*
- *z/OS JES3 Initialization and Tuning Guide*

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. The next sections describe planning decisions to consider before coding the JES2 or JES3 initialization statements and writing a startup procedure.

Determining how many functional subsystems to define (basic mode)

You can create one or more IP PrintWay FSSs. Follow these guidelines to determine how many FSS to create.

Guidelines:

- JES lets you create up to 2000 FSSs.
- For improved throughput and more efficient use of system resources, define several FSSs, with several FSAs per FSS.
- If you define multiple IP PrintWay FSS, consider sharing the IP PrintWay transmission-queue data set among the FSSs. For information, see “Sharing the transmission-queue data set (basic mode)” on page 316.
- 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. For information, see “Determining work-selection criteria (basic mode)” on page 303.
 - 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. For information about retaining data sets on the JES spool, see *z/OS Infoprint Server Operation and Administration*.

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 - 8 megabytes of storage below the 16 MB line, an abend can occur if an FSS retains more than 300-500 data sets.

Determining how many functional subsystem applications to define (basic mode)

You should create at least one or two IP PrintWay FSAs in each FSS. You do not need to create more FSAs because one IP PrintWay FSA can transmit data sets to many printers or email destinations. Follow these guidelines to determine how many FSAs to create per FSS.

Guidelines:

- You can define up to 64 FSAs per FSS. However, for optimal performance do not define more than 35 FSAs per FSS.
- For improved throughput, 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. An IP PrintWay FSA waits until all retries to a printer are attempted before printing to another printer.
- Determine how many jobs need to print concurrently (the burst rate), and define 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 FSA for each critical printer. In the JES FSA definition, specify unique work-selection criteria so that the FSA selects output data sets only for that one printer. For example, the JES work-selection criteria could be the DEST value (or a combination of DEST and FORMS values) that is specified in the printer definition for the printer.

In addition, the FSS for the FSA should share the same transmission-queue data set with no more than 9 other FSSs. For information, see “Sharing the transmission-queue data set (basic mode)” on page 316.

Related task: If you create more than one IP PrintWay FSS, you might need to increase the number of active z/OS UNIX processes allowed on your system. For information, see “Updating the BPXPRMxx member of SYS1.PARMLIB (basic mode)” on page 301.

Determining work-selection criteria (basic mode)

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.

Do not confuse JES work-selection criteria with the criteria that IP PrintWay uses to select printers, called “printer-selection criteria”. In most cases, JES work-selection criteria are more general, while IP PrintWay printer-selection criteria are more specific:

- JES uses work-selection criteria to select output data sets from the JES spool for FSAs. For example, you might specify class P as the JES work-selection criterion for several FSAs.
- 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. For example, if CLASS=P and DEST=MYPRT are specified on the OUTPUT JCL statement for an output data set, the IP PrintWay FSA that acquired this data set from JES selects the printer definition the administrator defined with CLASS P

and DEST MYPRRT. For more information on using DEST, CLASS, and FORMS to select a printer definition, see *z/OS Infoprint Server Operation and Administration*.

Some considerations related to selecting work-selection criteria are:

- 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.
- 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. This is because JES3 passes the form name specified as a JCL parameter to the IP PrintWay FSA only if form is a JES work-selection criterion.

If you specify form as a work-selection criterion, and also specify DYNAMIC=YES in the JES3 DEVICE statement, JES3 will not start the IP PrintWay FSA dynamically unless an output data set's form attribute (and all other attributes) match the FSA's work selection criterion. Therefore, consider specifying DYNAMIC=NO (this is the JES3 default).

- 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.

Specifying a default font and page definition for data transforms (basic mode)

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 PRT $nnnnn$ statement for the IP PrintWay FSA. If you don't 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 PRT $nnnnn$ 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. If you don't specify a default page definition, JES3 uses 6 as the default FCB name. (IP PrintWay adds a P1 prefix, so the default page definition is P16.)

When you select the **resubmit for filtering** option in a printer definition, IP PrintWay basic mode passes the JES default font and page definition names, if any, to the AFP to PCL, AFP to PDF, and AFP to PostScript transforms. If the transforms receive a JES default font or page definition name, the transforms do not use the default font or page definition name specified in the printer definition or, if none are specified, in the transform configuration file.

If you want the transforms to use the default font and page definition names that are specified in the printer definition or, if none are specified, in the transform configuration file, you must prevent JES from supplying default font and page definition names:

- **JES2:**
 - To prevent JES2 from supplying a default font name, specify UCS=0 on the JES2 PRTnnnnn statement. When UCS=0, JES2 ignores any font specified in the NIUCS parameter.
 - To prevent JES2 from supplying a default page definition name, do not specify the NIFCB parameter on the JES2 PRINTDEF statement and do not specify the FCB parameter in the JES2 PRTnnnnn statement.
- **JES3:**
 - To prevent JES3 from supplying a default font name, specify PDEFAULT=CHARS and WS=U on the JES3 DEVICE statement for the IP PrintWay FSA.
 - To prevent JES3 from supplying a default page definition name, specify PDEFAULT=FCB and WS=C on the JES3 DEVICE statement for the IP PrintWay FSA.
 - To prevent JES3 from supplying a default font and page definition name, specify PDEFAULT=(CHARS,FCB) and WS=(U,C) on the JES3 DEVICE statement for the IP PrintWay FSA.

Related information: For more information about these JES initialization statements, see one of these:

- *z/OS JES2 Initialization and Tuning Reference*
- *z/OS JES3 Initialization and Tuning Reference*

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, these 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.

- **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. For more information, see “Specifying a default font and page definition for data transforms (basic mode)” on page 304.

Sample JES2 initialization statements

Figure 24 shows sample JES2 statements for one FSS named PRINTWAY and one associated FSA named PRT123.

FSS (PRINTWAY)	PROC=ANFWPROC,AUTOSTOP=YES
PRT123	CLASS=E,FSS=PRINTWAY,MODE=FSS,PRESELECT=NO, START=NO,TRKCELL=YES,UCS=0,WS=(Q)

Figure 24. Sample JES2 initialization statements. These statements are commonly found in the initialization-deck member of SYS1.PARMLIB.

The next section describes the JES2 initialization statements shown in Figure 24, including some additional parameters that you might want to specify for an IP PrintWay FSS and FSA.

Tip: These sample statements do not include all possible JES parameters. For a complete list of JES parameters and detailed descriptions of each of the parameters, see *z/OS JES2 Initialization and Tuning Reference*.

FSS(fss_name) statement

The FSS(*fss_name*) initialization statement is optional but suggested. If you omit it, JES2 generates a default FSS when an FSA is started.

Each FSS must have a unique name, specified as a 1- to 8-character name. In the example, the FSS name is PRINTWAY. This FSS name must match the FSS name in the PRTnnnnn statement for each associated FSA.

FSS statement parameters are:

PROC=procedure_name

Specifies the name of the procedure for starting this FSS. Different FSS statements can refer to the same startup procedure. For more information, see “Customizing the IP PrintWay startup procedure (basic mode)” on page 320. This parameter is required.

AUTOSTOP=YES | NO

Specifies whether the FSS address space is to be stopped automatically if all FSAs under control of the FSS are stopped. Specifying AUTOSTOP=YES can conserve system resources. This parameter is optional. If you omit it, the default is NO.

HASPFSSM=name | HASPFSSM

Specifies the 1- to 8-character name of the load module to be loaded into the functional subsystem address space. This load module contains the various JES2-supplied FSI service routines. If your installation uses the default libraries and module names for the JES2 component, as distributed by IBM, omit this parameter or code HASPFSSM=HASPFSSM.

PRTnnnnn statement

A PRTnnnnn statement defines the FSA to JES. This section describes the PRTnnnnn parameters that you might want to specify for an IP PrintWay FSA. For a complete list of parameters, see *z/OS JES2 Initialization and Tuning Reference*.

Each FSA must have a unique name. Therefore, *nnnnn* must be a unique number from 1 to 32767. JES2 lets you define more than one FSA with one *PRTnnnnn* statement. For more information about how to define a range of FSAs with one *PRTnnnnn* statement, see *z/OS JES2 Initialization and Tuning Reference*.

If you create an FSA definition in the Printer Inventory for this FSA, you must specify this FSA name in the FSA definition. Although JES might let you use different formats to specify the FSA name in this statement, for example, *PRT(nnnnn)*, always specify the FSA name as *PRTnnnnn* in the FSA definition.

PRTnnnnn statement parameters include:

CLASS | QUEUE=(classes)

Specifies the output classes processed by this FSA. List all classes to be selected by this FSA. Do not separate each class with a comma. If you designate CL or Q as a work-selection criterion on the WS parameter, the FSA processes data sets whose class matches one of the values specified here.

FCB=fcb/pagedefinition

Specifies the 1- to 4-character name of a default FCB and page definition. IP PrintWay uses this name as the FCB and page definition for output that does not specify the FCB and PAGEDEF parameters on the OUTPUT JCL statement. Do not code the P1 prefix of the page definition name. IP PrintWay adds the P1 prefix.

If you omit this parameter, JES2 determines the default FCB and page definition name from the NIFCB parameter of the JES2 PRINTDEF statement.

Tip: Omit this parameter and also omit the NIFCB parameter if you want the AFP to PCL, AFP to PostScript, and AFP to PDF transforms to use a page definition specified in a printer definition or in the transform configuration file for output submitted to IP PrintWay with OUTPUT JCL statements. For more information, see “Specifying a default font and page definition for data transforms (basic mode)” on page 304.

FORMS=(form_names)

Specifies the 1- to 8-character form names processed by this FSA. List from one to eight different form names that can be selected by this FSA. Separate each form name with a comma.

FSS=fss_name

Specifies the name of the FSS associated with this FSA. This parameter is required.

SETUP=HALT | NOHALT

Specifies whether JES2 issues the device setup message (\$HASP190) after a change of device characteristics. This parameter does not apply to an IP PrintWay FSA.

MODE=FSS

Specifies that the FSA is managed by an FSS. This parameter is optional. The default is FSS if you code the FSS parameter.

PRESELCT=YES | NO

Specifies whether output data sets are preselected for this FSA. This parameter is optional. If you omit it, the default is YES.

Guideline: Specify *PRESELCT=NO* to make sure that data sets print in the correct order.

ROUTECD=(*destinations*)

Specifies the destinations processed by this FSA. List from one to four different destination names that can be selected by this FSA. Separate each destination name with a comma.

START=YES | NO

Specifies whether JES2 is to automatically start this FSA whenever JES2 starts. If you specify START=NO, the operator must start the FSA. Consider specifying START=NO so that TCP/IP can be started before IP PrintWay. This parameter is optional. If you omit it, START=YES is the default.

TRKCELL=YES | NO

Specifies whether track-cell despooling is to be used with this FSA. You specify the size of the track cell, in terms of buffers, in the TRKCELL parameter of the JES2 SPOOLDEF statement. For improved performance, IBM suggests that you specify TRKCELL=YES on this statement and TRKCELL=5 on the JES2 SPOOLDEF statement. The default is NO.

UCS={*font* | 0}

Specifies the 1- to 4-character member name of a coded font in a font library, not including the 2-character prefix. IP PrintWay uses this font name for output that does not specify the CHARS or UCS parameter on the OUTPUT JCL statement. IP PrintWay adds X1 to the name.

UCS=0 indicates that JES2 will not supply a default font to IP PrintWay.

If you omit this parameter, JES2 determines the default font from the NIUCS parameter of the JES2 PRINTDEF statement or, if NIUCS is not specified, uses GF10 as the default.

Tip: Specify UCS=0 if you want the AFP to PCL, AFP to PostScript, and AFP to PDF transforms to use the font specified in the printer definition or transform configuration file for jobs submitted to IP PrintWay with OUTPUT JCL statements. For more information, see “Specifying a default font and page definition for data transforms (basic mode)” on page 304.

WS=(*work_selection_criteria*)

Specifies the work-selection criteria for this FSA. Separate each value with a comma. For the valid values and defaults, see *z/OS JES2 Initialization and Tuning Reference*. For an IP PrintWay FSA, consider these values:

- | | |
|---------------|--|
| CL Q | Specifies that the FSA selects only those data sets with the same class that are specified in the CLASS parameter of this statement. |
| R | Specifies that the FSA selects only those data sets with the same destination that are specified in the ROUTECDE parameter of this statement. For considerations related to this work-selection criterion, see “Determining work-selection criteria (basic mode)” on page 303. |
| F | Specifies that the FSA selects only those data sets with the same form name that is specified in the FORMS parameter of this statement. |

This parameter is optional but suggested. If you omit it, the FSA selects output data sets for processing according to default work-selection criteria.

Tip: You might want to specify the SEP=NO parameter to prevent transmission of the JESNEWS data set.

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

Sample JES3 initialization statements

Figure 25 shows sample JES3 statements for one FSS named PRINTWAY and one associated FSA named PRT1.

```
FSSDEF,FSSNAME=PRINTWAY,PNAME=ANFWPROC,TYPE=WTR
DEVICE,DTYPE=PRTAFP1,FSSNAME=PRINTWAY,JNAME=PRT1,JUNIT=(,SYS1,UR,ON),
MODE=FSS,PDEFAULT=(CHARS,FCB),WC=(E),WS=(CL,F,C,U)
```

Figure 25. Sample JES3 initialization statements. These statements are commonly found in the initialization-deck member of SYS1.PARMLIB. Continuation characters are not shown.

The next section describes the JES3 initialization statements shown in Figure 25, including some additional parameters that you might want to specify for an IP PrintWay FSS and FSA.

Tip: These sample statements do not show all possible JES parameters. For a complete list of parameters and for detailed descriptions of each of the parameters, see *z/OS JES3 Initialization and Tuning Reference*.

FSSDEF statement

The FSSDEF initialization statement is optional but suggested. If you omit it, JES3 generates a default FSS when an FSA is started.

FSSDEF statement parameters are:

FSSNAME=*fss_name*

Specifies the name of this FSS. Each FSS must have a unique 1- to 8-character name. This FSS name must match the FSS name in the DEVICE statement for each associated FSA. This parameter is required.

MSGDEST (*dest_class* | **JES**)

Specifies the console destination class for messages concerning this FSS. This parameter is optional. If you omit it, JES sets a default.

PNAME=*procedure_name*

Specifies the name of the procedure for starting this FSS. Different FSSDEF statements can refer to the same startup procedure. For more information, see “Customizing the IP PrintWay startup procedure (basic mode)” on page 320. This parameter is required.

SYSTEM=*system_name*

Specifies the JES3 processor on which the FSS will run. This parameter is optional. If you omit it, JES determines the default from the DEVICE statement.

TERM=**YES** | **NO**

YES specifies that the FSS will terminate if the JES3 global address space is terminated by a *RETURN or *DUMP operator command. This parameter is optional. If you omit it, the default is NO.

TYPE=WTR

Specifies that the FSS is an output writer. This parameter is required.

DEVICE statement

A DEVICE statement defines each FSA to JES. This section describes DEVICE parameters that you might want to specify for an IP PrintWay FSA. For a complete list of parameters, see *z/OS JES3 Initialization and Tuning Reference*.

DGROUP=destination_name

Specifies a 1- to 8-character destination processed by this FSA. If you designate destination as a work-selection criterion on the WS parameter, this FSA selects data sets that match the value specified. This parameter is optional. Specify this parameter only if you specify destination as a work-selection criteria on the WS parameter.

DTYPE=PRTAFP1

Specify PRTAFP1 as the device type for an IP PrintWay FSA. This parameter is required.

DYNAMIC=YES[, timeout] | NO

Specifies whether JES3 is to start and stop this FSA dynamically.

YES Specifies that JES3 is to start this FSA when work is available for it. If you do not specify a *timeout* value, JES3 stops this FSA and deactivates the address space when no work is available. You can specify a *timeout* value to control when JES3 stops the FSA. For more information, see *z/OS JES3 Initialization and Tuning Reference*.

NO Specifies that the operator is to start and stop this FSA. Specify NO to keep the address space active between transmission of data sets.

This parameter is optional. If you omit it, the default is NO.

If you specify form as a work-selection criterion (WS=F), and also specify DYNAMIC=YES, JES3 does not start the IP PrintWay FSA dynamically unless an output data set's form attribute (and all other attributes) match the FSA's work selection criterion. Therefore, consider specifying DYNAMIC=NO.

FSSNAME=fss_name

Specifies a unique FSS associated with this FSA. This value must match the value coded for the FSSNAME parameter in the corresponding FSSDEF statement. This parameter is optional. If you omit it, the default is the name of this FSA specified in the JNAME parameter.

JNAME=fsa_name

Specifies the unique 1- to 8-character name of this FSA. This parameter is required.

JUNIT=(devnum,main,msgdest,ON | OFF)

devnum For IP PrintWay, do *not* specify a device address. Use a comma as a position holder.

main Name of the processor on which IP PrintWay is running.

msgdest Destination for messages about the FSA.

ON | OFF Specifies whether the FSA is initially online or offline.

This parameter is required.

MODE=FSS

Specifies that this FSA is managed by an FSS. This parameter is required.

PDEFAULT={NONE | CHARS | FCB | (CHARS,FCB)}

Specifies the JES3 default values that JES3 does not supply to IP PrintWay.

CHARS	Indicates that JES3 will not supply a default font name to IP PrintWay when the WS=U parameter is specified on the DEVICE statement.
FCB	Indicates that JES3 will not supply a default FCB name and page definition name to IP PrintWay when the WS=C parameter is specified on the DEVICE statement.
NONE	Indicates that JES3 will supply a default font name, FCB name, and page definition name to IP PrintWay.

This parameter is optional. The default is NONE.

Tip: Specify PDEFAULT=(CHARS,FCB) and WS=(U,C) if you want the AFP to PCL, AFP to PostScript, and AFP to PDF transforms to use the default font and page definition specified in the printer definition or, if none is specified, in the transform configuration file for jobs submitted to IP PrintWay with a JCL statement. For more information, see “Specifying a default font and page definition for data transforms (basic mode)” on page 304.

WC=(classes)

Specifies the output classes processed by this FSA. List all classes to be selected by this FSA. Separate each class with a comma. If you designate CLASS as a work-selection criterion on the WS parameter, this FSA selects data sets that match the values specified here. This parameter is optional. If you omit it, this FSA selects data sets with any output class.

WS=(work_selection_criteria)

Specifies the work-selection criteria for this FSA. Separate each value with a comma. For the valid values and the default, see *z/OS JES3 Initialization and Tuning Reference*. For an IP PrintWay FSA, consider these values:

C	Causes JES3 to pass the FCB name specified on the OUTPUT JCL statement to the FSA.
CL	Specifies that the FSA selects only those data sets with the same class that are specified in the WC parameter of this statement.
D	Specifies that the FSA selects only those data sets with the same destination as in the DGROUP parameter of this statement. For considerations related to this work-selection criterion, see “Determining work-selection criteria (basic mode)” on page 303.
F	Requests that JES3 pass the forms name specified on a JCL statement to the FSA.
U	Requests that JES3 pass the font name specified on a JCL statement to the FSA.

Notes:

1. If you want IP PrintWay to use the form name specified in the FORMS JCL parameter to select printer definitions in the Printer Inventory,

specify `WS=F` as one of the work-selection criteria. If you do not specify `WS=F`, JES3 does not pass the form name specified in JCL to the IP PrintWay FSA.

2. If you want IP PrintWay to use the FCB specified in the FCB JCL parameter to format output, specify `WS=C` as one of the work-selection criteria. If you do not specify `WS=C`, JES3 does not pass the FCB name specified in JCL to the IP PrintWay FSA.
3. If you want IP PrintWay or any data stream transforms to use the page definition specified in the PAGEDEF (or FCB) JCL parameter, specify `WS=C` as one of the work-selection criteria. If you do not specify `WS=C`, JES3 does not pass the page definition name specified in JCL to the IP PrintWay FSA.
4. If you want IP PrintWay or any data stream transforms to use the font specified in the CHARS (or UCS) JCL parameter, specify `WS=U` as one of the work-selection criteria. If you do not specify `WS=U`, JES3 does not pass the font name specified in JCL to the PrintWay FSA.

Tip: You might want to specify the `BURST=NO` parameter to prevent transmission of the JESNEWS data set.

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.
- FSA definition. IP PrintWay uses values specified in this definition when an IP PrintWay FSA starts.

You can use either the Infoprint Server ISPF panels or the Printer Inventory Definition Utility (PIDU) to create FSS and FSA definitions. For information about how to use the Infoprint Server ISPF panels or PIDU, see *z/OS Infoprint Server Operation and Administration*.

Creating FSS definitions (basic mode)

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.

Table 37 on page 313 shows the fields that you can specify in an FSS definition. The table lists the field names on the ISPF panel, describes each field, and indicates the default value that IP PrintWay uses if you do not create an FSS definition or if you leave a field blank.

Table 37. IP PrintWay FSS definition

ISPF field name	Description	Default
Applid	<p>The application program ID that IP PrintWay uses to establish a VTAM session with printers. This ID must match the name of the APPL statement defined to VTAM.</p> <p>Note: Specify this field only after you have installed Infoprint Coaxial Printer Support for z/OS. Otherwise, IP PrintWay ends abnormally.</p>	No default. This field is required to use the IP PrintWay VTAM protocol.
Description	A description of the FSS definition, for information purposes only.	No default.
Document code page	<p>The default EBCDIC code page used as the document code page when IP PrintWay translates data from EBCDIC to ASCII. You can specify 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 <i>z/OS XL C/C++ Programming Guide</i>.</p> <p>Infoprint Server converts data from (1) the document code page to the printer code page and (2) the code page for the z/OS locale to the document code page. Therefore, if you specify a custom code page, make sure that conversion tables exist to let Infoprint Server convert between code pages. For information about how to create conversion tables, see information about code set conversion utilities in <i>z/OS XL C/C++ Programming Guide</i>.</p>	Code page IBM-1047.
FSS name	<p>The name of the FSS.</p> <p>If the IP PrintWay startup procedure does not contain a JOB JCL statement, specify either:</p> <ul style="list-style-type: none"> • In JES2, the name of the JES FSS initialization statement • In JES3, the name of the JES FSSDEF initialization statement <p>If the IP PrintWay startup procedure contains a JOB JCL statement, specify either:</p> <ol style="list-style-type: none"> 1. The name of the EXEC JCL statement (step name) in the IP PrintWay startup procedure (default name is IEFPROC) 2. If no step name is specified, the name of the PROC JCL statement in the IP PrintWay startup procedure (default is ANFWPROC) 	No default. This field is required.

Table 37. IP PrintWay FSS definition (continued)

ISPF field name	Description	Default
Hiperspace™ blocks	<p>The maximum number of 4 KB blocks that each FSA can use in hiperspace. The number you specify can affect system performance and also can limit the size of the data sets the FSA can process. The maximum amount of hiperspace available is 2 gigabytes.</p> <p>You might need to increase this value to process large data sets. IP PrintWay always writes data sets to hiperspace when it uses the LPR transmission protocol. It also writes data sets to hiperspace when it uses the IPP, direct-sockets, or email transmission protocol and either the Maximum document size or Delete form feed option is selected in the printer definition. IP PrintWay does <i>not</i> write data sets to hiperspace when it transmits data to VTAM-controlled printers.</p>	8,000 (32 MB).
National language	The language IP PrintWay uses for messages. You can select either English or Japanese.	English messages.
Old style translation	<p>An indicator of how IP PrintWay is to convert data from EBCDIC to ASCII when a printer definition does not specify the code pages or translate table to use. That is, the printer definition does not specify (1) either a document or a printer code page, (2) a customized TCP/IP translate table, or (3) a DBCS translate table.</p> <ul style="list-style-type: none"> • If this field is selected, IP PrintWay uses the standard TCP/IP translate table to convert data from EBCDIC to ASCII. • If this field is not selected, IP PrintWay uses code pages to convert data from EBCDIC to ASCII. The EBCDIC code page is the code page specified in the Document code page field of the IP PrintWay FSS definition in the Printer Inventory. The ASCII code pages is IBM-850. <p>This field is ignored when IP PrintWay prints on VTAM-controlled printers or to email destinations. ¹</p>	See the “Description” column.
TCP/IP job name	The name of the TCP/IP startup procedure.	The default TCP/IP name is used.
Trace mode	The type of IP PrintWay tracing for the FSS, and the default tracing mode for all FSAs. Tracing starts when the FSS and FSAs starts. The trace mode specified in an FSA definition overrides this trace mode.	No tracing is done.
Trace prompt	An indicator of whether the operator is prompted each time the FSS starts. If this option is selected, message ANFM020A is sent to the operator.	No prompting is done.

Table 37. IP PrintWay FSS definition (continued)

ISPF field name	Description	Default
Trace table size	The number of 4 KB pages of storage (above the 16 MB line) to allocate for each internal FSA trace table.	32 (128 KB).
1. When printing to VTAM-controlled printers or to email destinations, if the printer definition does not specify code pages or a customized TCP/IP translate table, IP PrintWay does not convert data between code pages.		

This ISPF panel shows a sample IP PrintWay FSS definition:

```

                                IP PrintWay FSS

FSS name. . . PRINTWAY
Description . My IP PrintWay FSS (extend)

      Old-style translation
Hiperspace blocks . . 8000
TCP/IP job name . . . MYTCPIP
Document code page. . IBM-850
Applid. . . . . PWAPPL01
National language . . 1 1. English 2. Japanese
Trace mode. . . . . 2 1. None 2. Internal 3. No printing 4. Full
      Trace prompt
      Trace table size . ___
  
```

Note: If you create an FSS definition after the FSS has started or if you change any information in the FSS definition, you must restart the IP PrintWay FSS so that your changes take effect.

Creating FSA definitions (basic mode)

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 a 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.

This ISPF panel shows a sample IP PrintWay FSA definition:

```

                                IP PrintWay FSA

FSA Name. . . PRT1
Description . Printers in Boulder(extend)

Trace mode . . . 2 1. None 2. Internal 3. No printing 4. Full
  
```

Note: If you create an FSA definition after the FSA is started or if you change the FSA definition, you must restart the IP PrintWay FSA to use the changed values. However, you do not need to restart the FSS and any other IP PrintWay FSA.

Creating the transmission-queue data set (basic mode)

The transmission-queue data set is a required VSAM key-sequenced data set. You must allocate at least one transmission-queue data set before you start IP PrintWay basic mode. IP PrintWay extended mode does not use the transmission-queue data set.

The transmission-queue data set contains a queue for each printer. It also contains an entry for each data set that IP PrintWay is currently processing, and an entry for each data set that IP PrintWay has retained on the JES spool after processing.

IP PrintWay uses information in the transmission-queue data set to manage the transmission of data sets to printers. For example, IP PrintWay transmits data sets to a printer's queue (or port number) in the order in which the IP PrintWay functional subsystem applications (FSAs) acquire the data sets from the JES spool. Also, IP PrintWay FSAs do not attempt to transmit multiple data sets to the same printer's queue (or port number) at the same time.

The operator can use the Infoprint Server ISPF panels to modify and delete entries in the transmission-queue data set. For example, the ISPF panels let you print and delete retained data sets, and move data sets that IP PrintWay is not currently processing to other printers. The ISPF panels let you view the entries in one transmission-queue data set at a time. For information about how to use the Infoprint Server ISPF panels to work with the transmission-queue data set, see *z/OS Infoprint Server Operation and Administration*

Sharing the transmission-queue data set (basic mode)

All IP PrintWay FSAs in a functional subsystem (FSS) share the same transmission-queue data set. Multiple IP PrintWay FSSs, on the same or different z/OS systems, can also share the same transmission-queue data set. As an alternative, you can allocate a separate transmission-queue data set for one or more FSSs.

Advantages of sharing the transmission-queue data set among FSSs are:

- Data sets print on a printer's queue (or port number) in the order in which FSAs within all the FSSs acquire them from the JES spool.
- FSAs in all the FSSs do not attempt to print to a printer's queue (or port number) at the same time. This eliminates delays in printing and failed jobs because of timeout situations.

If the transmission-queue data set is not shared, an IP PrintWay FSA in one FSS might need to wait (for the connection timeout interval) until the printer becomes available. If the printer does not become available in the connection timeout interval, IP PrintWay waits for the retry interval and retries the connection. Waiting can cause a delay in printing because IP PrintWay is not notified when the printer becomes available. In addition, if IP PrintWay completes its retries before the printer becomes available, IP PrintWay fails the print job.

- You can use Infoprint Server ISPF panels to look at the transmission queues for all printers at one time.

Disadvantages of sharing the transmission-queue data set among FSSs are:

- Enqueues on the shared transmission-queue data set can result in printing delays.

- If a very large number of FSSs share the transmission-queue data set, all IP PrintWay printers might stop printing because of the enqueues on the data set.

Guidelines:

- Do not share the same transmission-queue data set among more than 10-15 FSSs. If you have a large workload, a large number of FSAs per FSS, or critical printers that require high performance, do not share the transmission-queue data set among more than 10 FSSs.
- If FSAs within multiple FSSs can transmit output data sets to the same printers, these FSSs should share the same transmission-queue data set, up to the 10-15 FSS limit. This ensures that data sets are printed in order and that the FSSs do not try to send data to the same printer at the same time.

Tip: To determine whether FSSs can transmit data sets to the same printers, look at the JES work-selection criteria for each FSA. If the JES work-selection criteria are the same for FSAs in multiple FSSs, these FSSs can transmit data sets to the same printer. For example, if the JES work-selection criterion for FSAs in two FSSs is class P, both FSSs can acquire the same output data sets from the JES spool and, therefore, can transmit data sets to the same printer.

- Allocate a separate transmission-queue data set for each FSS that transmits output to a separate set of printers. A separate transmission-queue data set provides the best performance. However, if you want the convenience of using the same set of ISPF panels to work with all output data sets, consider sharing the transmission-queue data set among these FSSs as well, up to the 10-15 FSS limit.
- If you allocate more than one transmission-queue data set, put each data set on a separate volume. The volume should not be heavily used.
- Ask your administrator to specify no retention time for most successfully transmitted print jobs and the shortest retention time that is reasonable for failed print jobs. This is because IP PrintWay keeps entries in the transmission-queue data set for retained print jobs. In addition, retaining fewer print jobs for shorter times reduces the possibility of a system abend because of an out-of-storage condition in the IP PrintWay address space. For more information about the out-of-storage condition, see “Determining how many functional subsystems to define (basic mode)” on page 302.

Allocating and initializing the transmission-queue data set (basic mode)

Before you start IP PrintWay basic mode, you must allocate and initialize the transmission-queue data set. The default name for this data set is ANF.QUEUE.

Guidelines:

- Put the transmission-queue data set on a volume that is not heavily used. If you allocate more than one transmission-queue data set, put each data set on a separate volume.
- Periodically use the VSAM REPRO command to reclaim space in the transmission-queue data set. Do this only when all IP PrintWay FSAs that share the transmission-queue data set are stopped.

Tip: A VSAM LISTCAT of the transmission-queue data set might show the number of extents increasing over time. The extents increase 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.

- If you do not periodically use the VSAM REPRO command to reclaim space, IP PrintWay might abnormally terminate (reason code 0205) because it cannot access the transmission-queue data set. If this occurs, delete the current transmission-queue data set and follow the steps to allocate and initialize a new data set. (To find the name of the current transmission-queue data set, select the **PrintWay Queue** option on the Infoprint Server ISPF main panel. The **Queue Data Set name** field contains the name of the current data set.)

Steps for allocating and initializing the transmission-queue data set:

1. Use the IDCAMS program with the specifications shown to allocate the data set. Sample JCL is in SYS1.SAMPLIB(ANFDEAL).

```

DEFINE CLUSTER -
  (NAME(ANF.QUEUE) -
   VOLUMES(volser) -
   INDEXED -
   SPEED -
   SHAREOPTIONS(4 3)) -
  DATA -
  (NAME(ANF.QUEUE.DATA) -
   CYL(2 1) -
   KEYS(20 0) -
   RECORDSIZE(1292 2048) -
   FREESPACE(10,10) -
   CISZ(24576)) -
  INDEX -
  (NAME(ANF.QUEUE.INDEX))

```

Figure 26. IDCAMS parameters to allocate a transmission-queue data set — SYS1.SAMPLIB(ANFDEAL)

Tips:

- a. You can use any data set name. If you change the name, specify the new data set name in the NAME parameter on the DEFINE statement.
 - b. Replace *volser* in the VOLUMES parameter with the volume ID.
 - c. Do not change the values for the RECORDSIZE, SHAREOPTIONS, and KEYS parameters.
 - d. Adjust the value of the FREESPACE parameter based on experience with the data set. The size of the IP PrintWay transmission-queue data set changes dynamically, according to the number of data sets retained after successful or unsuccessful transmission.
-
2. Run the ANFQINIT program to initialize the data set with binary zeroes. SYS1.SAMPLIB(ANFQINIT) contains sample JCL. If you did not use the default name, ANF.QUEUE, for the data set, change the name of the data set in the sample JCL.
-

Tip: For more information about allocating VSAM data sets, see *z/OS DFSMS Access Method Services Commands*.

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.

Creating the 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(ANFMIJCL) 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 suggested.
- Block size (BLKSIZE) of about 10 records per block. 800 is suggested.

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.

Tip: 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. For more information, see “Message exit (ANFUXMSG) — basic mode” on page 290.

You can view messages in the message-log data set by using the Infoprint Server ISPF panels. For information, see *z/OS Infoprint Server Operation and Administration*.

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

This table summarizes the customization tasks related to setting up the startup procedure. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies to your installation.

Task	Condition	See page
Creating a user ID to assign to the IP PrintWay startup procedure (basic mode)	Required	320
Defining the IP PrintWay startup procedure to RACF (basic mode)	Required	321
Editing the ANFWPROC startup procedure (basic mode)	Optional: To change default settings	321

Creating a user ID to assign to the IP PrintWay startup procedure (basic mode)

The user ID assigned to the IP PrintWay startup procedure must be defined to RACF with access to z/OS UNIX.

Steps for creating a RACF user ID with access to z/OS UNIX:

1. Create a user ID with an OMVS segment 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. You can specify any home directory or use the default home directory.

Example: These RACF commands define user AOPSTC with default group AOPOPER and home directory /tmp. 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) PROGRAM('/bin/sh')) HOME(/tmp)
DFLTGRP(AOPOPER) NOPASSWORD
```

2. 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.

Example: 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.MSGFILE ACCESS(UPDATE) ID(AOPSTC)
PERMIT TCPIP.TCPIP.DATA ACCESS(READ) ID(AOPSTC)
```

Defining the IP PrintWay startup procedure to RACF (basic mode)

You must define a profile for the IP PrintWay startup procedure in the RACF STARTED class.

Example: These RACF commands define a profile for the ANFWPROC procedure. This 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
```

Tip: If you want to use the started procedure table (ICHRIN03) instead, see *z/OS Security Server RACF Security Administrator's Guide* for more information.

Editing the ANFWPROC startup procedure (basic mode)

Infoprint Server provides procedure ANFWPROC, which you can use to start an IP PrintWay FSS. You can use ANFWPROC 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.

ANFWPROC is shipped in SYS1.IBM.PROCLIB. However, during install ANFWPROC might have been copied to another data set in the STC PROCLIB concatenation.

Steps for editing the IP PrintWay startup procedure

1. Copy ANFWPROC to a procedure library that is known to JES. For information about procedure libraries, see *z/OS JES2 Initialization and Tuning Guide* and *z/OS JES3 Initialization and Tuning Guide*.
2. Edit the ANFWPROC procedure:
 - If the high-level qualifier for the TCP/IP TCPIP.DATA data set is not TCPIP, change the TCPHLQ parameter on the PROC statement. If the TCPIP.DATA data set is not *hlq*.TCPIP.DATA, change the data set name in the SYSTCPD DD 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.
 - If you use an IP PrintWay exit and the library that contains the exit is not in the standard z/OS search order, add a STEPLIB statement to identify the library.
 - If you did not add the Language Environment runtime libraries (*hlq*.SCEERUN and *hlq*.SCEERUN2) and the C++ runtime library (*hlq*.SCLBDLL) to the system LNKLST, add them to the STEPLIB DD statement.

- If you plan to use the IPP or email protocol and did not install Infoprint Server or z/OS UNIX sendmail files in default locations, specify the directories in environment variables in the STDENV data definition (DD).
 - If you want to customize the email function, specify environment variables in the STDENV data definition (DD).
-
3. Make sure that the name of this procedure is specified in the JES initialization statements for all IP PrintWay FSSs that you want to use this procedure.
-
4. If IP PrintWay is running, stop all IP PrintWay FSAs in the FSS that uses this startup procedure and then restart the FSAs so that your changes take effect. If JES2 does not stop the FSS automatically, you also need to stop the FSS before restarting the FSAs. For information about how to stop and start IP PrintWay, see *z/OS Infoprint Server Operation and Administration*.
-

Related information: For information about how to access the Language Environment runtime libraries and improve the performance of runtime libraries, see *z/OS UNIX System Services Planning*.

ANFWPROC procedure

Figure 27 shows the ANFWPROC procedure. JES initialization statements for an FSS that uses this startup procedure are shown in Figure 24 on page 306 (JES2) and in Figure 25 on page 309 (JES3).

```
//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.
//* This must be modified if this is not the
//* name being used for the inventory.
//*-----
//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 27. IP PrintWay startup procedure — *SYS1.IBM.PROCLIB(ANFWPROC)*

The JCL statements you can specify in the startup procedure are:

```
//proc_name PROC HLQ=ANF, TCPHLQ=TCPIP
```

Names a cataloged procedure. *proc_name* must match the procedure name in a JES initialization statement for this FSS:

- PROC parameter of the JES2 FSS(*fss_name*) statement
- PNAME parameter of the JES3 FSSDEF statement

The HLQ keyword identifies the high-level qualifier for the IP PrintWay data sets. TCPHLQ identifies the high-level qualifier for the TCPIP.DATA data set. Change these values for your installation if necessary.

//label EXEC PGM=ANFFIEP,REGION=4M,TIME=NOLIMIT,PARM='INV=inventory'
Invokes the IP PrintWay program.

PGM=ANFFIEP

The IP PrintWay program entry point.

REGION=4M The amount of storage below the 16 MB line that the FSS requires.

Tips:

- A region size of 4 megabytes is suitable for many installations. However, if many FSAs run in the IP PrintWay FSS, you might need to specify a region size larger than 4M.
- Do not specify 0M. If you specify 0M and available storage becomes exhausted, a system F02 abend can occur. A system F02 abend might cause JES to fail and might require that you restart your z/OS system. If you specify a value other than 0M and available storage becomes exhausted, an abend other than system F02 will occur. JES should not be affected and you should not need to restart your z/OS system.

TIME=NOLIMIT

TIME=NOLIMIT prevents TIMEOUT abends.

PARM='INV=inventory'

The name assigned to the Printer Inventory in the **inventory** attribute in the **aopd.conf** configuration file. Specify AOP1 (the default name) if you did not create an **aopd.conf** configuration file or if the **inventory** attribute is omitted. This name is case-sensitive. This parameter is required.

//ANFQUEUE DD DSN=&HLQ..QUEUE,DISP=SHR

The IP PrintWay transmission-queue data set. This statement is required.

Tip: To make sure that data sets are always transmitted in the same order as IP PrintWay acquires them from the JES spool, specify the same transmission-queue data set for each IP PrintWay FSS.

//SYSTCPD DD DSN=&TCPHLQ..TCPIP.DATA,DISP=SHR

The name of the TCP/IP TCPIP.DATA data set. For a description of how IP PrintWay searches for the TCPIP.DATA data set, see “Customizing the TCPIP.DATA data set” on page 232.

The SYSTCPD DD statement can refer to either an MVS data set or a z/OS UNIX file. If it refers to a z/OS UNIX file, you must specify the LRECL, BLKSIZE, RECFM=FB, and PATHOPTS=ORDONLY parameters. The maximum line length (LRECL) can be 256. For example, if the line length is 80:

```
//SYSTCPD DD PATH='/etc/resolv.conf',PATHOPTS=ORDONLY,  
// LRECL=80,BLKSIZE=80,RECFM=FB
```

//ANFMSG DD DSN=&HLQ..MSGFILE,DISP=SHR

The IP PrintWay message-log data set. This statement is required.

//STDENV DD DSN=datasetname,DISP=SHR

//STDENV DD PATH='UNIXfilename',PATHOPTS=ORDONLY

Specifies an MVS data set or z/OS UNIX file that contains environment

variables. You need to specify this DD statement only if IP PrintWay uses the Internet Printing Protocol (IPP) or email transmission protocol and the default values for the environment variables are not suitable.

These variables apply to the IPP and email protocols:

- LANG
- LIBPATH
- NLSPATH
- PATH

These variables apply to the email protocol:

- AOP_MAIL_DO_NOT_ADD_SUFFIX
- AOP_MAIL_PRESERVE_SUFFIXES
- AOPMAILER
- AOPMAILER_OPTIONS

For a description of these environment variables and their default values, see Appendix C, “Environment variables,” on page 409.

Rules:

- If you specify the environment variables in an MVS data set, it must have these DCB attributes: RECFM=VB and LRECL=255.
- Specify each environment variable in the format *variable=value*. Specify one environment variable per line or record, starting in column 1. For example:

```
AOP_MAIL_PRESERVE_SUFFIXES=.123 .xls .doc
```

Changing Language Environment runtime options (basic mode)

Figure 28 shows the runtime options for the Language Environment environment that PrintWay establishes. If you want to change the Language Environment runtime options, create a CEEUOPT object file as described in *z/OS Language Environment Programming Guide*. Then create an SMPE usermod to link-edit the object file with ANFICONV.

```
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)
```

Figure 28. Language Environment runtime options for IP PrintWay

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 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.

For information about how to configure PSF for printer sharing, see *PSF for z/OS: Customization*.

Chapter 8. Customizing Infoprint Central

This information describes how to customize Infoprint Central, a print management application that users access from the web. Infoprint Central lets help desk operators diagnose and correct problems with print jobs and printers. Print job submitters might want to use Infoprint Central to work with their own print jobs and check printer status. Other authorized users can use Infoprint Central as well.

Infoprint Central lets users display information about and work with these objects:

- Print jobs submitted to z/OS. A *print job* typically consists of one output data set. However, a print job can consist of several output data sets submitted together in the same batch job. In JES2, a print job is called an *output group*.
- Printers that IP PrintWay extended mode controls (called *IP PrintWay printers*).
- Printers that PSF for z/OS controls and AFP Download Plus senders (called *PSF printers*). (AFP Download Plus is a feature of PSF for z/OS.)
- Logical units that NetSpool controls (called *NetSpool LUs*).
- IP PrintWay extended mode job-selection rules.

Infoprint Central lets users display information about, but not work with, these objects:

- Printer definitions
- Printer pool definitions
- Infoprint Server daemons

You should customize Infoprint Central if you run IP PrintWay extended mode or PSF for z/OS because Infoprint Central lets operators do functions that you cannot do using other tools, such as JES commands or SDSF. For example, you can do these functions:

- IP PrintWay extended mode: You can stop and start printers, redirect print jobs to alternate printers, and turn TCP/IP-connected printers online and offline. In addition, you can cancel or hold print jobs that are currently printing, see whether print jobs have been retained on the JES spool, and see all messages in the common message log for both print jobs and printers. Plus, you can see the status of print jobs that are no longer on the JES spool.

Infoprint Central does not let you work with IP PrintWay basic mode printers or print jobs that IP PrintWay basic mode is processing.

- PSF for z/OS: You can turn TCP/IP-connected printers online and offline. In addition, you can see information about print jobs that are no longer on the JES spool, and see all messages in the common message log for print jobs.

For more information about the functions that Infoprint Central provides, see “Infoprint Central” on page 21.

Table 38 on page 328 summarizes the tasks to customize Infoprint Central. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies to your installation.

Table 38. Customization tasks for Infoprint Central

Task	Condition	See page
Identifying software requirements for Infoprint Central	Required	328
Editing the aopd.conf file for Infoprint Central	Required	329
Initializing the search database	Required	331
Customizing the z/OS HTTP Server	Required	332
Setting up security for Infoprint Central	Required	341
Editing the system configuration definition for Infoprint Central	Optional: To specify dynamic attributes	363
Customizing web browsers	Required	365
Logging on to Infoprint Central from the web	Required	366
Customizing PSF for z/OS to use the Printer Inventory	Optional: To work with PSF printers from Infoprint Central	366
Customizing JES3 console standards (CONSTD statement)	Optional: To change JES3 console editing characters	366

Table 39 lists other components of Infoprint Server you might need to customize:

Table 39. Other components to customize for Infoprint Central

Task	Condition	See page
Customizing the Printer Inventory Manager	Required	27
Customizing IP PrintWay extended mode	Optional: To work with IP PrintWay printers from Infoprint Central	199

Related tasks: After you customize Infoprint Central, do these tasks:

- Stop and restart all Infoprint Server daemons so that your changes take effect. For information about how to stop and restart Infoprint Server daemons, see *z/OS Infoprint Server Operation and Administration*.
- Restart the z/OS HTTP Server so that your changes take effect. For information about how to restart the HTTP Server, see *z/OS HTTP Server Planning, Installing, and Using*.
- If you plan to use Infoprint Central to work with PSF printers, restart the PSF printers after you restart the Infoprint Server daemons.

Identifying software requirements for Infoprint Central

This additional IBM software beyond the z/OS base elements is required:

- IBM XML Toolkit for z/OS, C++ edition, V1.10.0 (5655-J51)
- One of these Java products:
 - IBM 31-bit SDK for z/OS, Java Technology Edition, V7 (5655-W43)
 - IBM 64-bit SDK for z/OS, Java Technology Edition, V7 (5655-W44)
 - IBM 31-bit SDK for z/OS, Java Technology Edition, V6.0.1 (5655-R31)
 - IBM 64-bit SDK for z/OS, Java Technology Edition, V6.0.1 (5655-R32)
 - IBM 31-bit SDK for z/OS, Java Technology Edition, V6 (5655-R31)
 - IBM 64-bit SDK for z/OS, Java Technology Edition, V6 (5655-R32)

For information about Java products for z/OS, see <http://www-03.ibm.com/systems/z/os/zos/tools/java/>

- Microsoft Internet Explorer 6.0 (or a later level)

Editing the `aopd.conf` file for Infoprint Central

The `aopd.conf` file contains attributes that Infoprint Server uses when it starts. The default location of this file is `/etc/Printsrv/aopd.conf`.

To start the Infoprint Central daemon, you must specify the following attribute in the `aopd.conf` 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. Enclose all values in braces. By default, only the Printer Inventory Manager and LPD daemons start.

Rule: If you add or remove `ssid`, start Infoprint Server again. You do not need to stop any daemons before restarting Infoprint Server unless *all* these conditions are met:

- You are adding value `ssid`.
- Dynamic configuration is not enabled.
- You did not specify the `log-retention` attribute or you specified `log-retention=0` the last time you started all Infoprint Server daemons.

In this case, you must first stop all daemons. In addition, before you restart Infoprint Server, if `log-retention=0` is specified, change the value to 1 or higher.

Example: In this example, the Printer Inventory Manager, LPD, IP PrintWay extended mode, and Infoprint Central daemons start:

```
start-daemons = {lpd outd ssid}
```

Default: `start-daemons = {lpd}`

If dynamic configuration is not enabled, you can also specify the following attributes in the `aopd.conf` file:

console-name = name

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

Rules:

- The console name must be 2 – 8 alphanumeric or national (#, \$, and @) characters. The first character cannot be numeric. Enclose a name that contains national characters in single or double quotation marks.
- Do not use these reserved names: HC, INSTREAM, INTERNAL, OPERLOG, SYSIOSRS, SYSLOG, UNKNOWN.
- If you specify the Infoprint Central daemon (`aopssid`) in the `start-daemons` attribute, this console name must be unique among all active and inactive console names on all systems in a sysplex. To display the active and inactive console names, enter this MVS command:

```
DISPLAY EMCS,ST=L
```

For information about the DISPLAY EMCS command, see *z/OS MVS System Commands*.

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute while the Infoprint Central daemon (**aopssid**) is running, stop **aopssid** and restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

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

Example: console-name = PRINTCON

log-retention = days

The number of days worth of messages that Infoprint Server retains in the common message log. Also, this attribute specifies the number of days for which Infoprint Central displays information about print jobs that have been removed from the JES spool.

You can specify a value from 0–59. A value of 0 means that Infoprint Server does not write any messages to the common message log and Infoprint Central does not display information about print jobs that have been removed from the JES spool.

Guidelines:

- If you do not start the Infoprint Central daemon, specify a value greater than 0 so that Infoprint Server writes messages to the common message log.
- If you specify a value greater than 0 or start the Infoprint Server daemon, you might need to increase the amount of space available to the file system mounted at the **/var** or **/var/Printsrv** mount point because the common message log and Historical Inventory can contain a large amount of data. See “Calculating DASD space requirements for the **/var/Printsrv** directory” on page 32.
- Because the common message log and Historical Inventory can contain a large amount of data, start with a value of 1 day. Increase the value by 1 day if there is enough space in the file system and if users request to see more messages or historical information.
- If the size of the common message log in the z/OS UNIX file system grows too large, you can decrease its size. For more information, see “Reducing the size of the common message log in the z/OS UNIX file system” on page 111.
- If you use Infoprint Central to view messages, set the TZ and either the LC_TIME or LC_ALL environment variables. These variables affect the date and time displayed in messages. See “Setting environment variables for the Printer Inventory Manager” on page 43.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: If you start the Infoprint Central daemon (**aopssid**), the default is 1 day. Otherwise, the default is 0. If you enable dynamic configuration, the default changes to 1 day regardless of whether you start **aopssid**.

Example: log-retention = 3

Related information: For information about:

- All attributes in the Infoprint Server configuration file, see Appendix B, “Infoprint Server files,” on page 389.
- How to edit the Infoprint Server configuration file, see “Steps for editing the Infoprint Server configuration file (aopd.conf)” on page 42.
- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 68.

Initializing the search database

Infoprint Server maintains a search database so that Infoprint Central can find objects in the Printer Inventory. The search database is created automatically when you start Infoprint Server. However, you must run the z/OS UNIX **sdbu** command to initialize it.

Before you begin: You must be a member of the RACF group for Infoprint Server administrators (AOPADMIN) or have an effective UID of 0.

Steps for initializing the search database:

1. If Infoprint Server is not running, start Infoprint Server daemons. The search database daemon (**aopsdbd**) must be started before you can initialize the search database. For example, enter this MVS command to run the AOPSTART JCL procedure:

```
START AOPSTART
```

2. Initialize the search database:

```
sdbu -z
```

Result: The search database files are initialized in the **/var/Printsrv/searchdb** directory. If you specified a different base directory in the **base-directory** attribute, the search database files are initialized in that base directory instead of in the **/var/Printsrv** base directory.

Steps for recreating the search database: If the search database daemon (**aopsdbd**) fails, delete the search database and run the **sdbu** command to recreate it.

1. Stop all Infoprint Server daemons. For example, enter this MVS command to run the AOPSTOP JCL procedure:

```
START AOPSTOP
```

2. Change to the Infoprint Server base directory (the default base directory is **/var/Printsrv**):

```
cd /var/Printsrv
```

If you specify a different base directory in the **base-directory** attribute in the Infoprint Server configuration file (**aopd.conf**), change to that directory instead.

3. Delete the search database (**searchdb** directory):

```
rm -r searchdb
```

4. Restart Infoprint Server daemons. For example, enter this MVS command to run the AOPSTART JCL procedure:

```
START AOPSTART
```

When you start Infoprint Server, the search database is created if it does not exist.

5. Initialize the search database:

```
sdbu -z
```

Result: The search database files are created and initialized in the **/var/Printsrv/searchdb** directory. If you specified a different base directory in the **base-directory** attribute, the search database files are created and initialized in that base directory instead of in the **/var/Printsrv** base directory.

Customizing the z/OS HTTP Server

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

Guidelines:

- 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 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.

Related information: For complete information about how to customize the HTTP Server, see *z/OS HTTP Server Planning, Installing, and Using*.

This section describes these tasks:

Task	Condition	See page
Editing the z/OS HTTP Server configuration file (httpd.conf)	Required	333
Editing the z/OS HTTP Server environment variables file (httpd.envvars)	Required	335

Editing the z/OS HTTP Server configuration file (httpd.conf)

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 restrict access to Infoprint Central web pages and to tune the HTTP Server.

Adding required directives

The **Service**, **ServerInit**, **Pass**, and **AddType** directives for Infoprint Central are required. Add these directives to the HTTP Server configuration file *in the order shown* in the following examples. After you update this configuration file, restart the HTTP Server so that your changes take effect.

English version: If you run the English version of Infoprint Central, add these directives in the **Service directives** section of the configuration file:

```
# =====
#
#      Service directives
#
# =====
...
# =====
# *** Infoprint Central Pass, ServerInit, and Service directives ***
# =====
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*
```

All versions: For all versions of Infoprint Central, add these directives at the *end* of the **Addtype directives** section of the configuration file:

```
# =====
#
#      AddType directive:
#
# =====
# *** Infoprint Central AddType directives ***
# =====
...
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
```

Japanese version: If you run the Japanese version of Infoprint Central, add these directives in the **Service directives** section of the configuration file:

```
# =====
#
#      Service directives
#
# =====
...
# =====
# *** Infoprint Central Pass, ServerInit, and Service directives ***
# =====
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
Pass /Infoprint/Ja_JP/*.html /usr/lpp/Printsrv/InfoprintCentral/html/Ja_JP/*.html.binary
Pass /Infoprint/help/Ja_JP/*.htm /usr/lpp/Printsrv/InfoprintCentral/help/Ja_JP/*.htm.binary
Pass /Infoprint/help/Ja_JP/* /usr/lpp/Printsrv/InfoprintCentral/help/Ja_JP/*
AddEncoding .binary 8bit 1.0
AddType .html text/html binary 1.0 # Ja_JP html
AddType .htm text/html binary 1.0 # Ja_JP htm
ServerInit /usr/lpp/Printsrv/lib/aopcentral.so:initialize*
Service /Infoprint* /usr/lpp/Printsrv/lib/aopcentral.so:dllMain*

```

Tips:

- 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.
- If the HTTP Server modifies this configuration file in future releases, you might need to customize it again for Infoprint Central.

Rules:

- 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.

Syntax rules:

- Put each directive on a separate line.
- Do not continue a directive on another line.
- Do not include any blanks after the last character on a line.

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.

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.

Rule: Specify the **Protect** directive *before* the **Pass** and **ServerInit** directives for Infoprint Central.

Example: 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* {
  ServerId System_Logon
  UserId %%CLIENT%%
  Authtype Basic
  PasswdFile %%SAF%%
  Mask USER01,user01,USER02,user02
}

```

Tuning the z/OS HTTP Server

The z/OS HTTP Server lets you specify directives to tune the HTTP Server. For more information about directives for tuning, see *z/OS HTTP Server Planning, Installing, and Using*.

For example, you might want to increase the value in the **OutputTimeout** directive to prevent the HTTP Server from timing out before Infoprint Central can display a large number of search results. When the HTTP Server times out, users of Infoprint Central might see no search results, or might see partial search results or a “Page not found” error.

Tip: If you do not want to increase the value in the **OutputTimeout** directive (default is 2 minutes), you can limit the number of search results that Infoprint Central returns. For more information about limiting the search results, see the AOPLIMIT environment variable in “Editing the z/OS HTTP Server environment variables file (httpd.envvars).”

Editing the z/OS HTTP Server environment variables file (httpd.envvars)

The z/OS HTTP Server environment variable file, **httpd.envvars**, contains environment variables that affect Infoprint Central. The default location is **/etc/httpd.envvars**.

Table 40 summarizes the variables you might need to set in this file. You must always set the **required** variables. You must set the **optional** variables only if the default values are not suitable for your installation.

Table 40. Environment variables in httpd.envvars

Environment variable	Condition	Default
AOP_READ_COMMUNITY	Optional	public
AOP_WRITE_COMMUNITY	Optional	public
AOPCENTRAL	Optional	/usr/lpp/Printsrv/InfoprintCentral
AOPCENTRAL_CODEPAGE	Optional	IBM-1047
AOPCONF	Optional	/etc/Printsrv/aopd.conf
AOPLIMIT	Optional	250
AOPLIMIT_DOCS	Optional	250
AOPLIMIT_JOBS	Optional	250
CLASSPATH	Optional	/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	Optional	Infoprint Central uses Java files in these directories, in the order shown: 1. /usr/lpp/java/J7.0 2. /usr/lpp/java/J6.0 3. /usr/lpp/java/J6.0.1
LANG	Optional	C
LC_ALL	Optional	C
LC_TIME	Optional	C
LIBPATH	Required	None

Table 40. Environment variables in `httpd.envvars` (continued)

Environment variable	Condition	Default
NLSPATH	Required	None
TZ	Optional	EST5EDT

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.

Tips:

- 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

Example: `AOP_READ_COMMUNITY=mypass`

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.

Tips:

- 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

Example: `AOP_WRITE_COMMUNITY=mypass`

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: /usr/lpp/Printsrv/InfoprintCentral

Example: AOPCENTRAL=/usr/lpp/mylib/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

Example: AOPCENTRAL_CODEPAGE=IBM-297

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: /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.

If you set a limit for print jobs in the AOPLIMIT_JOBS environment variable, the limit in AOPLIMIT does not apply to print jobs.

If you set a limit for documents in the AOPLIMIT_DOCS environment variable, the limit in AOPLIMIT does not apply to documents.

Tips:

- 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.
- When the HTTP Server times out, users might not see any search results, or might see partial search results or a "Page not found" error.

Default: 250

Example: AOPLIMIT=100

AOPLIMIT_DOCS

The maximum number of documents 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.

For additional information, see the AOPLIMIT environment variable.

Default: The limit specified in the AOPLIMIT environment variable is used. If that environment variable is not specified, the limit is 250.

Example: AOPLIMIT_DOCS=100

AOPLIMIT_JOBS

The maximum number of print jobs 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.

For additional information, see the AOPLIMIT environment variable.

Default: The limit specified in the AOPLIMIT environment variable is used. If that environment variable is not specified, the limit is 250.

Example: AOPLIMIT_JOBS=100

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.

Rule: Separate directories with a colon, and do not include any blanks in the value. Specify the entire value on one line unless you specify the `_CEE_ENVFILE_CONTINUATION` environment variable (see "Tip").

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. Infoprint Central requires Java 7.0, Java 6.0, or Java 6.0.1. This environment variable is optional. However, you must set it if you did not install the Java files in the default Java directory.

Default: Infoprint Central uses Java files in these directories, in the order shown:

1. */usr/lpp/java/J7.0*
2. */usr/lpp/java/J6.0*
3. */usr/lpp/java/J6.0.1*

Example: JAVA_HOME=/usr/lpp/java/J7.0

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. This locale overrides the locale in the LC_TIME variable.

Rule: In the z/OS HTTP environment variables file (**httpd.envvars**), do not specify a value that is associated with a code page other than IBM-1047.

Examples:

aopstart EXEC: LC_ALL=Fr_FR.IBM-297

httpd.envvars file: LC_ALL=Fr_FR.IBM-1047

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)

LIBPATH

The path used to locate dynamic link libraries (DLLs). This environment variable is required. Add these IBM XML Toolkit V1.10 and Infoprint Server libraries before any existing values in LIBPATH:

```
/usr/lpp/ixm/IBM/xml4c-5_7/lib  
/usr/lpp/ixm/IBM/xslt4c-1_11/lib  
/usr/lpp/Printsrv/lib
```

Rules:

- Separate directories with a colon, and do not include any blanks in the value. Specify the entire value on one line unless you specify the _CEE_ENVFILE_CONTINUATION environment variable (see “Tip”).
- Add the XML Toolkit V1.10 libraries before the libraries for any other XML Toolkit versions.

Default: None

Example:

```
LIBPATH=/usr/lpp/ixm/IBM/xml4c-5_7/lib:/usr/lpp/ixm/IBM/xslt4c-1_11/lib:/usr/lpp/Printsrv/lib:other values
```

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 in NLSPATH:

```
/usr/lpp/Printsrv/%L/%N  
/usr/lpp/Printsrv/En_US/%N
```

Rule: Separate directories with a colon, and do not include any blanks in the value. Specify the entire value on one line unless you specify the _CEE_ENVFILE_CONTINUATION environment variable (see “Tip”).

Default: None

Example:

NLSPATH=*existing values*:/usr/lpp/Printsrv/%L/%N:/usr/lpp/Printsrv/En_US/%N

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.

For information about the allowed values, see *z/OS UNIX System Services Command Reference*.

Default: The z/OS HTTP Server default is EST5EDT (Eastern U.S. time zone). The default for the Printer Inventory Manager is GMT0.

Rules:

- 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 or the variables specified in the STDENV data definition (DD) for the AOPSTART JCL procedure.
- Start each environment variable on a separate line.
- Do not continue an environment variable on another line unless you specify the `_CEE_ENVFILE_CONTINUATION` environment variable (see “Tip”).
- Do not include any blanks after the last character on a line unless you specify the `_CEE_ENVFILE_S` runtime environment variable (see “Tip”).
- After you update the HTTP Server environment variables file, restart the HTTP Server so that your changes take effect.

Tip: To make editing the **httpd.envvars** file easier, you can specify the following z/OS XL C/C++ environment variables. For more information about these environment variables, see *z/OS XL C/C++ Programming Guide*.

`_CEE_ENVFILE_COMMENT`

This environment variable defines a comment character so that you can comment out lines. Specify it at the beginning of the **httpd.envvars** file. For example, to define an asterisk as the comment character, specify:

```
_CEE_ENVFILE_COMMENT=*
```

Then include the comment character (*) in column 1 of a line that you want to treat as a comment.

`_CEE_ENVFILE_CONTINUATION`

This environment variable defines a continuation character so that you can continue environment variables on subsequent lines. Specify it at the beginning of the **httpd.envvars** file. For example, to define a backslash as the continuation character, specify:

```
_CEE_ENVFILE_CONTINUATION=\
```

Then include the continuation character (\) at the end of a line to continue the value on a subsequent line. For example, specify:

```
LIBPATH=\  
/usr/lpp/ixm/IBM/xm14c-5_7/lib:\  
/usr/lpp/ixm/IBM/xs1t4c-1_11/lib:\  
/usr/lpp/Printsrv/lib
```

`_CEE_ENVFILE_S`

This runtime environment variable strips trailing white spaces from each name=value line so that blank characters at the ends of lines are ignored.

Specify it as a parameter to the IMWHTTPD program when you start the z/OS HTTP Server. For example, in the z/OS HTTP Server startup procedure (IMWEBSRV), specify:

```
// LE Parm='ENVAR("_CEE_ENVFILE_S=/etc/httpd.envvars")'
```

Setting up security for Infoprint Central

To set up security for Infoprint Central, you can use the Resource Access Control Facility (RACF) or a functionally equivalent product that follows system authorization facility (SAF) protocol. For more information about how to use RACF, see *z/OS Security Server RACF Security Administrator's Guide*.

This section describes these tasks. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies to your installation.

Task	Condition	See page
Authorizing the AOOPER group to RACF profiles	Required	341
Defining Infoprint Central users as z/OS UNIX users	Required	344
Activating the PRINTSRV class	Optional: To define profiles in the PRINTSRV class	344
Authorizing users to read the Printer Inventory	Optional: To let Infoprint Central users view objects in the Printer Inventory	344
Setting up security for printers	Optional: To restrict who can work with printers	345
Setting up security for NetSpool logical units (LUs)	Optional: To restrict who can work with NetSpool LUs	353
Setting up security for IP PrintWay job-selection rules	Optional: To restrict who can work with job-selection rules	354
Setting up security for print jobs	Optional: To restrict who can work with print jobs	356
Setting up security for Infoprint Server daemons	Optional: To restrict who can view the status of daemons	362
Setting up security for printer and printer pool definitions	Optional: To restrict who can view printer definitions	362
Requesting RACF notification and auditing	Optional: To change default RACF notification and auditing	362

Authorizing the AOOPER group to RACF profiles

You must give the user ID that starts Infoprint Server access to these RACF profiles:

- **JES2:** Profiles in the OPERCMDS class that protect JES operator commands
- **JES3:** Profiles in the OPERCMDS class that protect JES operator commands, and profiles in the JESSPOOL class that protect data sets

Table 41 (JES2) and Table 42 (JES3) list the profiles in the OPERCMDS class and the minimum access required. These profiles are optional. However, if you have defined any of these profiles, you must give the user ID that starts Infoprint Server the minimum access. Because the user ID that starts Infoprint Server must be a member of the RACF group for Infoprint Server operators (AOPADMIN), you can give the AOPADMIN group access to the profiles.

Table 41. Profiles in the OPERCMDS class and minimum access required for the AOPOPER group— JES2 systems

Profile name in OPERCMDS class	Minimum access
<i>jesname</i> .BACKSP.DEV	UPDATE
<i>jesname</i> .CANCEL.DEV	UPDATE
<i>jesname</i> .DISPLAY.DEV	READ
<i>jesname</i> .DISPLAY.BATOUT	READ
<i>jesname</i> .DISPLAY.STCOUT	READ
<i>jesname</i> .FORWARD.DEV	UPDATE
<i>jesname</i> .HALT.DEV	UPDATE
<i>jesname</i> .INTERRUPT.DEV	UPDATE
<i>jesname</i> .MODIFY.DEV	UPDATE
<i>jesname</i> .MODIFY.BATOUT	UPDATE
<i>jesname</i> .MODIFY.STCOUT	UPDATE
<i>jesname</i> .REPEAT.DEV	UPDATE
<i>jesname</i> .START.DEV	UPDATE
<i>jesname</i> .STOP.DEV	UPDATE
MVS.MODIFY	UPDATE
MVS.VARY.NET	UPDATE

Table 42. Profiles in the OPERCMDS and JESSPOOL classes and minimum access required for the AOPOPER group— JES3 systems

Profile name in OPERCMDS class	Minimum access
<i>jesname</i> .CALL.WTR	UPDATE
<i>jesname</i> .CANCEL.DEV. <i>dev</i>	UPDATE
<i>jesname</i> .DISPLAY.D	READ
<i>jesname</i> .DISPLAY.U	READ
<i>jesname</i> .MODIFY.U	UPDATE
<i>jesname</i> .RESTART.DEV. <i>dev</i>	UPDATE
<i>jesname</i> .START.DEV. <i>dev</i>	UPDATE
<i>jesname</i> .VARY.DEV	UPDATE
MVS.MODIFY	UPDATE
MVS.VARY.NET	UPDATE
Profile name in JESSPOOL class	Minimum access
<i>nodeid</i> . <i>userid</i> . <i>jobname</i> . <i>jobid</i> . <i>Ddsid</i> . <i>dsname</i>	UPDATE
Note: For <i>jesname</i> , substitute the name of the JES system. For example, substitute JES2.	
Note: For <i>jesname</i> , substitute the name of the JES system. For example, substitute JES3.	

The user who starts Infoprint Server must be a member of the AOPOPER RACF 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.) Therefore, you can give the AOPOPER group access to the profiles in the OPERCMDS and JESSPOOL classes as shown in these examples. If someone with a user ID of 0 who is not a member of the AOPOPER 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.

Guideline: To limit access, use the AOPSTART JCL procedure, instead of the `aopstart` command, to start Infoprint Server because you can associate a user ID with the AOPSTART JCL procedure. Then make that user ID the sole member of the AOPOPER group.

Steps for authorizing the AOPOPER group to JES commands:

1. Give the RACF group for Infoprint Server operators (AOPOPER group) access to profiles that protect all operator commands that Infoprint Central uses.

Examples: If the group name for Infoprint Server operators is AOPOPER, 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(AOPOPER)
PERMIT MVS.MODIFY CLASS(OPERCMDS) ACCESS(UPDATE) ID(AOPOPER)
PERMIT MVS.VARY.NET CLASS(OPERCMDS) ACCESS(UPDATE) ID(AOPOPER)
```

JES3:

```
PERMIT JES3.*.* CLASS(OPERCMDS) ACCESS(UPDATE) ID(AOPOPER)
PERMIT JES3.*.*.* CLASS(OPERCMDS) ACCESS(UPDATE) ID(AOPOPER)
PERMIT MVS.MODIFY CLASS(OPERCMDS) ACCESS(UPDATE) ID(AOPOPER)
PERMIT MVS.VARY.NET CLASS(OPERCMDS) ACCESS(UPDATE) ID(AOPOPER)
PERMIT N1.*.*.*.D*.* CLASS(JESSPOOL) ACCESS(UPDATE) ID(AOPOPER)
```

Tips:

- If you define more specific profiles in the OPERCMDS and JESSPOOL classes, give the AOPOPER group access to those profiles as well.
- For information about the fields in the profile in the JESSPOOL class, see “Protecting print jobs with profiles in the JESSPOOL class” on page 359.

-
2. Refresh the OPERCMDS and JESSPOOL classes.

Examples:

JES2:

```
SETROPTS RACLIST(OPERCMDS) REFRESH
```

JES3:f

```
SETROPTS RACLIST(OPERCMDS JESSPOOL) REFRESH
```

-
3. If you permit the AOPOPER group to profiles while the Infoprint Central daemon (`aopssid`) is started, stop and restart it so that your changes take effect. For example, enter these MVS commands to run the AOPSTOP and AOPSTART JCL procedures:

```
START AOPSTOP,OPTIONS='-d ssid'
START AOPSTART
```

Defining Infoprint Central users as z/OS UNIX users

You must define users of Infoprint Central to RACF as z/OS UNIX users. This section summarizes the required steps. However, for complete information, see *z/OS UNIX System Services Planning*.

Tips:

- 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.

Steps for defining users to RACF as z/OS UNIX users:

1. For each user, establish an OMVS segment and set the user's default group to a group that has an OMVS segment with a group identifier (GID). You can specify any home directory or use the default home directory. Use either the ADDUSER command to define a new user ID or the ALTUSER command to modify an existing user ID.

Example: In this ALTUSER command, for *userid*, specify an existing user ID. For *user-identifier*, specify an integer that is different from other UIDs in your installation. For *group*, specify any group that has an OMVS GID.

```
ALTUSER userid OMVS(UID(user-identifier)) PROGRAM('/bin/sh')
      DFLTGRP(group)
```

Activating the PRINTSRV class

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.

Example: This SETROPTS command activates the PRINTSRV class and copies profiles into virtual storage.

```
SETROPTS CLASSACT(PRINTSRV) RACLIST(PRINTSRV)
```

Authorizing users to read the Printer Inventory

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. For information about how to define the AOP.ADMINISTRATOR profile, see "Setting up security for the Printer Inventory and operator commands" on page 47.

Example: The PERMIT command gives users READ access to profile AOP.ADMINISTRATOR in the PRINTSRV class. The SETROPTS command refreshes the PRINTSRV class:

```
PERMIT AOP.ADMINISTRATOR CLASS(PRINTSRV) ACCESS(READ) ID(userid or groupid)
SETROPTS RACLIST(PRINTSRV) REFRESH
```

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 these classes:

- **PRINTSRV class:** Infoprint Server profiles in the PRINTSRV class restrict who can work with IP PrintWay and PSF printers.
- **OPERCMDMDS class:** Profiles in the OPERCMDMDS class restrict who can work with PSF printers.

If you define profiles in both classes, users must have access to *both* profiles to do actions on PSF printers.

Guidelines:

- **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 OPERCMDMDS class to protect printer actions. Otherwise, any Infoprint Central user can work with any PSF printer.

Define profiles in the OPERCMDMDS class if you want to authorize the same users to do actions on all PSF printers. If you want to authorize one group of users to do actions on one set of printers and another group of users to do actions on another set of printers, define profiles in the PRINTSRV class.

Protecting printers with profiles in the PRINTSRV class

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 Denver, and another profile for all printers in Houston. 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.

Table 43 lists the printer actions users can do in Infoprint Central and the minimum access required to the printer's profile in the PRINTSRV class.

Table 43. Infoprint Central printer actions and the minimum access required to profiles in the PRINTSRV class

To do this action in Infoprint Central	Access to this profile in the PRINTSRV class is required	This minimum access is required
Change forms – PSF printers only	Profile specified in Printer Inventory	UPDATE
Change job selection – PSF printers only	Profile specified in Printer Inventory	UPDATE

Table 43. Infoprint Central printer actions and the minimum access required to profiles in the PRINTSRV class (continued)

To do this action in Infoprint Central	Access to this profile in the PRINTSRV class is required	This minimum access is required
Find and display printers (See Notes [®])	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 only (See Notes)	Profile specified in Printer Inventory	CONTROL
Repeat – PSF printers only	Profile specified in Printer Inventory	CONTROL
Reset (See Notes)	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 (See Notes)	Profile specified in Printer Inventory	CONTROL
Turn online (See Notes)	Profile specified in Printer Inventory	CONTROL
View log	Profile specified in Printer Inventory	READ
View properties	Profile specified in Printer Inventory	READ
<p>Notes:</p> <ol style="list-style-type: none"> To redirect an IP PrintWay printer to an alternate printer, CONTROL access is required to the profiles for both printers. 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. For information about how to set this environment variable, see “Editing the z/OS HTTP Server environment variables file (httpd.envvars)” on page 335. To do 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. For information about how to set this environment variable, see “Editing the z/OS HTTP Server environment variables file (httpd.envvars)” on page 335. The RACF profile for the printer in the PRINTSRV class restricts access to the reset, turn offline, and turn online actions. For information, see “Protecting printers with profiles in the PRINTSRV class” on page 345. These PSF printer actions do not apply to AFP Download Plus senders: Interrupt, pause, repeat, reset, space, stop printer and delete the current print job, turn offline, and turn online. 		

Steps for defining profiles in the PRINTSRV class to protect printers:

1. Define a resource profile to RACF in the PRINTSRV class.

You can define a profile for each printer, or you can define one profile for a group of printers or all printers.

Guidelines:

- Select any name for the profile that RACF allows. However, do not start the name with A0P to avoid conflict with profile names that IBM uses now or in the future. Profile names can be up to 64 characters.
- Give the profile universal READ access if you want all Infoprint Central users to be able to view properties of the printer.
- For information about specifying the NOTIFY and AUDIT parameters on the RDEFINE command, see “Requesting RACF notification and auditing” on page 362.

Example: To define a profile named PRINTERS.DENVER for printers in the Denver location, with universal READ access, enter this RACF command:

```
RDEFINE PRINTSRV (PRINTERS.DENVER) UACC(READ)
```

2. Give users access to the resource profile.

For information about the access that is required to do printer actions in Infoprint Central, see Table 43 on page 345. To let users do all printer actions, give CONTROL access.

Example: To give users CONTROL access to the profile, enter this RACF command:

```
PERMIT PRINTERS.DENVER CLASS(PRINTSRV) ACCESS(CONTROL) ID(userid or groupid)
```

3. Refresh the PRINTSRV class.

Example:

```
SETROPTS RACLIST(PRINTSRV) REFRESH
```

4. IP PrintWay printers: Specify the name of the profile in the printer definition for the printer in the Printer Inventory.

Important: If more than one printer definition exists for a printer, specify the same profile in all printer definitions. Otherwise, Infoprint Central might not check the profile. To find all printer definitions for a printer, on the Select Printer Definitions ISPF panel, specify the host name or IP address of the printer in the **IP address** field.

Tips:

- If the same profile applies to a group of printers, for example to all printers in a particular location, specify the same profile in the printer definitions for all the printers.
- You can use the Infoprint Server ISPF panels or the Printer Inventory Definition Utility (PIDU) to edit the printer definition. If you want to specify the same operator security profile in many IP PrintWay printer definitions, it is easier to use PIDU as shown in the following example. For information about how to use the ISPF panels and PIDU, see *z/OS Infoprint Server Operation and Administration*.
- 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.

- If the profile you specify is not defined to RACF, Infoprint Central proceeds as if the user has the required access to the profile.

Examples:

- This ISPF screen shows how to specify the name of the profile in a printer definition that uses the LPR protocol.

```

                                LPR Protocol

Printer definition name . myprinter_____
Operator security profile _____
. . . PRINTERS.DENVER_____

Printer IP address . myprinter.xyzcorp.denver_____ (extend)
Print queue name . . PASS_____ (extend)

:

```

- These PIDU commands 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.

- a. 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 = \"PRINTERS.DENVER\";"}' > /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**.

- b. Inspect the **/tmp/defs** file to make sure the **modify** commands are acceptable.
- c. Enter this command to update the Printer Inventory:

```
pidu /tmp/defs
```

5. PSF printers: Specify the name of the profile in the printer's FSA definition in the Printer Inventory.

Tips:

- If the same profile applies to a group of printers, for example to all printers in a particular location, specify the same profile in the PSF FSA definitions for all the printers.
- You can use the Infoprint Server ISPF panels or the Printer Inventory Definition Utility (PIDU) to edit the FSA definition. If you want to put the same operator security profile in all PSF FSA definitions, it is easier to use PIDU as shown in the example. For information about how to use the ISPF panels and PIDU, see *z/OS Infoprint Server Operation and Administration*.
- If the profile you specify is not defined to RACF, Infoprint Central proceeds as if the user has the required access to the profile.

Examples:

- This ISPF screen shows how to specify the name of the profile in the PSF FSA definition for the printer.

```

Add                               PSF FSA, Channel
Command ==>

FSA Name. . . PRT00123
Description . . . _____ (extend)
Location. . . DENVER _____ (extend)
Operator security profile
. . . PRINTERS.DENVER _____

Processing Information:
:

```

- These PIDU commands show how to specify the name of the same RACF profile in all PSF FSA definitions for TCP/IP-attached printers that do not already have a profile.

- a. Enter these commands as one command on the z/OS UNIX command line:

```

pidu -qc "list fsa where fsa-type=psf-tcpip and
operator-security-profile=null;" |awk '{print "modify fsa " $1
" operator-security-profile=\"PRINTERS.DENVER\";"}' > /tmp/defs

```

The **list** command lists the names of all PSF FSA definitions for TCP/IP-attached printers with no value in the **operator-security-profile** attribute. These names are piped to the **awk** program, which writes **modify** commands to modify the **operator-security-profile** attribute.

- b. Inspect the **/tmp/defs** file to make sure the **modify** commands are acceptable.
- c. Enter this command to update the Printer Inventory:

```

pidu < /tmp/defs

```

6. PSF printers: Make sure that users are authorized to read the Printer Inventory. For information, see “Authorizing users to read the Printer Inventory” on page 344.

Protecting PSF printers with profiles in the OPERCMDS class

You can define profiles in the OPERCMDS class to restrict who can do actions on PSF printers. Profiles in the OPERCMDS class apply only to PSF printers and not to IP PrintWay printers. Profiles in the OPERCMDS class also restrict access to JES commands.

Note: In Infoprint Central, the term “PSF printer” refers to printers controlled by PSF and to AFP Download Plus senders. (AFP Download Plus is a feature of PSF.)

You can create:

- A broad profile to protect actions on all PSF printers. You control the actions users can do by giving them either READ or UPDATE access to the broad profile. Each type of access lets the user do 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.

Guideline: If you have already defined profiles to protect printers in the OPERCMDS class, these profiles should be suitable for Infoprint Central. However, be sure to give Infoprint Central users and the AOPOPER group the required access to these existing profiles.

Related information: For more information about how profiles in the OPERCMDS class apply to JES2 and JES3 commands, see:

- *z/OS JES2 Initialization and Tuning Guide*
- *z/OS JES3 Initialization and Tuning Guide*

Table 44 and Table 45 on page 351 lists the PSF printer actions users can do in Infoprint Central, the profiles in the OPERCMDS class that restrict access to each action, and the minimum access required to the profiles.

Table 44. Infoprint Central printer actions and minimum access required to profiles in the OPERCMDS class — JES2 systems

To do this action in Infoprint Central	Access to this profile in the OPERCMDS class is required	This minimum access is required
Change forms	<i>jesname</i> .MODIFY.DEV	UPDATE
Change job selection	<i>jesname</i> .MODIFY.DEV	UPDATE
Find and display printers (See Notes)	<i>jesname</i> .DISPLAY.DEV	READ
Interrupt	<i>jesname</i> .INTERRUPT.DEV	UPDATE
Pause	<i>jesname</i> .HALT.DEV	UPDATE
Ping and run traceroute	<i>jesname</i> .DISPLAY.DEV	READ
Repeat	<i>jesname</i> .REPEAT.DEV	UPDATE
Reset (See Notes)	<i>jesname</i> .DISPLAY.DEV	READ
Space – backward	<i>jesname</i> .BACKSP.DEV	UPDATE
Space – forward	<i>jesname</i> .FORWARD.DEV	UPDATE
Start	<i>jesname</i> .START.DEV	UPDATE
Stop printer and delete the current print job	<i>jesname</i> .CANCEL.DEV <i>jesname</i> .STOP.DEV	UPDATE UPDATE
Stop printer after the current print job completes	<i>jesname</i> .STOP.DEV	UPDATE
Turn printer offline (See Notes)	<i>jesname</i> .DISPLAY.DEV	READ
Turn printer online (See Notes)	<i>jesname</i> .DISPLAY.DEV	READ
View properties	<i>jesname</i> .DISPLAY.DEV	READ

Table 44. Infoprint Central printer actions and minimum access required to profiles in the OPERCMDS class — JES2 systems (continued)

To do this action in Infoprint Central	Access to this profile in the OPERCMDS class is required	This minimum access is required
Notes: <ul style="list-style-type: none"> For <i>jesname</i>, substitute the name of the JES system. For example, substitute JES2. 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 must be specified in the AOP_READ_COMMUNITY environment variable. For information about how to set this environment variable, see “Editing the z/OS HTTP Server environment variables file (httpd.envvars)” on page 335. To do 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 is specified in the AOP_WRITE_COMMUNITY environment variable. For information about how to set this environment variable, see “Editing the z/OS HTTP Server environment variables file (httpd.envvars)” on page 335. These PSF printer actions do not apply to AFP Download Plus senders: Interrupt, pause, repeat, reset, space, stop printer and delete the current print job, turn offline, and turn online. 		

Table 45. Infoprint Central printer actions and the minimum access required to profiles in the OPERCMDS class — JES3 systems

To do this action in Infoprint Central	Access to this profile in the OPERCMDS class is required	This minimum access is required
Change forms	<i>jesname</i> .CALL.WTR <i>jesname</i> .RESTART.DEV.dev <i>jesname</i> .START.DEV.dev <i>jesname</i> .VARY.DEV	UPDATE UPDATE UPDATE UPDATE
Change job selection	<i>jesname</i> .CALL.WTR <i>jesname</i> .RESTART.DEV.dev <i>jesname</i> .START.DEV.dev <i>jesname</i> .VARY.DEV	UPDATE UPDATE UPDATE UPDATE
Find and display printers	<i>jesname</i> .DISPLAY.D	READ
Interrupt	<i>jesname</i> .RESTART.DEV.dev	UPDATE
Ping	<i>jesname</i> .DISPLAY.D	READ
Repeat	<i>jesname</i> .RESTART.DEV.dev	UPDATE
Space – backward or forward	<i>jesname</i> .RESTART.DEV.dev	UPDATE
Start	<i>jesname</i> .CALL.WTR <i>jesname</i> .START.DEV.dev <i>jesname</i> .VARY.DEV	UPDATE UPDATE UPDATE UPDATE
Stop printer and delete the current print job	<i>jesname</i> .CANCEL.DEV.dev <i>jesname</i> .RESTART.DEV.dev	UPDATE UPDATE
Stop printer after the current print job completes	<i>jesname</i> .RESTART.DEV.dev	UPDATE
View properties	<i>jesname</i> .DISPLAY.D	READ

Table 45. Infoprint Central printer actions and the minimum access required to profiles in the OPERCMDS class — JES3 systems (continued)

To do this action in Infoprint Central	Access to this profile in the OPERCMDS class is required	This minimum access is required
Notes: <ul style="list-style-type: none"> • For <i>jesname</i>, substitute the name of the JES system. For example, substitute JES3. • For <i>dev</i>, substitute the name of the printer. The printer name is the name of the PSF FSA definition in the Printer Inventory and the printer name defined to JES. • These PSF printer actions do not apply to AFP Download Plus senders: Interrupt, repeat, space, stop printer and delete the current print job. 		

Steps for defining profiles in the OPERCMDS class to protect PSF printers:

1. Define profiles in the OPERCMDS class. You can define a broad profile to protect all printer actions or specific profiles to protect specific actions. For a list of profiles, see Table 44 on page 350 and Table 45 on page 351.

Guideline: For information about specifying the NOTIFY and AUDIT parameters on the RDEFINE command, see “Requesting RACF notification and auditing” on page 362.

Examples:

- JES2: To define a broad profile to protect all actions on all PSF printers, enter this RACF command:

```
RDEFINE OPERCMDS (JES2.*.DEV) UACC(NONE)
```

To define a specific profile for the "stop printer and delete the current print job" action, enter this RACF command:

```
RDEFINE OPERCMDS (JES2.CANCEL.DEV) UACC(NONE)
```

- JES3: To define a broad profile to protect all actions on all PSF printers, enter these RACF commands:

```
RDEFINE OPERCMDS (JES3.*.DEV) UACC(NONE)
```

```
RDEFINE OPERCMDS (JES3.*.DEV.*) UACC(NONE)
```

To define a specific profile for the "stop printer and delete the current print job" action, enter this RACF command:

```
RDEFINE OPERCMDS (JES3.CANCEL.DEV.*) UACC(NONE)
```

-
2. Give users access to the resource profiles.

For information about the minimum access required to the JES resource profiles, see Table 44 on page 350 and Table 45 on page 351.

Examples:

- JES2: To give some users READ access and other users UPDATE access to the broad profiles, enter these RACF commands:

```
PERMIT JES2.*.DEV CLASS(OPERCMDS) ACCESS(READ) ID(userid or groupid)
PERMIT JES2.*.DEV CLASS(OPERCMDS) ACCESS(UPDATE) ID(userid or groupid)
```

To give users UPDATE access to the specific profile, enter this RACF command:

```
PERMIT JES2.CANCEL.DEV CLASS(OPERCMDS) ACCESS(UPDATE) ID(userid or groupid)
```

- JES3: To give some users READ access and other users UPDATE access to the broad profiles, enter these RACF commands:

```
PERMIT JES3.*.DEV CLASS(OPERCMDS) ACCESS(READ) ID(userid or groupid)
PERMIT JES3.*.DEV.* CLASS(OPERCMDS) ACCESS(READ) ID(userid or groupid)
PERMIT JES3.*.DEV CLASS(OPERCMDS) ACCESS(UPDATE) ID(userid or groupid)
PERMIT JES3.*.DEV.* CLASS(OPERCMDS) ACCESS(UPDATE) ID(userid or groupid)
```

To give users UPDATE access to the specific profile, enter this RACF command:

```
PERMIT JES3.CANCEL.DEV.* CLASS(OPERCMD5) ACCESS(UPDATE) ID(userid or groupid)
```

Tip: If you define both broad and specific profiles, give users who have access to the broad profile the appropriate access to the specific profiles as well so they do not lose access.

-
3. Give the group for Infoprint Server operators (AOPOPER group) the highest level of access to the profiles you created. For information, see “Authorizing the AOPOPER group to RACF profiles” on page 341.

-
4. Refresh the OPERCMD5 class.

Example:

```
SETRPTS RACLIST(OPERCMD5) REFRESH
```

-
5. Make sure that users are authorized to read the Printer Inventory. For information, see “Authorizing users to read the Printer Inventory” on page 344.

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.

Guideline: Define the AOP.NETSPOOL profile. Otherwise, any Infoprint Central user can display, start, and stop NetSpool LUs.

Table 46 lists the NetSpool LU actions users can do in Infoprint Central and the access that is required to the AOP.NETSPOOL profile in the PRINTSRV class.

Table 46. Infoprint Central NetSpool LU actions and the minimum access required to the AOP.NETSPOOL profile in the PRINTSRV class

To do this action in Infoprint Central	Access to this profile in the PRINTSRV class is required	This minimum access is required
Find and display LUs	AOP.NETSPOOL	READ
Start	<ul style="list-style-type: none"> • AOP.NETSPOOL¹ • AOP.ADMINISTRATOR 	<ul style="list-style-type: none"> • CONTROL • UPDATE
Stop	<ul style="list-style-type: none"> • AOP.NETSPOOL¹ • AOP.ADMINISTRATOR 	<ul style="list-style-type: none"> • CONTROL • UPDATE
View log	AOP.NETSPOOL	READ
1. If a user has CONTROL access to the AOP.NETSPOOL profile, the AOP.ADMINISTRATOR profile is not checked.		

Steps for defining the AOP.NETSPOOL profile to protect NetSpool LUs:

1. Define the AOP.NETSPOOL resource profile in the PRINTSRV class.

If you define the profile with universal READ access, anyone can display NetSpool LUs, and anyone authorized to update the Printer Inventory can start and stop NetSpool LUs.

Guideline: For information about specifying the NOTIFY and AUDIT parameters on the RDEFINE command, see “Requesting RACF notification and auditing” on page 362.

Example: To define AOP.NETSPOOL in the PRINTSRV class with universal READ access, enter this RACF command:

```
RDEFINE PRINTSRV (AOP.NETSPOOL) UACC(READ)
```

2. Give users access to the AOP.NETSPOOL resource profile.

For information about the minimum access required, see Table 46 on page 353.

Example: To let users display, start, and stop NetSpool LUs, enter this RACF command:

```
PERMIT AOP.NETSPOOL CLASS(PRINTSRV) ACCESS(CONTROL) ID(userid or groupid)
```

3. Refresh the PRINTSRV class.

Example:

```
SETROPTS RACLIST(PRINTSRV) REFRESH
```

4. Give the RACF group for Infoprint Server operators (AOPOPER group) UPDATE access to the MVS.MODIFY and MVS.VARY.NET profiles in the OPERCMDS class. For information, see “Authorizing the AOPOPER group to RACF profiles” on page 341.

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 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.

Guideline: 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.

Table 47 lists the job-selection rule actions users can do in Infoprint Central and the minimum access required to the job-selection rule's profile.

Table 47. Infoprint Central job-selection rule actions and minimum access required to the profile in the PRINTSRV class

To do this action in Infoprint Central	Access to this profile in the PRINTSRV class is required	This minimum access is required
Display	AOP.ADMINISTRATOR	READ
Start	Profile specified in Printer Inventory	CONTROL
Stop	Profile specified in Printer Inventory	CONTROL

Steps for defining profiles to protect job-selection rules:

1. Define a resource profile in the PRINTSRV class.

You can define a separate profile for each job-selection rule, or you can define one profile for several or all job-selection rules. You can use the same profiles that you defined for printers if you want the same groups of users to start and stop job-selection rules.

Guidelines:

- Select any name for the profile that RACF allows. However, do not start the name with AOP to avoid conflict with profile names that IBM uses. Profile names can be up to 64 characters.
- For information about specifying the NOTIFY and AUDIT parameters on the RDEFINE command, see “Requesting RACF notification and auditing” on page 362.

Example: To define a profile named JOBSELECTION.ALL for job-selection rules, enter this RACF command:

```
RDEFINE PRINTSRV (JOBSELECTION.ALL) UACC(NONE)
```

2. Give users CONTROL access to the profile.

To let users start and stop job-selection rules, give CONTROL access to the profile. READ access has no meaning for profiles that control job selection.

Example: To give users CONTROL access to profile JOBSELECTION.ALL, enter this RACF command:

```
PERMIT JOBSELECTION.ALL CLASS(PRINTSRV) ACCESS(CONTROL) ID(userid or groupid)
```

3. Refresh the PRINTSRV class.

Example:

```
SETROPTS RACLIST(PRINTSRV) REFRESH
```

4. Specify the profile name in the job-selection rule in the Printer Inventory.

Specify the name of the profile for the job-selection rule in the **Operator security profile** field on the Job Selection Rule ISPF panel.

Tips:

- You can use the Infoprint Server ISPF panels or the Printer Inventory Definition Utility (PIDU) to edit the job-selection rule. For more information, see *z/OS Infoprint Server Operation and Administration*.
- If the profile you specify is not defined to RACF, Infoprint Central proceeds as if the user has the required access to the profile.

Example: This ISPF panel shows how to specify the name of the profile in the job-selection rule.

```

Job Selection Rule

Rule name . . Class-P
Description . Start all jobs in class P (extend)
Operator security profile
. . . JOBSELECTION.ALL

DEST. . .
CLASS . . P
FORMS . .
Creator .
WRITER. .

DEST IP address . 3 1. Include 2. Exclude 3. Ignore

Record limit . . . . Lower Upper
Page limit (AFP-only). Lower Upper

```

-
5. Make sure that users are authorized to read the Printer Inventory. For information, see “Authorizing users to read the Printer Inventory” on page 344.
-

Setting up security for print jobs

Infoprint Central lets users work with print jobs that are on the JES spool or with historical print jobs, which are no longer on the JES spool. You can restrict access to print jobs with resource profiles in these classes:

- **OPERCMDS class:** Profiles in the OPERCMDS class restrict which actions users can do on print jobs.
- **JESSPOOL class:** Profiles in the JESSPOOL class restrict who can work with specific print jobs.
- **PRINTSRV class:** The AOP.PRIORITY profile in the PRINTSRV class restricts who can change the priority of a print job.

If you define multiple profiles in the OPERCMDS, JESSPOOL, and PRINTSRV classes that apply to a print job, users must have access to the profiles in *all* classes to do an action. If you define specific profiles in the JESSPOOL class that apply to individual data sets, users must have access to *all* the data sets within a print job to do an action on that print job.

Infoprint Central lets job submitters display and do other actions on their own print jobs according to the system that they use to submit print jobs:

- From any z/OS system, or from a non-z/OS system that authenticates z/OS user IDs:

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.

SAP R/3 authenticates z/OS user IDs. Also, Microsoft Windows authenticates z/OS user IDs when the SMB protocol is used to print.

- From non-z/OS systems that do not authenticate z/OS user IDs:

Infoprint Central does not let job submitters display or do any actions on print jobs, unless the profiles that protect the print jobs specify the required universal access.

The Infoprint Port Monitor for Windows does not authenticate z/OS user IDs. UNIX, AIX, and IBM I systems do not authenticate z/OS user IDs.

Protecting actions on print jobs with profiles in the OPERCMDS class

You can define profiles in the OPERCMDS class to restrict who can do actions on print jobs. Profiles in the OPERCMDS class also restrict access to JES commands.

You can create:

- A broad profile to protect all actions on print jobs. You control the actions users can do by giving them either READ or UPDATE access to the broad profile. Each type of access lets the user do a different set of actions.
- Specific profiles to protect specific actions on print jobs.

Guideline: The profiles you have already defined in the OPERCMDS class to protect data sets on the JES spool should be suitable for use with Infoprint Central. If profiles already exist in the OPERCMDS class, give Infoprint Central users and the AOPOPER group the appropriate access to those profiles.

Related information: For more information about how profiles in the OPERCMDS class protect JES2 and JES3 commands, see:

- *z/OS JES2 Initialization and Tuning Guide*
- *z/OS JES3 Initialization and Tuning Guide*

Table 48 and Table 49 on page 358 list the print job actions users can do in Infoprint Central, the profiles in the OPERCMDS class that restrict access to each action, and the minimum access required to the profile.

Table 48. Infoprint Central print job actions and minimum access required to profiles in the OPERCMDS class – JES2

To do this action in Infoprint Central	Access to this profile in the JES2 OPERCMDS class is required	This minimum access is required
Change priority	<i>jesname</i> .MODIFY.BATOUT <i>jesname</i> .MODIFY.STCOUT	UPDATE
Delete	<i>jesname</i> .MODIFY.BATOUT <i>jesname</i> .MODIFY.STCOUT	UPDATE
Edit properties	<i>jesname</i> .MODIFY.BATOUT <i>jesname</i> .MODIFY.STCOUT	UPDATE
Find and display print jobs	<i>jesname</i> .DISPLAY.BATOUT and <i>jesname</i> .DISPLAY.STCOUT	READ
Hold	<i>jesname</i> .MODIFY.BATOUT <i>jesname</i> .MODIFY.STCOUT	UPDATE
Move	<i>jesname</i> .MODIFY.BATOUT <i>jesname</i> .MODIFY.STCOUT	UPDATE
Release	<i>jesname</i> .MODIFY.BATOUT <i>jesname</i> .MODIFY.STCOUT	UPDATE
View log	<i>jesname</i> .DISPLAY.BATOUT <i>jesname</i> .DISPLAY.STCOUT	READ

Table 48. Infoprint Central print job actions and minimum access required to profiles in the OPERCMDS class – JES2 (continued)

To do this action in Infoprint Central	Access to this profile in the JES2 OPERCMDS class is required	This minimum access is required
View properties	<i>jesname</i> .DISPLAY.BATOUT <i>jesname</i> .DISPLAY.STCOUT	READ
Note: For <i>jesname</i> , substitute the name of the JES system. For example, substitute JES2.		

Table 49. Infoprint Central print job actions and the minimum access required to profiles in the OPERCMDS class – JES3

To do this action in Infoprint Central	Access to this profile in the JES3 OPERCMDS class is required	This minimum access is required
Change priority	<i>jesname</i> .MODIFY.U	UPDATE
Delete	<i>jesname</i> .MODIFY.U	UPDATE
Edit properties	<i>jesname</i> .MODIFY.U	UPDATE
Find and display print jobs	<i>jesname</i> .DISPLAY.U	READ
Hold	<i>jesname</i> .MODIFY.U	UPDATE
Move	<i>jesname</i> .MODIFY.U	UPDATE
Release	<i>jesname</i> .MODIFY.U	UPDATE
View log	<i>jesname</i> .DISPLAY.U	READ
View properties	<i>jesname</i> .DISPLAY.U	READ
Note: For <i>jesname</i> , substitute the name of the JES system. For example, substitute JES3.		

Steps for defining profiles in the OPERCMDS class to protect print jobs:

1. Define profiles in the OPERCMDS class.

You can create one broad profile to protect all actions, or you can define more specific profiles to protect specific actions.

For a list of profiles, see Table 48 on page 357 and Table 49.

Guideline: For information about specifying the NOTIFY and AUDIT parameters on the RDEFINE command, see “Requesting RACF notification and auditing” on page 362.

Examples:

- JES2: To define a broad profile to protect all actions, enter these RACF commands:

```
RDEFINE OPERCMDS (JES2.*.BATOUT) UACC(NONE)
RDEFINE OPERCMDS (JES2.*.STCOUT) UACC(NONE)
```

To define specific profiles to protect actions that modify print jobs, enter these RACF commands:

```
RDEFINE OPERCMDS (JES2.MODIFY.BATOUT) UACC(NONE)
RDEFINE OPERCMDS (JES2.MODIFY.STCOUT) UACC(NONE)
```

- JES3: To define a broad profile to protect all actions, enter this RACF command:

```
RDEFINE OPERCMDS (JES3.*.U) UACC(NONE)
```

To define a specific profile to protect actions that modify print jobs, enter this RACF command:

```
RDEFINE OPERCMDS (JES3.MODIFY.U) UACC(NONE)
```

2. Give users access to the resource profiles.

For information about the minimum access required to the JES resource profiles, see Table 48 on page 357 and Table 49 on page 358.

Examples:

- JES2: To give users READ or UPDATE access to the broad profiles, enter these RACF commands:

```
PERMIT JES2.*.BATOUT CLASS(OPERCMD5) ACCESS(READ) ID(userid or groupid)
PERMIT JES2.*.STCOUT CLASS(OPERCMD5) ACCESS(READ) ID(userid or groupid)
```

```
PERMIT JES2.*.BATOUT CLASS(OPERCMD5) ACCESS(UPDATE) ID(userid or groupid)
PERMIT JES2.*.STCOUT CLASS(OPERCMD5) ACCESS(UPDATE) ID(userid or groupid)
```

To give users UPDATE access to the specific profiles, enter these RACF commands:

```
PERMIT JES2.MODIFY.BATOUT CLASS(OPERCMD5) ACCESS(UPDATE) ID(userid or groupid)
PERMIT JES2.MODIFY.STCOUT CLASS(OPERCMD5) ACCESS(UPDATE) ID(userid or groupid)
```

- JES3: To give some users READ access and other users UPDATE access to the broad profiles, enter these RACF commands:

```
PERMIT JES3.*.U CLASS(OPERCMD5) ACCESS(READ) ID(userid or groupid)
PERMIT JES3.*.U CLASS(OPERCMD5) ACCESS(UPDATE) ID(userid or groupid)
```

To give users UPDATE access to the specific profile, enter this RACF command:

```
PERMIT JES3.MODIFY.U CLASS(OPERCMD5) ACCESS(UPDATE) ID(userid or groupid)
```

Tip: If you define both broad and specific profiles, give users who have access to the broad profile the appropriate access to the specific profiles as well so they do not lose access.

3. Give the group for Infoprint Server operators (AOPOPER group) the highest level of access to the profiles you created. For information, see “Authorizing the AOPOPER group to RACF profiles” on page 341.

4. Refresh the OPERCMD5 class.

Example:

```
SETROPTS RACLIST(OPERCMD5) REFRESH
```

Protecting print jobs with profiles in the JESSPOOL class

You can define profiles in the JESSPOOL class to restrict who can work with print jobs. The profiles in the JESSPOOL class also restrict access to JES commands.

You can define:

- A broad profile to protect all data sets.
- Specific profiles to protect specific data sets or data sets submitted with specific z/OS user IDs.

Tips:

- To let Infoprint Central users work with print jobs submitted from remote non-z/OS systems that do not authenticate z/OS user IDs, give Infoprint Central users access to a broad profile in the JESSPOOL class that protects all data sets as shown in the following example.

- The profiles that already exist in the JESSPOOL class should be suitable for use with Infoprint Central. However, be sure to give Infoprint Central users the appropriate access to those profiles.

Related information: For more information about how profiles in the JESSPOOL class protect JES2 and JES3 commands, see:

- *z/OS JES2 Initialization and Tuning Guide*
- *z/OS JES3 Initialization and Tuning Guide*

Table 50 lists the print job actions users can do in Infoprint Central, the profiles in the JESSPOOL class that restrict access to each action, and the minimum access required to the profile.

Table 50. Infoprint Central print job actions and the minimum access required to profiles in the JESSPOOL class

To do this action in Infoprint Central	Access to this profile in the JESSPOOL class is required	This minimum access is required
Change priority	<i>nodeid.userid.jobname.jobid.Ddsid.dsname</i>	ALTER
Delete	<i>nodeid.userid.jobname.jobid.Ddsid.dsname</i>	ALTER
Edit properties	<i>nodeid.userid.jobname.jobid.Ddsid.dsname</i>	ALTER
Find and display print jobs	<i>nodeid.userid.jobname.jobid.Ddsid.dsname</i>	READ
Hold	<i>nodeid.userid.jobname.jobid.Ddsid.dsname</i>	ALTER
Move	<i>nodeid.userid.jobname.jobid.Ddsid.dsname</i>	ALTER
Release	<i>nodeid.userid.jobname.jobid.Ddsid.dsname</i>	ALTER
View log	<i>nodeid.userid.jobname.jobid.Ddsid.dsname</i>	READ
View properties	<i>nodeid.userid.jobname.jobid.Ddsid.dsname</i>	READ

In the table:

nodeid

NJE node ID of the JES2 or JES3 subsystem

userid

z/OS user ID of the job owner. For some output data sets that Print Interface or NetSpool allocates on the JES spool, this is either the user ID of the user who started the Infoprint Server daemons or BPXAS.

jobname

Job name. For some data sets that Print Interface or NetSpool allocates on the JES spool, this is the user ID of the user who started the Infoprint Server daemons or BPXAS.

Ddsid

Data set ID number that identifies the data set prefixed by the required letter D. The D character in the fifth qualifier of the profile name distinguishes the data set's profile from other JESSPOOL profiles.

dsname

Data set name.

Steps for defining profiles in the JESSPOOL class to protect data sets:

1. Define profiles in the JESSPOOL class.

You can define a broad profile to protect all print jobs or more specific profiles to protect specific data sets. For a list of the profiles, see Table 50 on page 360.

Guideline: For information about specifying the NOTIFY and AUDIT parameters on the RDEFINE command, see “Requesting RACF notification and auditing” on page 362.

Example:

- To define a broad profile to protect all print jobs on node N1, enter this RACF command:

```
RDEFINE JESSPOOL (N1.*.*.D*.) UACC(NONE)
```

2. Give users access to the resource profiles.

For the minimum access required to the JES resource profiles, see Table 50 on page 360.

Example: To give some users READ access and other users ALTER access to the broad profile, enter these RACF commands:

```
PERMIT N1.*.*.D*.* CLASS(JESSPOOL) ACCESS(READ) ID(userid or groupid)
PERMIT N1.*.*.D*.* CLASS(JESSPOOL) ACCESS(ALTER) ID(userid or groupid)
```

3. Refresh the JESSPOOL class.

Example:

```
SETROPTS RACLIST(JESSPOOL) REFRESH
```

Protecting the change priority action with a profile in the PRINTSRV class

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.

Steps for defining the AOP.PRIORITY profile:

1. Define the AOP.PRIORITY resource profile in the PRINTSRV class.

Guideline: For information about specifying the NOTIFY and AUDIT parameters on the RDEFINE command, see “Requesting RACF notification and auditing” on page 362.

Example:

```
RDEFINE PRINTSRV (AOP.PRIORITY) UACC(NONE)
```

2. Give users UPDATE access to the AOP.PRIORITY resource profile.

Example:

```
PERMIT AOP.PRIORITY CLASS(PRINTSRV) ACCESS(UPDATE) ID(userid or groupid)
```

3. Refresh the PRINTSRV class.

Example:

```
SETROPTS RACLIST(PRINTSRV) REFRESH
```

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.

Table 51 lists the actions related to daemons that users can do in Infoprint Central and the minimum access required to the AOP.DAEMON profile in the PRINTSRV class.

Table 51. Infoprint Central daemon actions and the minimum access required to the AOP.DAEMON profile in the PRINTSRV class

To do this action in Infoprint Central	Access to this profile in the PRINTSRV class is required	This minimum access is required
Display	AOP.DAEMON	READ

Steps for defining the AOP.DAEMON profile to protect daemons:

1. Define the AOP.DAEMON resource profile in the PRINTSRV class.

Guideline: For information about specifying the NOTIFY and AUDIT parameters on the RDEFINE command, see “Requesting RACF notification and auditing.”

Example: To define AOP.DAEMON with universal NONE access, enter this RACF command:

```
RDEFINE PRINTSRV (AOP.DAEMON) UACC(NONE)
```

2. Give users access to the AOP.DAEMON resource profile.

Example: Give users READ access:

```
PERMIT AOP.DAEMON CLASS(PRINTSRV) ACCESS(READ) ID(userid or groupid)
```

3. Refresh the PRINTSRV class.

Example:

```
SETROPTS RACLIST(PRINTSRV) REFRESH
```

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. For information about how to authorize users to read the Printer Inventory, see “Authorizing users to read the Printer Inventory” on page 344.

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.

Guideline: 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.

Editing the system configuration definition for Infoprint Central

The system configuration definition in the Printer Inventory contains attributes that let you customize Infoprint Server while Infoprint Server is running. If dynamic configuration is enabled, you can specify these attributes in the system configuration definition to customize Infoprint Central:

console-name = *name*

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

Rules:

- The console name must be 2 – 8 alphanumeric or national (#, \$, and @) characters. The first character cannot be numeric. Enclose a name that contains national characters in single or double quotation marks.
- Do not use these reserved names: HC, INSTREAM, INTERNAL, OPERLOG, SYSIOSRS, SYSLOG, UNKNOWN.
- If you specify the Infoprint Central daemon (**aopssid**) in the **start-daemons** attribute, this console name must be unique among all active and inactive console names on all systems in a sysplex. To display the active and inactive console names, enter this MVS command:

```
DISPLAY EMCS,ST=L
```

For information about the DISPLAY EMCS command, see *z/OS MVS System Commands*.

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute while the Infoprint Central daemon (**aopssid**) is running, stop **aopssid** and restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

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

Example: console-name = PRINTCON

log-retention = days

The number of days worth of messages that Infoprint Central displays for print jobs and printers. Also, the number of days for which Infoprint Central displays print jobs that have been removed from the JES spool.

If you do not use the MVS system logger for the common message log, this attribute also determines how many days worth of messages Infoprint Server retains in the z/OS UNIX file system. If you use the system logger for messages, the retention period specified in the system logger log stream definition is used instead.

You can specify a value from 0 – 59. A value of 0 means that Infoprint Central cannot display any messages and cannot display information about print jobs that have been removed from the JES spool. In addition, if you do not use the system logger for messages, Infoprint Server does not write any messages in the z/OS UNIX file system.

Guidelines:

- If you do not use the system logger for messages, specify a value greater than 0 or use the default of 1 day, so that Infoprint Server writes messages in a z/OS UNIX file system.
- If you use the system logger for messages and also use Infoprint Central, the value should be less than or equal to the retention period specified in the log stream definition. If you specify a higher value than the retention period for the log stream, you might not be able to see the messages for all the historical print jobs that Infoprint Central displays.
- If you use the system logger but do not use Infoprint Central, you can specify 0 to save space in the z/OS UNIX file system.
- If you specify a value greater than 0 or use the default of 1 day, you might need to increase the amount of space available to the file system mounted at the **/var** or **/var/Printsrv** mount point because the common message log and Historical Inventory can contain a large amount of data. See “Calculating DASD space requirements for the **/var/Printsrv** directory” on page 32.
- Because the common message log and Historical Inventory can contain a large amount of data, start with a value of 1 day. Increase the value by 1 day if there is enough space in the file system and if users request to see more messages or historical information.
- If the size of the common message log in the z/OS UNIX file system grows too large, you can decrease its size. For more information, see “Reducing the size of the common message log in the z/OS UNIX file system” on page 111.
- If you use Infoprint Central to view messages, set the TZ and either the LC_TIME or LC_ALL environment variables. These variables affect the date and time displayed in messages. See “Setting environment variables for the Printer Inventory Manager” on page 43.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.

- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: log-retention = 1

Example: log-retention = 3

Example: This screen shows a portion of the ISPF panel that you use to edit the system configuration definition for Infoprint Central:

```

                                Infoprint Server: System Configuration

Messages:
  Log retention period (days) . . . . . 1

Infoprint Central:
  Extended MCS console name . . AOP1xxxx
```

Related information: For information about:

- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 68.
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 80.

Customizing web browsers

Infoprint Central supports only the Microsoft Internet Explorer 6.0 (or a later level) web browser.

You must customize the browser's Internet settings:

Required settings:

- Enable JavaScripts.
- Enable style sheets.

Guidelines:

- Request that your browser always retrieve the latest pages from the web so that the Infoprint Central refresh function works correctly. In Microsoft Internet Explorer, change the setting for temporary Internet files to check for newer versions of stored pages on every visit to the page. Do not use the Microsoft default setting, which is to check for newer versions of stored pages automatically.
- Do not block or disable all cookies. Infoprint Central stores cookies on your system to save your search values.
- In Microsoft Internet Explorer, do not disable the meta refresh security option. Infoprint Central uses the meta refresh function to display your search results automatically.

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`

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.

Example: If the HTTP Server receives requests at the default port, you can enter:

`http://hostname/Infoprint/En_US/IPS.html`

or

`https://hostname/Infoprint/En_US/IPS.html`

Tip: 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.

Customizing PSF for z/OS to use the Printer Inventory

If you want to use Infoprint Central to work with AFP printers that PSF for z/OS controls (called *PSF printers*), you must customize PSF for z/OS to use the Printer Inventory. For information, see Chapter 11, "Customizing PSF for z/OS to use the Printer Inventory," on page 379.

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 (").

Example: To specify the quotation mark (") as the escape character and the # symbol as the backspace character, code:

```
EDIT=("",#,)
```

Tips:

- You must code two quotation marks in the first position because the quotation mark is a special character used by the syntax scanner as a control character. Therefore, you must precede the quotation mark with the escape character, which is the quotation mark.
- If you code the EDIT parameter, do *not* omit the first value, for example EDIT=(, #,). If you omit the first value, no escape character is defined.

For information about the CONSTD statement, see *z/OS JES3 Initialization and Tuning Reference*.

Chapter 9. Installing and customizing the Infoprint Server Windows client

This information describes how to customize the Infoprint Server client that runs on Windows systems. The client is called IBM Infoprint Port Monitor for Windows. The Infoprint Port Monitor is software that you install on a Windows workstation. It lets you print files on z/OS printers just like you print files on local Windows printers. You can print files from any Windows application that has a printing function.

You can also use several other print-submission methods from a Windows system that do not use the Infoprint Port Monitor. Table 52 compares the print-submission methods that you can use: Infoprint Port Monitor, Windows Server Message Block (SMB) protocol, Internet Printing Protocol (IPP), and TCP/IP commands that use the LPR to LPD protocol including the **lprafp** command.

Table 52. Comparison of Windows printing methods

Function	Infoprint Port Monitor	SMB	IPP	LPR/LPD
Specify Infoprint Server job attributes	Yes	No	Yes ¹	No
Obtain print job status	Yes ²	Yes ³	Yes	No
Windows systems supported	Windows Server 2003, Windows Server 2008, Windows 7, Windows Vista (Business and Enterprise editions), Windows XP ⁴	Windows Server 2003, Windows XP Professional	Any Windows systems that support IPP	Any Windows systems with an LPR command
Infoprint Port Monitor must be installed on Windows system	Yes ⁵	No	No	No
Infoprint Server daemon that must be started	aoplpd (LPD)	None ⁶	aopippd xp (IPP Server)	aoplpd (LPD)
<ol style="list-style-type: none"> 1. You can specify IPP job attributes instead of Infoprint Server job attributes. However, if the printer is not IPP-enabled Infoprint Server supports only a subset of IPP job attributes. For more information, see "Customizing the IPP workstation client" on page 143. 2. You can see the documents that are waiting to print. If an error occurred, you can view the Windows Event Log. 3. When the Windows job status window is open, z/OS system performance might be degraded. 4. Infoprint Port Monitor V3.1 can run on 32-bit Windows systems and on 64-bit Windows systems based on x64 architecture. However, it cannot run on 64-bit Windows systems that are based on Intel Itanium architecture. 5. If you define the z/OS printer as a shared Windows network printer, you do not need to install the Infoprint Port Monitor on each user's Windows system. However, users can modify Infoprint Server job attributes before printing documents only from the system on which the Infoprint Port Monitor is installed. 6. You must customize and start the z/OS SMB print server. For information, see "Customizing the z/OS SMB server to use Print Interface" on page 138. 				

To use the Infoprint Port Monitor:

1. Make sure you have Windows administrator authority on your Windows system.

2. If you are currently using an earlier version of the Infoprint Port Monitor, do **not** uninstall it.

3. Customize the Printer Inventory Manager and Print Interface, including the Infoprint Server LPD (**aoplpd**).

4. Download and install the Infoprint Port Monitor from one of these locations:
 - z/OS system: Use a file transfer program (such as, **ftp**) to download one of these files in binary:
 - For 32-bit Windows systems: **/usr/lpp/Printsrv/win/En_US/aopwin32.exe**
 - For 64-bit Windows systems: **/usr/lpp/Printsrv/win/En_US/aopwin64.exe**
 - To download from the z/OS system, you need an account on the z/OS system where Infoprint Server is installed.
 - IBM website: Download the Port Monitor at: <http://www-01.ibm.com/support/docview.wss?uid=psd1P1000597>

Tips:

- a. (Optional) You can download the Infoprint Port Monitor to a shared Windows folder.
 - b. For more information about downloading and installing the Infoprint Port Monitor, see *z/OS Infoprint Server User's Guide*.
-
5. Make sure that Infoprint Sever is started and the Print Interface line printer daemon (**aoplpd**) is started.

 6. Define z/OS printers to your Windows system. For instructions, see the Infoprint Port Monitor online help. A shortcut to the help, located in the Windows folder "IBM Infoprint Port Monitor", was put on your desktop during the installation of the Port Monitor.

Windows printers that you added for the earlier version of the Infoprint Port Monitor are still defined. You do not need to change them.

To define a printer, you need this information:

 - The host name or the IP address of the z/OS system where Infoprint Server is running.
 - The port number on the z/OS system where the Infoprint Server line printer daemon (**aoplpd**) is listening. The port number is usually 515. Specify the same number as specified in the **lpd-port-number** attribute.
 - The name of the printer. This is the name of the printer definition in the Printer Inventory. (When you configure the Infoprint port on the Windows system, you select this name from a list.)
 - The name of the printer driver. The printer driver converts documents to a format that the printer understands. For example, to AFP format (for AFP printers) or to PostScript format (for PostScript printers).

If a PCL to AFP or PostScript to AFP transform filter is specified in the printer definition in the Printer Inventory, you can specify a PCL or PostScript printer driver for an AFP printer.

- If you use the AFP Printer Driver, the model number and resolution of the AFP printer.

-
7. (Optional) You can define the z/OS printer as a shared Windows printer in your Windows network. If you do, other Windows users can print on the printer without downloading the Infoprint Port Monitor.

However, when the printer is defined as a shared Windows printer, other Windows users cannot modify the settings of the Infoprint port before printing a document to the shared printer. For example, a Windows user cannot specify separator sheet information or other job attributes before printing a document.

If you define a z/OS printer as a Windows shared printer:

- In the Infoprint Port Configuration window, select **Unattended port**.
 - In the Infoprint Port Monitor Options window, do not select **Prompt for attributes when printing**. This window is displayed only on the Windows system where the Infoprint Port Monitor is installed.
-

Chapter 10. Customizing the SNMP subagent

This information describes how to customize the SNMP subagent component of Infoprint Server. You must customize the SNMP subagent to monitor printer characteristics and printer status for printers controlled by PSF that are either not TCP/IP-attached or lack internal SNMP agents. For example, you can monitor channel-attached printers.

The Infoprint Server SNMP subagent works with PSF, the z/OS Communications Server SNMP agent, and an SNMP manager application, such as Ricoh Network Printer Manager for the Web. For an overview of the SNMP subagent and how it fits into your z/OS system, see “SNMP subagent” on page 22.

Table 53 summarizes the tasks to customize the SNMP subagent. All tasks are required if you want to use the SNMP subagent.

Table 53. Customization tasks for SNMP subagent

Task	Condition	See page
Identifying software requirements	Required	373
Customizing and starting the z/OS Communications Server SNMP agent	Required	374
Editing the <code>aopd.conf</code> file for the SNMP subagent	Required	374
Editing the system configuration definition for the SNMP subagent	Optional: To change default configuration	375
Enabling SNMP reporting in the Printer Inventory	Required	376
Customizing the SNMP manager	Required	377

This table lists other components of Infoprint Server that you must customize:

Task	Condition	See page
Customizing the Printer Inventory Manager	Required	27

Related task: After you customize the SNMP subagent, the operator must do these tasks. For information about how to start the z/OS SNMP agent and the Infoprint Server SNMP subagent, see *z/OS Infoprint Server Operation and Administration*:

- Start the z/OS SNMP agent if it is not already started.
- Restart Infoprint Server to start the SNMP subagent.
- Start each PSF printer at least once after enabling SNMP reporting in the Printer Inventory and before defining PSF printers to an SNMP manager.

Identifying software requirements

This additional software beyond the z/OS base elements is required:

- Ricoh Network Printer Manager (NPM) for the Web, or an SNMP manager with equivalent function. The SNMP manager must be able to communicate with an SNMP agent that supports the general printer MIB (defined in RFC 1759), and the SNMP manager must support multiple printers defined with one IP address (the IP address of the z/OS system).

Customizing and starting the z/OS Communications Server SNMP agent

You must configure and start the z/OS Communications Server (CS) for SNMP agent before you start the Infoprint Server SNMP subagent. For information about how to configure and start the z/OS CS SNMP agent, see *z/OS Communications Server: IP Configuration Reference*.

Some considerations related to configuring and starting the z/OS CS SNMP agent are:

- The community-name function of SNMP can provide a higher-level of security for your installation. However, you should not use the community-name function with the Infoprint Server SNMP subagent because the Infoprint Server SNMP subagent does not let administrators change printer characteristics.
If your installation wants to use the community-name function, you must specify the *same* community name to (1) the z/OS CS SNMP agent, (2) the TCP/IP SNMP subagent, (3) the Infoprint Server SNMP subagent, and (4) the SNMP manager:
 - Specify the community name to the z/OS CS SNMP agent on the **-c** parameter when you start the SNMP agent. The z/OS CS SNMP agent also lets you specify a community name in the SNMPD.CONF or PW.SRC file. However, IBM suggests that you do *not* create an SNMPD.CONF file or a PW.SRC file unless you want to restrict the community name to use by a certain group of IP addresses (of SNMP managers). In this case, create a PW.SRC file, and do not specify the **-c** parameter when you start the z/OS CS SNMP agent.
 - Specify the community name to the TCP/IP SNMP subagent in the COMMUNITY keyword of the SACONFIG statement in the *hlq.PROFILE.TCPIP* data set.
 - Specify the community name to the Infoprint Server SNMP subagent in the **snmp-community** attribute.
 - For information about how to specify the community name, see the online help for NPM Manager for the Web.
- Provide SNMPTRAP.DEST information to the z/OS CS SNMP agent. SNMPTRAP.DEST information defines the IP addresses of the SNMP managers to which the z/OS CS SNMP agent sends traps. When you provide trap information, the z/OS SNMP agent can notify the SNMP manager of printer alert conditions more quickly.
- Do *not* reserve ports 161 and 162 for the z/OS CS SNMP agent in the *hlq.PROFILE.TCPIP* data set. For more information, see “Customizing z/OS Communications Server” on page 117.

Editing the aopd.conf file for the SNMP subagent

The **aopd.conf** file contains attributes that Infoprint Server uses when it starts. The default location of this file is **/etc/Printsrv/aopd.conf**.

To start the SNMP subagent daemon, **aopsnmpd**, you must specify the following attribute in the **aopd.conf** file:

start-daemons = {snmpd}

To start the SNMP subagent, add **snmpd** to the values in this statement. Enclose the values in braces. By default, the Printer Inventory Manager daemons and the LPD start. This statement is required.

Default: start-daemons = {lpd}

Example: In this example, the Printer Inventory Manager daemons and the SNMP subagent start:

```
start-daemons = {snmpd}
```

If dynamic configuration is not enabled, you can also specify the following attribute in the **aopd.conf** file:

snmp-community = name | public

The name assigned to the SNMP community for making SNMP requests to the z/OS system. This name must match both of these:

- The community name provided to the z/OS SNMP agent
- The community name defined to the SNMP manager (Network Printer Manager)

If you do not provide a community name to the z/OS SNMP agent, the z/OS SNMP agent accepts requests from any SNMP manager with the community name of **public**. Specify 1–32 uppercase or lowercase letters, numbers, or special characters. The community name is case-sensitive. Note that the z/OS agent and the SNMP manager might have other restrictions on the length of the community name or the allowed characters.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute while the SNMP subagent (**aopsnmpd**) is running, stop the subagent and restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: snmp-community = public

Related information: For information about:

- All attributes in the Infoprint Server configuration file, see Appendix B, “Infoprint Server files,” on page 389.
- How to edit the Infoprint Server configuration file, see “Steps for editing the Infoprint Server configuration file (aopd.conf)” on page 42.

Editing the system configuration definition for the SNMP subagent

The system configuration definition in the Printer Inventory contains attributes that let you customize Infoprint Server while Infoprint Server is running. If dynamic configuration is enabled, you can specify these attributes in the system configuration definition to customize the SNMP subagent:

snmp-community = *name* | **public**

The name assigned to the SNMP community for making SNMP requests to the z/OS system. This name must match both of these:

- The community name provided to the z/OS SNMP agent
- The community name defined to the SNMP manager (Network Printer Manager)

If you do not provide a community name to the z/OS SNMP agent, the z/OS SNMP agent accepts requests from any SNMP manager with the community name of **public**. Specify 1–32 uppercase or lowercase letters, numbers, or special characters. The community name is case-sensitive. Note that the z/OS agent and the SNMP manager might have other restrictions on the length of the community name or the allowed characters.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute while the SNMP subagent (**aopsnmpd**) is running, stop the subagent and restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: snmp-community = public

Example: This screen shows a portion of the ISPF panel that you use to edit the system configuration definition for the SNMP subagent:

```
Infoprint Server: System Configuration
SNMP Subagent:
Community name . . public
```

Related information: For information about:

- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 68.
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 80.

Enabling SNMP reporting in the Printer Inventory

The administrator can enable SNMP reporting for a PSF-controlled printer in the FSA definition for the PSF printer in the Printer Inventory. By default, SNMP reporting is disabled except if you start the Infoprint Central daemon. To enable SNMP reporting, the administrator must select the **SNMP reporting** field on the ISPF panel used to add or edit the FSA definition.

If a TCP/IP-connected printer contains an internal SNMP agent, IBM suggests that you define the printer directly to the SNMP manager and not enable SNMP reporting in the FSA definition. When you define a printer directly to the SNMP manager, the administrator can also view printer statistics and change some printer characteristics. See the documentation for your TCP/IP-enabled printers to determine if they have internal SNMP agents.

The administrator should enable SNMP reporting in the Printer Inventory and also start the PSF printers before defining the PSF-controlled printers to the SNMP manager.

For information about how to customize PSF to use information in the Printer Inventory and enable SNMP reporting, see *PSF for z/OS: Customization*. For information about how to add or edit an FSA definition, see *z/OS Infoprint Server Operation and Administration*.

Customizing the SNMP manager

To monitor PSF-controlled printers, you must install and use an SNMP manager. Ricoh Network Printer Manager (NPM) for the Web contains the necessary support to display the status of a print system (such as a z/OS system) that contains multiple printers. NPM for the Web requires a Java-enabled browser to monitor printers. NPM for the Web limits the number of printers you can monitor in each print system. For the maximum number of printers that NPM allows, see the NPM online help.

To download the latest version of NPM for the Web visit the Ricoh Web site. Use the NPM README file and online help for information about how to install and use NPM.

Some considerations related to customizing NPM for the Web are:

- To add the z/OS print system to NPM, you need to know the IP address (in dotted-decimal notation) of the z/OS host and the community name that you defined to the z/OS SNMP agent and to the SNMP subagent. If you have not defined a community name, use the default **public**.
- Before you add the z/OS print system to NPM, you must (1) start the z/OS SNMP agent and SNMP subagent, (2) enable SNMP reporting for PSF printers in the Printer Inventory, and (3) start or restart the PSF printers for which you enabled SNMP reporting.

Any PSF printer can be inactive when you define the z/OS print system, provided the printer has been started at least once after SNMP reporting was enabled in the Printer Inventory. If you do not start *any* PSF printers after enabling SNMP reporting, NPM displays a status of NOT SUPPORTED. If this situation occurs, cycle the NPM server.

- When you add the z/OS print system to NPM, NPM automatically discovers each PSF printer for which SNMP reporting is enabled in the Printer Inventory. NPM lists the FSA definition name for the PSF printers it discovers. If a PSF printer is not in the list, start or restart the PSF printer and cycle the NPM server.
- If you enable SNMP reporting for a PSF printer in the Printer Inventory after adding the z/OS print system to NPM, start or restart the PSF printer and cycle the NPM server.
- If you disable SNMP reporting for a PSF printer in the Printer Inventory after adding the z/OS print system to NPM, cycle the NPM server.

Chapter 11. Customizing PSF for z/OS to use the Printer Inventory

You can customize PSF for z/OS and the AFP Download Plus feature to use printer configuration information in the Printer Inventory. In addition, you can customize PSF V4R4 and the AFP Download Plus feature to write messages for print jobs and printers in the Infoprint Server common message log.

When PSF uses printer configuration information in the Printer Inventory:

- If you change information in the Printer Inventory for one PSF functional subsystem (FSA), you need to restart only the FSA that you changed.
- In Infoprint Central, authorized operators can work with PSF printers. For example, operators can start and stop (that is, drain) printers, change JES work-selection criteria, and turn TCP/IP-connected printers online and offline. Operators can also work with AFP Download Plus senders in the same way they work with PSF printers.

Table 54 summarizes the tasks to customize PSF for z/OS. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies to your installation.

Table 54. Customization tasks for PSF for z/OS

Task	Condition	See page
Identifying software requirements	Required	379
Creating PSF FSS and FSA definitions in the Printer Inventory	Required	379
Customizing PSF for z/OS to use the Printer Inventory	Required	380
Customizing PSF for z/OS to write messages to the common message log	Optional: To write PSF messages in the common message log	380

Identifying software requirements

This additional software beyond the z/OS base elements is required:

- To use printer information in the Printer Inventory, PSF for z/OS (5655-M32)
- To write messages in the common message log, PSF for z/OS V4R4 or later

Creating PSF FSS and FSA definitions in the Printer Inventory

PSF requires these definitions in the Printer Inventory:

- **PSF FSS definition:** Create one definition for each functional subsystem (FSS) that uses the Printer Inventory. In this definition, you specify information that applies to the entire FSS.
- **PSF FSA definition:** Create one definition for each functional subsystem application (FSA) in the FSS. In this definition, you specify printer information that you otherwise would specify in the PRINTDEV statement, in the AFPPARMS data set, and in PSF exits.

To create PSF FSS and FSA definitions in the Printer Inventory, you can use one of these methods:

- Infoprint Server ISPF panels
- Infoprint Server Printer Inventory Definition Utility (PIDU)

Tips:

- You can use the Infoprint Server migration program, **aopmig**, to create the required PSF FSS and FSA definitions in the Printer Inventory by copying information from the PSF startup procedure.
- To submit print jobs to PSF printers using JCL, you do not need to create printer definitions for the PSF printers in the Printer Inventory. You only need to create PSF FSA definitions. However, to submit print jobs to PSF printers from the local system using the **lp** command, from remote systems, or from VTAM applications, you must create printer definitions for the PSF printers in addition to the PSF FSA definitions.

Related information: For information about:

- PSF and AFP Download Plus information that you can specify in the Printer Inventory, see:
 - *PSF for z/OS: Customization*
 - *PSF for z/OS: AFP Download Plus*

You can also use the ISPF online help for each field in the PSF FSS and PSF FSA definitions.

- How to use the Infoprint Server ISPF panels and run PIDU, see *z/OS Infoprint Server Operation and Administration*
- How to run the **aopmig** program, see *PSF for z/OS: Customization*

Customizing PSF for z/OS to use the Printer Inventory

After you create FSS and FSA definitions in the Printer Inventory, you must customize PSF to use the Printer Inventory.

Rule: If you plan to use Infoprint Central, restart the PSF FSSs 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.

Related information: For information about how to customize PSF and AFP Download Plus to use the Printer Inventory, see:

- *PSF for z/OS: Customization*
- *PSF for z/OS: AFP Download Plus*

Customizing PSF for z/OS to write messages to the common message log

PSF V4R4 and later can write messages in the Infoprint Server common message log so that you can see all messages from Infoprint Server and PSF in one place. When PSF writes messages in the log:

- Using Infoprint Central, operators can view PSF messages for print jobs that PSF has processed and can view PSF messages for PSF printers.

- Using Infoprint Central, operators can view information about print jobs that PSF has finished processing. This information includes the final status of the print jobs and on which PSF printers the print jobs printed.
- Using the **aoplogu** command, administrators can view PSF messages for print jobs that PSF has processed and can view PSF messages for PSF printers.

To customize PSF to write messages in the Infoprint Server common message log:

1. Use the Infoprint Server ISPF panels to select the **Log messages** field in the PSF FSS definitions in the Printer Inventory.
2. If PSF is running, you must restart the PSF functional subsystem (FSS) before the change to the **Log messages** field can take effect.

Example: This sample Infoprint Server ISPF panel for a PSF FSS definition shows the **Log messages** field selected:

```

                                PSF FSS
FSS name. . . _____
Description . _____(extend)
TCP/IP job name. . . _
NST trace dsname . . _____
Trace table size . . 32
_ Trace prompt
_ Unicode enabled
_/ Log messages

```

Tip: If you use Infoprint Server Printer Inventory Definition Utility (PIDU) to create or edit PSF FSS definitions, specify the **log-messages=yes** attribute.

Related information: For information about how to use the Infoprint Server ISPF panels and PIDU, see *z/OS Infoprint Server Operation and Administration*.

Appendix A. Checklist of Infoprint Server customization tasks

This checklist summarizes the major tasks involved in customizing Infoprint Server and refers you to the section that describes each task. Do steps 1-3 in order. Do subsequent steps in any order unless otherwise noted.

- ___ 1. Install the Infoprint Server element of z/OS. See *z/OS Program Directory*.
- ___ 2. Determine which of these Infoprint Server components you need to customize for your installation:
 - ___ Printer Inventory Manager (All installations must customize the Printer Inventory Manager component.)
 - ___ Print Interface
 - ___ Transform Interface
 - ___ NetSpool
 - ___ IP PrintWay basic mode
 - ___ IP PrintWay extended mode
 - ___ Infoprint Central
 - ___ Infoprint Port Monitor for Windows
 - ___ SNMP subagent

For information about which components you need to customize, see Chapter 2, “Customization roadmap,” on page 25.

In addition, you might want to customize PSF for z/OS to use the Printer Inventory and to write messages in the Infoprint Server common message log.

- ___ 3. **Printer Inventory Manager:** See Chapter 3, “Customizing the Printer Inventory Manager,” on page 27.
 - ___ a. “Identifying software requirements for Printer Inventory Manager” on page 28
 - ___ b. “Enabling the Infoprint Server feature” on page 28
 - ___ c. “Updating the BPXPRMxx member of SYS1.PARMLIB” on page 28
 - ___ d. “Making Language Environment and C++ runtime libraries available” on page 29
 - ___ e. “Creating Infoprint Server directories” on page 30
 - ___ f. “Creating the Infoprint Server configuration file (aopd.conf)” on page 34
 - ___ g. “Sending non-console messages to the hardcopy log” on page 84
 - ___ h. “Setting environment variables for the Printer Inventory Manager” on page 43
 - ___ i. “Setting up security for the Printer Inventory and operator commands” on page 47
 - ___ j. “Setting permissions for directories and executable files -- aopsetup” on page 54
 - ___ k. “Customizing startup and shutdown commands and procedures” on page 56
 - ___ l. “Enabling Infoprint Server ISPF panels” on page 66
 - ___ m. “Enabling dynamic configuration” on page 68

- ___ n. "Editing the system configuration definition for Printer Inventory Manager" on page 76
- ___ o. "Setting up the MVS system logger for the common message log" on page 88
- ___ p. "Backing up and restoring the Printer Inventory" on page 105
- ___ q. "Saving messages in the common message log in the z/OS UNIX file system" on page 110
- ___ r. "Reducing the size of the common message log in the z/OS UNIX file system" on page 111
- ___ s. "Setting Workload Manager goals" on page 112
- ___ 4. **Print Interface:** See Chapter 4, "Customizing Print Interface," on page 115.
 - ___ a. "Identifying software requirements for Print Interface" on page 116
 - ___ b. "Common customization for Print Interface" on page 116
 - ___ 1) "Customizing z/OS Communications Server" on page 117
 - ___ 2) "Editing the **aopd.conf** file for Print Interface" on page 120
 - ___ 3) "Writing a filter program" on page 121
 - ___ 4) "Customizing JES to specify no blank truncation" on page 127
 - ___ 5) "Setting environment variables for Print Interface" on page 127
 - ___ 6) "Editing the system configuration definition for Print Interface" on page 128
 - ___ c. "Customizing the z/OS UNIX printing commands" on page 130
 - ___ d. "Customizing the AOPPRINT JCL procedure" on page 130
 - ___ e. "Customizing the Print Interface LPD" on page 132
 - ___ 1) "Customizing z/OS Communications Server (TCP/IP)" on page 133
 - ___ 2) "Editing the **aopd.conf** file for the LPD" on page 134
 - ___ 3) "Editing the system configuration definition for the LPD" on page 135
 - ___ 4) "Using the LPD compatibility filter" on page 136
 - ___ 5) "Customizing LPR clients" on page 137
 - ___ f. "Customizing the z/OS SMB server to use Print Interface" on page 138
 - ___ 1) "Customizing the z/OS SMB server" on page 139
 - ___ g. "Customizing the Print Interface IPP Server" on page 140
 - ___ 1) "Editing the **aopd.conf** file for the IPP Server" on page 141
 - ___ 2) "Setting environment variables for the IPP Server" on page 142
 - ___ 3) "Editing the system configuration definition for the IPP Server" on page 142
 - ___ 4) "Customizing the IPP workstation client" on page 143
 - ___ h. "Customizing SAP R/3" on page 144

If the SAP R/3 application server runs on the z/OS system:

 - ___ 1) "Defining an SAP user for the callback daemon" on page 146
 - ___ 2) "Creating the SAP callback daemon configuration file (aopsapd.conf)" on page 146

- ___ 3) "Setting environment variables for the SAP OMS" on page 148
 - ___ 4) "Defining a ROMS and LOMS to the SAP R/3 application server" on page 148
 - ___ 5) "Modifying the LOMS to specify job attributes" on page 149
 - If the SAP R/3 application server does not run on the z/OS system:
 - ___ 1) "Customizing the Print Interface LPD" on page 132
 - ___ i. "Customizing the Print Interface subsystem" on page 153
- ___ 5. **Transform Interface:** See Chapter 5, "Customizing Transform Interface," on page 157.
 - ___ a. "Identifying software requirements for Transform Interface" on page 158
 - ___ b. "Editing the **aopd.conf** file for the Transform Manager" on page 158
 - ___ c. "Creating the transform configuration file (aopxfd.conf)" on page 159
 - ___ d. "Creating the remote transform configuration file (aoprxf.conf)" on page 165
 - ___ e. "Setting environment variables for Transform Interface" on page 167
 - ___ f. "Customizing the TCP/IP network firewall" on page 168
 - ___ g. "Setting up security for transform resources" on page 168
 - ___ h. "Creating a code page conversion table" on page 169
- ___ 6. **NetSpool:** See Chapter 6, "Customizing NetSpool," on page 171.
 - ___ a. "Updating the BPXPRMxx member of SYS1.PARMLIB" on page 172
 - ___ b. "Editing the **aopd.conf** file for NetSpool" on page 172
 - ___ c. "Customizing the NetSpool message log" on page 174
 - ___ d. "Customizing the NetSpool startup procedure" on page 174
 - ___ e. "Changing Language Environment runtime options" on page 181
 - ___ f. "Writing NetSpool exit and filter programs" on page 181
 - ___ g. "Setting environment variables for NetSpool" on page 194
 - ___ h. "Editing the system configuration definition for NetSpool" on page 195
- ___ 7. **IP PrintWay basic mode:** See Chapter 7, "Customizing IP PrintWay," on page 199.
 - ___ a. "Comparing IP PrintWay extended mode to basic mode" on page 202
 - ___ b. "Identifying software requirements for IP PrintWay" on page 205
 - ___ c. "Defining printer host names in the DNS" on page 231
 - ___ d. "Customizing z/OS Communications Server" on page 232
 - ___ e. "Creating a VTAM APPL resource definition for IP PrintWay" on page 236
 - ___ f. "Updating the BPXPRMxx member of SYS1.PARMLIB (basic mode)" on page 301
 - ___ g. "APF-authorizing SYS1.IMAGELIB (basic mode)" on page 301
 - ___ h. "Defining the IP PrintWay functional subsystem (basic mode)" on page 302
 - ___ i. "Creating FSS and FSA definitions in the Printer Inventory (basic mode)" on page 312
 - ___ j. "Creating the transmission-queue data set (basic mode)" on page 316

- ___ k. "Creating the IP PrintWay message-log data set (basic mode)" on page 319
- ___ l. "Customizing the IP PrintWay startup procedure (basic mode)" on page 320
- ___ m. "Writing IP PrintWay exit and filter programs" on page 237
- ___ n. "Changing Language Environment runtime options (basic mode)" on page 324
- ___ o. "Customizing PSF for sharing network printers" on page 324
- ___ 8. **IP PrintWay extended mode:** See Chapter 7, "Customizing IP PrintWay," on page 199.
 - ___ a. "Comparing IP PrintWay extended mode to basic mode" on page 202
 - ___ b. "Identifying software requirements for IP PrintWay" on page 205
 - ___ c. "Editing the **aopd.conf** file for IP PrintWay (extended mode)" on page 205
 - ___ d. "Increasing the space available in /var/Printsrv directory (extended mode)" on page 207
 - ___ e. "Updating the BPXPRMxx member of SYS1.PARMLIB (extended mode)" on page 207
 - ___ f. "Creating IP PrintWay job-selection rules (extended mode)" on page 208
 - ___ g. "Authorizing IP PrintWay to select data sets from the JES spool (extended mode)" on page 214
 - ___ h. "Authorizing IP PrintWay to write SMF type 6 records (extended mode)" on page 215
 - ___ i. "Setting environment variables for IP PrintWay (extended mode)" on page 215
 - ___ j. "Editing the system configuration definition for IP PrintWay (extended mode)" on page 223
 - ___ k. "APF-authorizing SYS1.IMAGELIB (extended mode)" on page 230
 - ___ l. "Customizing from AFP transforms for IP PrintWay (extended mode)" on page 230
 - ___ m. "Defining printer host names in the DNS" on page 231
 - ___ n. "Customizing z/OS Communications Server" on page 232
 - ___ o. "Creating a VTAM APPL resource definition for IP PrintWay" on page 236
 - ___ p. "Writing IP PrintWay exit and filter programs" on page 237
 - ___ q. "Customizing PSF for sharing network printers" on page 324
- ___ 9. **Infoprint Central:** See Chapter 8, "Customizing Infoprint Central," on page 327.
 - ___ a. "Identifying software requirements for Infoprint Central" on page 328
 - ___ b. "Editing the **aopd.conf** file for Infoprint Central" on page 329
 - ___ c. "Initializing the search database" on page 331
 - ___ d. "Customizing the z/OS HTTP Server" on page 332
 - ___ e. "Setting up security for Infoprint Central" on page 341
 - ___ f. "Editing the system configuration definition for Infoprint Central" on page 363

- ___ g. "Customizing web browsers" on page 365
- ___ h. "Logging on to Infoprint Central from the web" on page 366
- ___ i. "Customizing PSF for z/OS to use the Printer Inventory" on page 366
- ___ j. "Customizing JES3 console standards (CONSTD statement)" on page 366
- ___ 10. **Infoprint Port Monitor for Windows:** See Chapter 9, "Installing and customizing the Infoprint Server Windows client," on page 369.
- ___ 11. **PSF for z/OS:** See Chapter 11, "Customizing PSF for z/OS to use the Printer Inventory," on page 379.
 - ___ a. "Identifying software requirements" on page 379
 - ___ b. "Creating PSF FSS and FSA definitions in the Printer Inventory" on page 379
 - ___ c. "Customizing PSF for z/OS to use the Printer Inventory" on page 380
 - ___ d. "Customizing PSF for z/OS to write messages to the common message log" on page 380
- ___ 12. **SNMP subagent:** See Chapter 10, "Customizing the SNMP subagent," on page 373.
 - ___ a. "Identifying software requirements" on page 373
 - ___ b. "Customizing and starting the z/OS Communications Server SNMP agent" on page 374
 - ___ c. "Editing the **aopd.conf** file for the SNMP subagent" on page 374
 - ___ d. "Editing the system configuration definition for the SNMP subagent" on page 375
 - ___ e. "Enabling SNMP reporting in the Printer Inventory" on page 376
 - ___ f. Start each PSF printer for which you enabled reporting.
 - ___ g. "Customizing the SNMP manager" on page 377

Appendix B. Infoprint Server files

This appendix describes Infoprint Server files that let you customize Infoprint Server.

aopd.conf -- Infoprint Server configuration file

Purpose

This file contains attributes that customize Infoprint Server.

Format

```
[ #comment ]
[ ascii-codepage = codepage ]*
  base-directory = path
[ console-name = name ]*
[ dynamic-configuration = yes | no ]
[ ebcdic-codepage = codepage ]*
[ inventory = inventory ]
[ ipp-port-number = portnumber ]*
[ jes-name = name ]
[ job-prefix = prefix ]*
[ log-retention = days ]*
[ lpd-port-number = portnumber ]*
[ max-historical-inventory-size = megabytes ]*
[ resolve-printway-printers = yes | no ]
[ smf-recording = yes | no ]*
[ snmp-community = name | public ]*
[ start-daemons = {[ippd] [lpd] [netd] [outd] [snmpd] [ssid] [subd] [xfd] }
[ xcf-group-qualifier = q]
```

Note: Attributes marked with an asterisk are equivalent to attributes that you can specify in the system configuration definition. If dynamic configuration is enabled (**dynamic-configuration=yes**), do not specify these attributes in this file. Infoprint Server ignores them if specified.

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*.

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
- 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. For information about how to create conversion tables, see information about code set conversion utilities in *z/OS XL C/C++ Programming Guide*.

Infoprint Server uses the code page specified in this field in these ways:

- The Printer Inventory Manager displays this code page in the **Printer code page** field in ISPF panels for new IP PrintWay printer definitions that use the LPR, direct sockets, or IPP protocol to transmit data sets to the remote system.

- Print Interface uses this code page as the default source (document) code page when it converts data streams from ASCII to EBCDIC. A document code page specified in the printer definition or job attribute overrides this value.
- Print Interface uses this code page as the target (printer) code page when it converts PostScript documents that were created with a non-ASCII code page.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: `ascii-codepage = IS08859-1`

base-directory = *path*

The directory path in which Infoprint Server creates Printer Inventory files and other files. This attribute is required.

The directory path can contain up to 95 characters. However, if you use Infoprint Server transforms, the maximum length of the directory path might be less, depending on the length of the transform class names you define. If the transform class names can be up to 20 characters, the maximum length of the directory path is 60 characters.

Rules:

- Do not change this attribute while Infoprint Server (including NetSpool and IP PrintWay basic mode) is running. Also, if PSF for z/OS uses the Printer Inventory, do not change this attribute while PSF is running.
- If you use Infoprint Central and change this attribute, restart the z/OS HTTP Server.

Default: None.

Example: To use the suggested name for the base directory:

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

console-name = *name*

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

Rules:

- The console name must be 2 – 8 alphanumeric or national (#, \$, and @) characters. The first character cannot be numeric. Enclose a name that contains national characters in single or double quotation marks.
- Do not use these reserved names: HC, INSTREAM, INTERNAL, OPERLOG, SYSIOSRS, SYSLOG, UNKNOWN.
- If you specify the Infoprint Central daemon (**aopssid**) in the **start-daemons** attribute, this console name must be unique among all active and inactive console names on all systems in a sysplex. To display the active and inactive console names, enter this MVS command:

```
DISPLAY EMCS,ST=L
```

For information about the DISPLAY EMCS command, see *z/OS MVS System Commands*.

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute while the Infoprint Central daemon (**aopssid**) is running, stop **aopssid** and restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

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

Example: console-name = PRINTCON

dynamic-configuration = yes | no

Indicates whether the dynamic configuration function is enabled. If dynamic configuration is enabled, you can change many configuration attributes in the system configuration definition in the Printer Inventory while Infoprint Server is running, without stopping and restarting Infoprint Server.

yes Dynamic configuration is enabled.

no Dynamic configuration is not enabled.

Tip: For more information about dynamic configuration, see “Enabling dynamic configuration” on page 68.

Rules:

- If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.
- If you use Infoprint Central and change this attribute, restart the z/OS HTTP Server.

Default: dynamic-configuration = no

Example: dynamic-configuration = yes

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*.

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
- 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. For information about how to create conversion tables, see information about code set conversion utilities in *z/OS XL C/C++ Programming Guide*.

Infoprint Server uses this code page in these ways:

- The Printer Inventory Manager displays this code page in the **Printer code page** field in ISPF panels for new PSF printer definitions, general printer definitions, and IP PrintWay printer definitions that use the VTAM or email protocol to transmit data sets to the remote system.
- NetSpool uses this code page as the default source (document) code page when it converts data streams from EBCDIC to ASCII. A document code page in the printer definition or job attribute overrides this value.
- IP PrintWay extended mode uses this code page as the default source (document) code page when it converts data streams from EBCDIC to ASCII. A document code page in the printer definition overrides this value.
- Print Interface uses this code page as the source (document) code page when Print Interface cannot determine the user locale for the **lp** command.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server. Also restart the NetSpool started task.

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. It *cannot* be a lowercase letter or numeric. The remaining characters can be uppercase letters or numeric. 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 and AFP Download Plus startup procedures if you want PSF or AFP Download Plus to use information that is specified in FSS and FSA definitions in the Infoprint Server Printer Inventory.

Rules:

- Do not change this attribute while Infoprint Server (including NetSpool and IP PrintWay basic mode) is running. Also, if PSF for z/OS uses the Printer Inventory, do not change this attribute while PSF is running.
- If you use Infoprint Central and change this attribute, restart the z/OS HTTP Server.

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.

Rules:

- Make sure that the port is not used by any other service on the same TCP/IP stack.
- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: ipp-port-number = 631

jes-name = *name*

The 1-4 character name of the JES2 secondary subsystem that Infoprint Server works with. If this attribute is specified, Print Interface and NetSpool create output data sets on the spool of the specified subsystem, and IP PrintWay extended mode selects output data sets for printing from the spool of the specified subsystem. Infoprint Server converts any lowercase letters to uppercase.

Tip: If you want Infoprint Server to work with the JES2 or JES3 primary subsystem, you can either omit this attribute or specify the name of the primary subsystem.

Rules:

- If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.
- If you use Infoprint Central and change this attribute, restart the z/OS HTTP Server.

Default: None. Infoprint Server works with the JES2 or JES3 primary subsystem.

Example: jes-name = JESX

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.

The prefix must be two alphanumeric or national (@, \$, and #) 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 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.
- 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.

Tips:

- z/OS also assigns a unique z/OS job ID to each output data set. The z/OS job ID might be different from the job ID of the output data set on the JES spool. JES operator commands use this z/OS job ID.
- The job ID that Print Interface returns to the job submitter does not contain this prefix.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: job-prefix = PS

Example: job-prefix = 'P#'

log-retention = days

The number of days worth of messages that Infoprint Server retains in the common message log. Also, this attribute specifies the number of days for which Infoprint Central displays information about print jobs that have been removed from the JES spool.

You can specify a value from 0–59. A value of 0 means that Infoprint Server does not write any messages to the common message log and Infoprint Central does not display information about print jobs that have been removed from the JES spool.

Guidelines:

- If you do not start the Infoprint Central daemon, specify a value greater than 0 so that Infoprint Server writes messages to the common message log.
- If you specify a value greater than 0 or start the Infoprint Server daemon, you might need to increase the amount of space available to the file system mounted at the **/var** or **/var/Printsrv** mount point because the common message log and Historical Inventory can contain a large amount of data. See “Calculating DASD space requirements for the **/var/Printsrv** directory” on page 32.
- Because the common message log and Historical Inventory can contain a large amount of data, start with a value of 1 day. Increase the value by 1 day if there is enough space in the file system and if users request to see more messages or historical information.
- If the size of the common message log in the z/OS UNIX file system grows too large, you can decrease its size. For more information, see “Reducing the size of the common message log in the z/OS UNIX file system” on page 111.
- If you use Infoprint Central to view messages, set the TZ and either the LC_TIME or LC_ALL environment variables. These variables affect the date and time displayed in messages. See “Setting environment variables for the Printer Inventory Manager” on page 43.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.

- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: If you start the Infoprint Central daemon (**aopssid**), the default is 1 day. Otherwise, the default is 0. If you enable dynamic configuration, the default changes to 1 day regardless of whether you start **aopssid**.

Example: log-retention = 3

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. 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.

Tips:

- Assign port 515 to the Print Interface LPD so that you can use all of the functions that Infoprint Server provides. These limitations apply if you assign a port other than port 515:
 - 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 Infoprint Port Monitor for Windows, however, is an exception. You can configure the Port Monitor to send print requests to any port.
 - IP PrintWay basic mode cannot transform data with the **resubmit for filtering** option. This is because IP PrintWay resubmits data 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.

Rules:

- Make sure that the port is not used by any other service on the same TCP/IP stack.
- This port number must not be reserved in the *hlq.PROFILE.TCPIP* data set. For more information about the PORT statement, see “Editing the *hlq.PROFILE.TCPIP* data set” on page 117.
- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: lpd-port-number = 515

max-historical-inventory-size = *megabytes*

The maximum size in megabytes (MBs) of the Historical Inventory, **/var/Printsrv/hinv**. The Historical Inventory contains information about print jobs and documents (output data sets) 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. However, the number of days specified in the **log-retention** attribute limits the amount of information that Infoprint Server keeps in the Historical Inventory.

Table 55 shows the number of documents one MB in the Historical Inventory can contain.

Table 55. Number of documents in the Historical Inventory

If print jobs on the JES spool contain:	One MB in the Historical Inventory contains:
1 document in each print job	1350 documents
3 documents in each print job	1750 documents
5 documents in each print job	1850 documents
10 documents in each print job	1950 documents

Guideline: Use this formula to estimate the maximum size of the Historical Inventory:

$$\text{Maximum size of Historical Inventory} = \frac{(\text{Documents processed in one day} \times \text{Days to retain information})}{\text{Documents in one MB}}$$

For example:

- Documents processed in one day = 100,000
- Days to retain information = 3 (**log-retention** = 3)
- Documents in one MB = 1350 documents

The maximum size of the Historical Inventory = $(100,000 \times 3) / 1350 = 223$ MB (rounded up)

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: max-historical-inventory-size = 10

Example: max-historical-inventory-size = 300

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 any of the supported 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 when specifying printer addresses for the same physical printer. For example, if the administrator defines more than one printer definition for the same physical printer, in each printer definition, the administrator must specify exactly the same IP address or host name so that IP PrintWay recognizes that all the printer definitions refer to the same printer.

Tip: If administrators and job submitters always specify dotted-decimal IP addresses for printers, consider specifying **resolve-printway-printers = no**. This limits the number of DNS lookups that IP PrintWay does.

Rules:

- If you specify **resolve-printway-printers = no**, when the operator searches for a printer in Infoprint Central, the operator must enter the printer's dotted-decimal IP address or host name exactly as it is specified in the printer definition or in the DEST=IP: parameter of the OUTPUT JCL statement. Otherwise, Infoprint Central cannot find the printer. For example, if the printer definition contains the printer's IP address, the operator must search for the printer using the IP address instead of the printer's host name.
- If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: resolve-printway-printers = yes

Example: resolve-printway-printers = no

smf-recording = yes | no

Indicates whether IP PrintWay extended mode writes SMF type 6 records for data sets that it sends to a printer or email destination. SMF type 6 records contain accounting information.

yes IP PrintWay extended mode writes SMF type 6 records.

no IP PrintWay extended mode does not write SMF type 6 records.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.
- This attribute does not apply to IP PrintWay basic mode.

Default: smf-recording = yes

Example: smf-recording = no

snmp-community = *name* | **public**

The name assigned to the SNMP community for making SNMP requests to the z/OS system. This name must match both of these:

- The community name provided to the z/OS SNMP agent
- The community name defined to the SNMP manager (Network Printer Manager)

If you do not provide a community name to the z/OS SNMP agent, the z/OS SNMP agent accepts requests from any SNMP manager with the community name of **public**. Specify 1–32 uppercase or lowercase letters, numbers, or special characters. The community name is case-sensitive. Note that the z/OS agent and the SNMP manager might have other restrictions on the length of the community name or the allowed characters.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute while the SNMP subagent (**aopsnmpd**) is running, stop the subagent and restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: snmp-community = public

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:

ippd	The IPP Server daemon, aopippd , starts. It processes print jobs that are submitted by an Internet Printing Protocol (IPP) client.
lpd	The line printer daemon, aoplpd , starts. It processes print jobs that are submitted with commands, such as lpr , and by the Infoprint Port Monitor for Windows.
netd	The NetSpool daemon, aopnetd , starts. It supports printing from VTAM applications such as CICS and IMS.
outd	The IP PrintWay extended mode daemons, aopoutd and aopwsmd , start. They select output data sets from the JES spool and send them to remote printers in a TCP/IP network or to email destinations.
snmpd	The Simple Network Management Protocol (SNMP) subagent daemon, aopsnmpd , starts. It provides status information to the z/OS SNMP agent about printers controlled by PSF.
ssid	The Infoprint Central daemon, aopssid , starts. It communicates between Infoprint Central and JES.
subd	The Print Interface subsystem daemon, aopsubd , starts. It

processes output data sets when the DD JCL statement requests the subsystem on the SUBSYS parameter.

xfd The Transform Manager, **aopxfd**, starts. It manages data stream transforms.

The 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.

Rule: If you add or remove a value from **start-daemons** while Infoprint Server is running, you must start Infoprint Server again. You do not need to stop any daemons before you restart Infoprint Server unless you add value **ssid** (see “Note”).

When you restart Infoprint Server:

- Daemons that were removed from **start-daemons** are stopped.
- Daemons that were added to **start-daemons** are started.
- Daemons that were not removed or added continue to run without interruption.

To run the AOPSTART JCL procedure to start daemons that you added to the **start-daemons** attribute, enter this MVS START command:

```
START AOPSTART
```

Note: Stop all Infoprint Server daemons before you restart Infoprint Server if *all* of these conditions are met:

- You added value **ssid**.
- Dynamic configuration is not enabled.
- You did not specify the **log-retention** attribute or you specified **log-retention=0** the last time you started all Infoprint Server daemons.

Before you restart Infoprint Server, if **log-retention=0** is specified, change the value to 1 or higher.

For example, enter these MVS commands to run the AOPSTART and AOPSTOP JCL procedures:

```
START AOPSTOP  
START AOPSTART
```

Default: `start-daemons = {lpd}`

Examples:

- In this example, all daemons start:
`start-daemons = {ippd lpd netd outd snmpd ssid subd xfd}`
- In this example, only the Printer Inventory Manager daemons start:
`start-daemons = {}`

xcf-group-qualifier = *q*

A one-character XCF group qualifier that Infoprint Server adds to the end of the XCF group name to make it unique in a sysplex. The character must be an alphanumeric or national (\$, #, and @) character. Enclose a national character in single or double quotation marks. A lowercase letter is converted to uppercase.

You must specify a qualifier only if you enable dynamic configuration and multiple instances of Infoprint Server run in a sysplex with the same Printer Inventory name. Keep in mind that in the XCF group name, the Printer Inventory name is always uppercase. Therefore, you need to specify a qualifier if the Printer Inventory names are the same except for case. For example, specify a qualifier if the Printer Inventories are named "AOP1" and "aop1".

As an alternative, you can change the Printer Inventory name so that it is unique for each instance of Infoprint Server running in the sysplex. (You specify the Printer Inventory name in the **inventory** attribute.)

Format of the XCF group name: AOP

pppp[*q*]:

AOP Infoprint Server identifier.

pppp Printer Inventory name. Any lowercase letters are converted to uppercase.

q A one-character XCF group qualifier if specified in **xcf-group-qualifier**. This qualifier is optional.

Rule: If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: None

Example: xcf-group-qualifier = 1

Usage notes

1. For information about dynamic configuration, see "Enabling dynamic configuration" on page 68.
2. The Infoprint Server configuration file resides in the directory specified in the **AOPCONF** environment variable. If **AOPCONF** is not set, the default location is **/etc/Printsrv/aopd.conf**.
3. To create the configuration file, copy sample file **/usr/lpp/Printsrv/samples/aopd.conf** to **/etc/Printsrv/aopd.conf** or to the file location you specify in **AOPCONF**.
4. When you edit this file, follow these syntax rules:
 - Use lowercase characters for the attributes and use uppercase or lowercase characters for the values, with blank characters before or after the equal sign, if desired.
 - If a value contains blank characters or special characters (such as {}-> =,# &), enclose the value in single or double quotation marks.
 - Start comments with the # symbol.
 - Include blank lines, if desired.

aopmsg.conf -- Infoprint Server message configuration file

Purpose

This file contains attributes that customize Infoprint Server message processing.

Note: If dynamic configuration is enabled, Infoprint Server does not use this file.

Format

```
[ #comment ]  
[ hardcopy-messages = all | list | none ]  
[ hardcopy-message-list = {messageID messageID ...} ]
```

hardcopy-messages = **all** | **list** | **none**

Specifies which additional messages Infoprint Server sends to the hardcopy log:

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.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopmsg.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: **hardcopy-messages** = **none**

Example: **hardcopy-messages** = **list**

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.

Tips:

- You do not need to specify messages that Infoprint Server sends to the console, such as messages with the API prefix, because these messages are automatically sent to the hardcopy log.
- If your installation uses the MPFLSTxx Parmlib member for message automation, you might want to specify the same message IDs in MPFLSTxx.
- If your installation uses an automated operations tool, the tool can process these messages.

Rules:

- If you specify this attribute, also specify **hardcopy-messages** = **list**.

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopmsg.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: None.

Example: `hardcopy-message-list={AOP3614I AOP3803E}`

Usage notes

1. For information about dynamic configuration, see “Enabling dynamic configuration” on page 68.
2. The Infoprint Server message configuration file is an optional file. If present, it resides in the directory specified in the AOPMSG_CONF environment variable. If AOPMSG_CONF is not set, the default location is **/etc/Printsrv/aopmsg.conf**.
3. To create the configuration file, copy sample file **/usr/lpp/Printsrv/samples/aopmsg.conf** to **/etc/Printsrv/aopmsg.conf** or to the file location you specify in AOPMSG_CONF.
4. All attributes in this configuration file are optional. If you do not specify an attribute, the default value is used.
5. Follow these syntax rules:
 - Use lowercase characters for the attributes, and use uppercase or lowercase characters for the values.
 - Type blank characters before or after equal signs, if desired.
 - Start comments with the # symbol.
 - Include blank lines, if desired.
6. If you create or edit the configuration file while Infoprint Server is running, stop and restart all Infoprint Server daemons so that your changes take effect. Be sure to wait for all daemons to stop before restarting Infoprint Server.

aoprxf.conf -- Infoprint Server remote transform configuration file

Purpose

This file is required to use transforms that run on another system, such as transforms that InfoPrint Transform Manager for Linux provides. InfoPrint Transform Manager for Linux is a Ricoh product.

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.

Tip: The InfoPrint Transform Manager for Linux user interface lists the IP address and port number of the transform server.

Default: None

Examples:

```
remote-transform-manager-ip-address = linux1.xyz.com  
remote-transform-manager-ip-address = 99.99.123.234
```

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

Example: remote-transform-manager-port-number = 6987

Usage notes

1. This configuration file resides in the directory specified in the AOPRXF_CONF environment variable in the **aopstart** EXEC. If AOPRXF_CONF is not set, the default location is **/etc/Printsrv/aoprxf.conf**.
2. To create the configuration file, copy sample file **/usr/lpp/Printsrv/samples/aoprxf.conf** to **/etc/Printsrv/aoprxf.conf** or to the file location specified in AOPRXF_CONF.
3. Follow these syntax rules:
 - Use lowercase characters for the attributes. Use uppercase or lowercase characters for the values.
 - Start comments with the # symbol.
 - Include blank lines, if desired.
 - Type blank characters before or after equal signs, if desired.
4. If you create or edit the configuration file while Infoprint Server is running Infoprint Server picks up the changes automatically. You do not need to restart Infoprint Server.

aopxfd.conf -- Infoprint Server transform configuration file

Purpose

The Infoprint Server Transform Manager requires this configuration file. The Transform Manager manages transforms, which convert data streams from one format to another. Create an entry for each transform. If you define transform classes, create a separate entry for each transform class.

Format

```
# comment
transform transformname[_transformclass]
  start-command = "daemon [ option ]..."
  [ environment = {name -> value [ name -> value ]...} ]
  [ maximum-active = number ]
  [ maximum-idle-time = seconds ]
  [ minimum-active = number ]
;
```

transform *transformname*[_*transformclass*]

This attribute indicates the beginning of a transform entry. This attribute is required and must be the first attribute in the entry.

transformname

The name of the transform, for example, **pcl2afp** or **ps2afp**.

transformclass

The name of an optional transform class. Specify from 1 to 63 characters, including letters, numbers, or special characters. The transform name is case-sensitive. To use the transform class, you must specify the transform class when you invoke the transform.

Default: No transform class. This transform entry is used when no transform class option (**-c**) is specified either in the printer definition or by the job submitter.

Example: transform **pcl2afp_letter_300**

To specify the **letter_300** transform class on the **pcl2afp** command, type:

```
pcl2afp -c letter_300
```

start-command = "*daemon* [*option*]... "

The name of the transform daemon and options. Enclose the value in single or double quotation marks if you specify an option. This attribute is required.

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 in the **aopstart** EXEC, specify the full directory path name of the daemon.

option

One or more options that the transform supports. For information about the transform options you can specify for each transform, see the documentation for the transform product.

environment = {*name* -> *value* [*name* -> *value*]... }

Environment variables that define the transform environment. Enclose the entire set of environment variables in braces.

For information about the environment variables you can specify for each transform, see the documentation for the transform product.

These 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 runtime environment.

Tip: 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. You can specify a job name of 1 to 8 alphanumeric characters. Incorrect job names are ignored. If you specify a different job name, you might need to list it in the Transaction Name Group (TNG) you defined for the z/OS Workload Manager (see “Setting Workload Manager goals” on page 112). For more information about the `_BPX_JOBNAME` variable, see *z/OS UNIX System Services Planning*.

maximum-active = number

The maximum number of transform daemons that the Infoprint Server Transform Manager activates concurrently. Specify a number greater than 0 and greater than or equal to the number specified in **minimum-active**. When this number is reached, the Infoprint Server Transform Manager does not start a new transform daemon to do the transform, and a print job waits until a transform daemon is available. This attribute is optional.

Guidelines:

- A value of 2 or 3 is suitable if you transform small print jobs and do not transform many print jobs at the same time. However, if you might need to transform large print jobs at the same time, 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.
- If you use the Print Interface subsystem in your installation, do not specify a low 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.

Default: No maximum number. Transform daemons are started when needed.

maximum-idle-time = seconds

The number of seconds before the Infoprint Server Transform Manager shuts down an idle transform daemon and system resources are freed. Specify a number 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.

Default: An idle transform daemon is not shut down.

minimum-active = number

The number of transform daemons that the Infoprint Server Transform

Manager activates concurrently. Also the minimum number of transform daemons that the Infoprint Server Transform Manager keeps active (that is, not shut down) even when the maximum-idle-time expires for an idle transform daemon. Specify a number less than or equal to the number specified in the **maximum-active** attribute. This attribute is optional.

Tip: Do not specify a very high number because active transform daemons use some system resources. However, starting transform daemons uses additional CPU resources, so you should keep some transform daemons active even when they are idle.

Default: minimum-active = 0

Example

```
transform pcl2afp
  start-command = "pcl2afpd"
  environment = { _BPX_JOBNAME -> PCL2AFPD }
  maximum-active = 2
  maximum-idle-time = 300 # 5 minutes
  minimum-active = 1
;
```

Usage notes

1. This configuration file resides in the directory specified in the AOPXFD_CONF environment variable. If this environment variable is not set, the default location is **/etc/Printsrv/aopxfd.conf**.
2. To create this configuration file, copy sample file **/usr/lpp/Printsrv/samples/aopxfd.conf** to **/etc/Printsrv/aopxfd.conf** or to the file location you specify in the AOPXFD_CONF environment variable.
3. After you edit the configuration file, stop and restart the Transform Manager to pick up the changes.
4. Follow these syntax rules:
 - Start comments with the # symbol.
 - End the last attribute in a transform entry with a semicolon.
 - Code blank characters before and after equal signs if desired.
 - Use lowercase characters for the attribute names, and use uppercase or lowercase characters for the attribute values.
 - Environment variable names are case-sensitive. Typically, environment variable names are all uppercase.
 - If a value contains blank characters or special characters (such as {}-> =, # &), enclose the value in single or double quotation marks.
 - To continue a line that contains a character string enclosed in quotation marks, end the line with a back slash and do not type any characters, even blank characters, after the back slash. For example:

```
environment = {AOP_FONTLIB -> "sys1.font300 \
  sys1.sfntlib inst.font300"}
```

aopsapd.conf -- Infoprint Server SAP callback daemon configuration file

Purpose

This file is required when you use the Infoprint Server SAP Output Management System (OMS).

Format

```
# comment
system system-name
    initial-target = instance-name
    client = clientname
    userid = userid
    password = password
    language = language
;

system system-name
    The name of the SAP R/3 system that contains an LOMS for which you
    want callback notification. The SAP R/3 system can run on any z/OS or
    non-z/OS system. This attribute must be the first attribute in the entry.

initial-target = instance-name
    The name of an SAP R/3 server instance on the SAP R/3 system defined
    in the system attribute. The Print Interface Callback daemon initially
    contacts this server instance to obtain configuration information about the
    LOMs defined on that SAP R/3 system.

client = clientname
    The name of the client defined to SAP for use by the Callback daemon.

userid = userid
    The user ID defined to SAP for use by the Callback daemon.

password = password
    The password defined to SAP for use by the user ID.

language = language
    The language used for messages from the SAP R/3 application server.
```

Usage notes

1. All attributes are required.
2. The configuration file resides in the directory specified in the AOPSAPD_CONF environment variable. If this environment variable is not set, the default location is `/etc/Printsrv/aopsapd.conf`.
3. To create this configuration file, copy sample file `/usr/lpp/Printsrv/samples/aopsapd.conf` to `/etc/Printsrv/aopsapd.conf` or to the file location you specify in the AOPSAPD_CONF environment variable.
4. If you edit the configuration file after starting Infoprint Server, stop and restart all Infoprint Server daemons so that your changes take effect. For example, you can enter these MVS commands:
AOPSTART AOPSTOP
AOPSTART AOPSTART

Also, restart the SAP R/3 Logical Output Management Systems (LOMS).

5. When you edit this file, follow these syntax rules:

- End the last attribute in each entry with a semicolon.
- Use lowercase characters for the attribute names and uppercase or lowercase characters for the attribute values. Code blank characters before or after the equal sign if desired.
- If a value contains blank characters or special characters (such as {}-> =, #), enclose the value in single or double quotation marks.
- Start comments with the # symbol.
- Include blank lines if desired.

Examples -- aopsapd.conf

This example shows two sample entries in the **aopsapd.conf** configuration file.

```
system EZ0
  initial-target = boefcs1_EZ0_00
  client = 000
  userid = mike
  password = xxxxxxx
  language = EN
;

system EX0
  initial-target = boefcs5_EX0_00
  client = 000
  userid = mike
  password = xxxxxxx
  language = EN
;
```

Appendix C. Environment variables

This appendix summarizes the environment variables that affect processing of Infoprint Server. Table 56 summarizes the variables and indicates where to set each variable. You set environment variables in one or more of these locations:

- **/etc/profile** file. For information about how to set variables, see “Setting environment variables for the Printer Inventory Manager” on page 43.
- **aopstart EXEC**. For security reasons, Infoprint Server daemons use some variables only if they are set in the **aopstart EXEC**. Daemons do not use the same variables set in the **/etc/profile** file. For information about how to set variables, see “Editing the aopstart EXEC” on page 57.

Note: If dynamic configuration is enabled, Infoprint Server ignores the environment variables specified in the **aopstart EXEC** that are marked with an asterisk in the table. You must specify the equivalent attribute in the system configuration definition in the Printer Inventory. For the equivalent attributes, see Table 11 on page 70.

- STDENV data definition (DD) of the AOPSTART JCL procedure. For information about how to set variables, see “Customizing startup and shutdown commands and procedures” on page 56.
- STDENV data definition (DD) of the IP PrintWay basic mode startup procedure. For information about how to set variables, see “Editing the ANFWPROC startup procedure (basic mode)” on page 321.
- z/OS HTTP Server environment variables file. You must set variables in this file only if you use Infoprint Central. For information about how to set variables, see “Editing the z/OS HTTP Server environment variables file (httpd.envvars)” on page 335.

Rules:

- If you specify an environment variable in the AOPSTART JCL procedure (STDENV DD), you do not need to specify it in the **aopstart EXEC**.
- If you specify an environment variable in the AOPSTART JCL procedure (STDENV DD) or the **aopstart EXEC**, also specify it in the other files if applicable (**etc/profile** file, IP PrintWay basic mode startup procedure (STDENV DD), **httpd.envvars** file). For example, you must set the AOPCONF environment variable in the **aopstart EXEC**, in the **/etc/profile** file, and in the **httpd.envvars** file.

You must always set **Required** variables. You must set **optional** variables if the default value is not suitable for your installation.

Table 56. Where to set environment variables

Environment variable	/etc/profile file	aopstart EXEC	AOPSTART JCL procedure (STDENV DD)	IP PrintWay basic mode startup procedure (STDENV DD)	z/OS HTTP Server environment variables file (httpd.envvars)
_BPX_SHAREAS	No	Required	No	No	No
_BPX_UNLIMITED_OUTPUT	No	Required	No	No	No

Table 56. Where to set environment variables (continued)

Environment variable	/etc/profile file	aopstart EXEC	AOPSTART JCL procedure (STDENV DD)	IP PrintWay basic mode startup procedure (STDENV DD)	z/OS HTTP Server environment variables file (httpd.envvars)
_BPXK_SETIBMOPT_TRANSPORT	No	Optional	No	No	No
_ICONV_MODE	No	Optional	No	Optional	No
AOP_64K_JOBID	No	No	No	No	No
AOP_ALLOW_ALL_CHARACTERS_IN_LINE_DATA ¹	No	Optional	No	No	No
AOP_APPLID ¹	No	Optional	No	No	No
AOP_BLANK_TRUNCATION_CLASSES ¹	No	Optional	No	No	No
AOP_IGNORE_DCF_ROUTING_ERRORS ¹	No	Optional	No	No	No
AOP_MAIL_DO_NOT_ADD_SUFFIX ¹	No	Optional	No	Optional	No
AOP_MAIL_PRESERVE_SUFFIXES ¹	No	Optional	No	Optional	No
AOP_MAIL_USE_FIRST_ADDRESS ¹	No	Optional	No	No	No
AOP_MAXTHREADTASKS ¹	No	Optional	No	No	No
AOP_MVS_RETURN_CODES	Optional	No	No	No	No
AOP_READ_COMMUNITY	No	No	No	No	Optional
AOP_SAP2AFP_RESOURCES ²	Optional	Optional	No	No	No
AOP_SELECT_WORK_FROM_HOLD_QUEUE ¹	No	Optional	No	No	No
AOP_SUPPRESS_POST_UNIX_FILTER_FORMATTING ¹	No	Optional	No	No	No
AOP_WRITE_COMMUNITY	No	No	No	No	Optional
AOPCENTRAL	No	No	No	No	Optional
AOPCENTRAL_CODEPAGE	No	No	No	No	Optional
AOPCONF	Optional	Optional	No	No	Optional
AOPDUMPON ³	Optional	Optional	Optional	No	No
AOPHINVD_MAXTHREADTASKS ¹	No	Optional	No	No	No
AOPIPPD_MAXTHREADTASKS ¹	No	Optional	No	No	No
AOPLIMIT	No	No	No	No	Optional
AOPLIMIT_DOCS	No	No	No	No	Optional
AOPLIMIT_JOBS	No	No	No	No	Optional
AOPLPD_MAXTHREADTASKS ¹	No	Optional	No	No	No
AOPMAILER ¹	No	Optional ³	No	Optional	No
AOPMAILER_OPTIONS ¹	No	Optional	No	Optional	No
AOPMSG_CONF	Optional	Optional	No	No	No
AOPNETD_MAXTHREADTASKS ¹	No	Optional	No	No	No
AOPNETD_USE_FIXED_JOBID ¹	No	Optional	No	No	No
AOPNETD_USE_UNALTERED_JOBID ¹	No	Optional	No	No	No
AOPOPTIONS	Optional	No	No	No	No

Table 56. Where to set environment variables (continued)

Environment variable	/etc/profile file	aopstart EXEC	AOPSTART JCL procedure (STDENV DD)	IP PrintWay basic mode startup procedure (STDENV DD)	z/OS HTTP Server environment variables file (httpd.envvars)
AOPOUTD_MAXTHREADTASKS ¹	No	Optional	No	No	No
AOPPATH	Optional	No	No	No	No
AOPRESUBMITUSER	No	Optional	No	No	No
AOPRXF_CONF	Optional	Optional	No	No	No
AOPSAPD_CONF	Optional	Optional	No	No	No
AOPSDBD_MAXTHREADTASKS ¹	No	Optional	No	No	No
AOPSSID_MAXTHREADTASKS ¹	No	Optional	No	No	No
AOPSUBD_MAXTHREADTASKS ¹	No	Optional	No	No	No
AOPTRACEBYTES ³	Optional	Optional	Optional	No	No
AOPTRACEDIR ³	No	Optional	No	No	Optional
AOPTRACEMAXFILES ³	Optional	Optional	Optional	No	No
AOPTRACEON ³	Optional	Optional	Optional	No	Optional
AOPTRACEON_DATA ³	Optional	Optional	Optional	No	No
AOPTRACEON_FILTER ³	Optional	Optional	Optional	No	No
AOPTRACEON_MSGLOG ³	Optional	Optional	Optional	No	No
AOPVALIDATEDB	Optional	Optional	Optional	No	No
AOPWSMD_MAXTHREADTASKS ¹	No	Optional	No	No	No
AOPXFD_CONF	Optional	Optional	No	No	No
CLASSPATH	No	Optional	No	No	Optional
JAVA_HOME	No	Optional	No	No	Optional
LANG	Optional	Optional	Optional	No	Optional
LC_ALL	Optional	Optional	Optional	No	Optional
LC_CTYPE	Optional	Optional	Optional	No	No
LC_TIME	Optional	Optional	Optional	No	Optional
LIBPATH	Required	Required	No	Optional	Required
MANPATH	Required	No	No	No	No
NLSPATH	Required	Required	No	Optional	Required
PATH	Required	Required	No	Optional	No
PRINTER or LPDEST	Optional	No	No	No	No
STEPLIB	Optional	Optional	No	No	No
TZ	Optional	Optional	Optional	No	Optional

Table 56. Where to set environment variables (continued)

Environment variable	/etc/profile file	aopstart EXEC	AOPSTART JCL procedure (STDENV DD)	IP PrintWay basic mode startup procedure (STDENV DD)	z/OS HTTP Server environment variables file (httpd.envvars)
Notes:					
1. This environment variable is equivalent to an attribute in the system configuration definition. If dynamic configuration is enabled, do not specify this environment variable in the aopstart EXEC . Infoprint Server ignores it if specified. For the equivalent attributes in the system configuration definition, see the description of the environment variable.					
2. The SAP to AFP transform uses this environment variable. For information about it, see <i>IBM Infoprint Transforms to AFP for z/OS</i> .					
3. These environment variables control tracing and dumping. For more information about them, see <i>z/OS Infoprint Server Messages and Diagnosis</i> .					

Variable	Meaning
----------	---------

_BPX_SHAREAS

Causes the spawn() callable service to create some Infoprint Server subtasks in the parent's address space. This variable is set to _BPX_SHAREAS=YES in the **aopstart EXEC**. Do not remove this variable or change its value.

_BPX_UNLIMITED_OUTPUT

Lets Infoprint Server daemons write an unlimited amount of data to the JES spool without ending abnormally. This variable is set to _BPX_UNLIMITED_OUTPUT=YES in the **aopstart EXEC**. Do not remove this variable or change its value.

_BPXK_SETIBMOPT_TRANSPORT

The TCP/IP job name associated with the TCP/IP stack you want to use. This variable is optional. However, you must set it if your installation has defined multiple TCP/IP stacks on the z/OS system and you start the Print Interface LPD or IP PrintWay extended mode. (The Print Interface LPD starts by default.) For more information about this environment variable, see *z/OS UNIX System Services Planning*.

_ICONV_MODE

Determines whether **iconv** uses customized TCP/IP code page translation tables. Set the value to C if you have customized any TCP/IP code page translation tables.

Example:

_ICONV_MODE=C

AOP_64K_JOBID

If this environment variable is defined in the process that calls the Infoprint Server API, Infoprint Server assigns job IDs in the range of 1 to 65535. If this environment variable is not defined, job IDs can be in the range of 1 to 999999.

Note: This variable is defined for use by the z/OS SMB server. Other applications should not define this variable because system performance might be degraded.

AOP_ALLOW_ALL_CHARACTERS_IN_LINE_DATA

This environment variable determines how Infoprint Server detects the data format of documents that contain line data. Valid line data contains carriage control characters, table reference characters, binary zeroes, and printable characters. However some applications might create line data that contains unprintable characters.

If this environment variable is set to YES, if line data contains unprintable characters, Infoprint Server detects the data format as "line data".

If this environment variable is not set, if line data contains unprintable characters, Infoprint Server detects the data format as "other". Line data that Infoprint Server detects as "other" might not print correctly because Infoprint Server does not convert it to the printer's code page.

Tips:

- Job submitters can specify the data format in the **document-format** job attribute. For example, job submitters can specify **document-format=line** in the PRTATTRS parameter on the OUTPUT JCL statement. If the **document-format** job attribute is specified, Infoprint Server ignores this environment variable because it does not detect the data format.
- Set this environment variable to YES if Infoprint Server does not detect the line data format and job submitters cannot specify the **document-format=line** job attribute.
- This environment variable does not affect how NetSpool checks SCS and 3270 data streams for validity. The input SCS and 3270 data streams must contain only valid characters.

Rules:

- If dynamic configuration is enabled, specify the equivalent **allow-all-characters-in-line-data** attribute in the system configuration definition.
- If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

Default: The environment variable is not set.

Example: AOP_ALLOW_ALL_CHARACTERS_IN_LINE_DATA=YES

AOP_APPLID The application program ID that IP PrintWay extended mode uses to establish a VTAM session with printers. This ID must match the name of the APPL statement defined to VTAM in the SYS1.VTAMLST data set. This environment variable is required to use IP PrintWay extended mode to print on VTAM-controlled printers.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **applid** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- This environment variable does not apply to IP PrintWay basic mode.

Default: The environment variable is not set.

Example: AOP_APPLID=PWAPPL

AOP_BLANK_TRUNCATION_CLASSES

The list of JES output classes for which IP PrintWay extended mode removes blanks from the end of records or lines. If a data set is allocated in one of the listed classes, and the data set contains either line data or text data, IP PrintWay removes any blank characters that occur at the end of each record or line. If this environment variable is not set, IP PrintWay does not remove any blank characters.

Values: One or more letters or numbers. Specify multiple letters and numbers with no separation. An asterisk (*) indicates all JES output classes. Lowercase letters are converted to uppercase.

Tips:

- Specify this environment variable only if line or text data prints with extra blank characters at the end of lines.
- This environment variable can slow performance and cause IP PrintWay to use more system resources.
- Before giving data to IP PrintWay extended mode, JES always restores blanks that it truncated from data sets to save space on the JES spool. JES restores blanks regardless of how you set the JES2 BLNKTRNC and JES3 TRUNC parameters. Therefore, if you specify either BLNKTRNC=YES (JES2) or TRUNC=YES (JES3), you might need to specify this environment variable if your printed output contains unwanted blank characters. For example, if you specify BLNKTRNC=YES for output classes C and D, you could specify AOP_BLANK_TRUNCATION_CLASSES=CD to request that IP PrintWay extended mode remove trailing blank characters for output classes C and D. (BLNKTRNC=YES and TRUNC=YES are the JES defaults.)

For information about specifying JES blank truncation, see one of these:

- *z/OS JES2 Initialization and Tuning Guide*
- *z/OS JES3 Initialization and Tuning Guide*

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **blank-truncation-classes** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- This environment variable does not apply to IP PrintWay basic mode.

Default: The environment variable is not set.

Example: AOP_BLANK_TRUNCATION_CLASSES=ABC123

AOP_IGNORE_DCF_ROUTING_ERRORS

If this environment variable is set to YES, IP PrintWay extended mode does not write an error message (AOP3201E) when it cannot

find a printer definition in the Printer Inventory that matches the DEST, CLASS, and FORMS parameters specified on the OUTPUT (or DD) JCL statement for a print job. Also, it does not save information about the print job in the Printer Inventory.

If this environment variable is not set, IP PrintWay extended mode writes an error message (AOP3201I) when it cannot find a printer definition that matches the DEST, CLASS, and FORMS parameters. Also, it saves information about the print job in the Printer Inventory so that Infoprint Central can display the print job and the associated AOP3201E message.

Tips:

- Set this environment variable to YES if your IP PrintWay job-selection rules cause IP PrintWay to select a large number of jobs whose DEST, CLASS, and FORMS parameters do not match printer definitions in the Printer Inventory. (In this situation, message AOP3201E occurs numerous times in the common message log.) Setting this environment variable to YES can conserve storage and processing resources.
- If you set this environment variable to YES, it might be difficult to determine that IP PrintWay did not process a print job because its CLASS, DEST, and FORMS parameters do not match a printer definition.
- The text of message AOP3201E is:
AOP3201E The DEST, CLASS, and FORMS JCL keywords do not match a printer definition in the Printer Inventory.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **ignore-dcf-routing-errors** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- This environment variable does not apply to IP PrintWay basic mode.

Default: The environment variable is not set.

Example: AOP_IGNORE_DCF_ROUTING_ERRORS=YES

AOP_MAIL_DO_NOT_ADD_SUFFIX

If this environment variable is set to YES, IP PrintWay does not append a suffix to any file names specified in the MAILFILE JCL parameter or the **mail-file-name** job attribute.

If this environment variable is not set, IP PrintWay appends one of these suffixes for the data formats it recognizes: afp, jpg, pcl, pdf, ps, sap, tif, txt, or octet-stream (for unrecognized data formats).

Tip: If the AOP_MAIL_DO_NOT_ADD_SUFFIX environment variable is set to YES, IP PrintWay ignores the AOP_MAIL_PRESERVE_SUFFIXES environment variable.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **mail-do-not-add-suffixes** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- If you run IP PrintWay basic mode, specify this environment variable in the IP PrintWay basic mode startup procedure.

Default: The environment variable is not set.

Example: AOP_MAIL_DO_NOT_ADD_SUFFIX=YES

AOP_MAIL_PRESERVE_SUFFIXES

The list of suffixes that IP PrintWay preserves in file names specified in the MAILFILE JCL parameter or in the **mail-file-name** job attribute. If the file name ends with one of the suffixes in this list, IP PrintWay does not append another suffix to the file name.

If no file name is specified in MAILFILE or **mail-file-name**, or if the suffix of the file name does not match a suffix in this environment variable, IP PrintWay appends one of these suffixes for the data formats it recognizes: afp, jpg, pcl, pdf, ps, sap, tif, txt, or octet-stream (for unrecognized data formats).

Tips:

- Specify the suffixes that users might specify in file names. For example, you might specify the .doc suffix for Microsoft Word documents.
- If the AOP_MAIL_DO_NOT_ADD_SUFFIX environment variable is set to YES, IP PrintWay ignores the AOP_MAIL_PRESERVE_SUFFIXES environment variable.

Rules:

- Begin each suffix with a period.
- To specify more than one suffix, separate the suffixes with a blank. Do not enclose the list of suffixes in single or double quotation marks.
- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **mail-preserve-suffixes** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- If you run IP PrintWay basic mode, specify this environment variable in the IP PrintWay startup procedure.

Default: The environment variable is not set.

Example: c

AOP_MAIL_USE_FIRST_ADDRESS

Specifies the email addresses that IP PrintWay extended mode uses when a print job (output group) contains more than one document (data set).

- NO** IP PrintWay sends each document to the email address or addresses specified for the document.
- YES** IP PrintWay sends each document to the email address or addresses specified for the first document in the print job.

Tips:

- AOP_MAIL_USE_FIRST_ADDRESS has no effect if the **Concatenate job** option is selected in the **Dataset grouping** field in the printer definition, or if the **Automatic dataset grouping** field is selected in the printer definition. When either concatenation option is selected, IP PrintWay extended mode sends all documents in a print job in one email to the email address or addresses specified for the first document in the print job.
- IP PrintWay basic mode ignores this environment variable. IP PrintWay basic mode always sends each document to the email address or addresses specified for the document unless the **Concatenate job** option is selected in the **Dataset grouping** field in the printer definition.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **mail-use-first-address** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- This environment variable does not apply to IP PrintWay basic mode.

Default: YES

Example: AOP_MAIL_USE_FIRST_ADDRESS=NO

AOP_MAXTHREADTASKS

The MAXTHREADTASKS limit for these Infoprint Server daemons: **aophinvd**, **aopippdxp**, **aoplpd**, **aopnetd**, **aopoutd**, **aopsdbd**, **aopssid**, **aopsubd**, and **aopwsmd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that a single process (daemon) can have concurrently active. This environment variable is optional.

If you set a MAXTHREADTASKS limit for a specific daemon in the MAXTHREADTASKS environment variable for that daemon, the limit in this environment variable does not apply for that daemon.

Tips:

- The default MAXTHREADTASKS limits that Infoprint Server uses are suitable for most installations. Consider setting a different limit only if you receive this message:

```
BPXI040I PROCESS LIMIT MAXTHREADTASKS HAS REACHED nn% OF ITS
CURRENT CAPACITY OF nnn FOR PID=nnnnnn IN JOB daemon
```

The BPXI040I message is an informational message and does not indicate that an error occurred. If the BPXI040I message indicates that less than 100% of capacity has been reached, threads run

without delay. However, when the limit is reached, a newly created thread does not run until an existing thread ends.

The LIMMSG statement in the BPXPRMxx member of SYS1.PARMLIB controls whether z/OS UNIX issues message BPXI040I. The default is that z/OS UNIX does not issue BPXI040I messages. You can change the LIMMSG value dynamically using the SETOMVS command.

- Do not set a value less than 32.
- If you increase the MAXTHREADTASKS limit, you might need to increase the maximum region size because each concurrent task requires additional storage:
 - If you use the AOPSTART JCL procedure to start Infoprint Server, increase the region size on the EXEC statement in the procedure.
 - If you use the **aopstart** command to start Infoprint Server, increase the maximum address space in the MAXASSIZE parameter in the BPXPRMxx member of SYS1.PARMLIB or in the ASSIZEMAX RACF parameter for the user ID that starts Infoprint Server (ASSIZEMAX overrides MAXASSIZE).
 - If you enter the **aopstart** command from the TSO/E OMVS command line, the SIZE parameter on the TSO/E LOGON panel determines the maximum region size of the address space.
- Do not set a very high limit because each concurrent task requires some storage below the 16 MB line. If the limit is too high, an abend can occur.
- If you exceed the default limit (200) when a lot of printers are waiting for operator intervention at the same time, you might need to increase the limit in the APOPOUTD_MAXTHREADTASKS environment variable or reduce the IP PrintWay response timeout value in the printer definition. For information about setting the response timeout value, see *z/OS Infoprint Server Operation and Administration*.

Rules:

- If dynamic configuration is enabled, specify the equivalent **max-thread-tasks** attribute in the system configuration definition.
- If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

Default: The limit is 200 for each daemon.

Example: In this example, the limit for the **aopoutd** daemon is 250, while the limit for all other daemons is 225:

```
APOPOUTD_MAXTHREADTASKS=250
AOP_MAXTHREADTASKS=225
```

AOP_MVS_RETURN_CODES

Specifies whether the **lp** command and transform commands return MVS return codes (0, 4, 8) or UNIX exit value (0, 1):

Values:

YES	MVS return codes:
0	Successful.

- 4 A warning occurred during the transform.
 - 8 The command was not accepted, a data stream error occurred during the transform, or the transform failed.
- NO** UNIX return codes:
- 0 Successful. However, a warning or data stream error might have occurred during the transform.
 - 1 The command was not accepted, or the transform failed.

Notes:

- For information about whether a transform command supports this environment variable, see the documentation for the transform.
- Specify this environment variable before the command when you run a command from the z/OS UNIX command line. For example, if you run the **lp** command, enter on the z/OS UNIX command line:

```
AOP_MVS_RETURN_CODES=YES lp -d printername myfile
```

You can also specify this variable in the **\$HOME/.profile** file (if you use the **/bin/sh** shell) or the **\$HOME/.tcshrc** file (if you use the **/bin/tcsh** shell).

- If you use the AOPPRINT JCL procedure, specify this environment variable in the STDENV data definition (DD) of the procedure. AOPPRINT runs the **lp** command and returns the code from the **lp** command.
- If you use the AOPBATCH JCL procedure to run the **lp** command or a transform command, specify this environment variable in the STDENV data definition (DD) of the procedure. AOPBATCH returns the code from the **lp** command or transform command.
- Do not specify this environment variable in the Infoprint Server transform configuration file, **aopxfd.conf**.

Default: NO

Example: AOP_MVS_RETURN_CODES=YES

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.

Tips:

- 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

Example: AOP_READ_COMMUNITY=mypass

AOP_SELECT_WORK_FROM_HOLD_QUEUE

Indicates whether IP PrintWay extended mode selects new held output groups from the JES spool.

Values:

- YES** IP PrintWay extended mode selects new held output groups from the JES spool. This lets you use Infoprint Central to move the held output groups to different IP PrintWay printers before releasing them.
- NO** IP PrintWay extended mode does not select new held output groups from the JES spool. This lets you use JES or SDSF commands to release individual data sets in an output group for printing.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **select-work-from-hold-queue** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- This environment variable does not apply to IP PrintWay basic mode.

Default: YES

Example: AOP_SELECT_WORK_FROM_HOLD_QUEUE = NO

AOP_SUPPRESS_POST_UNIX_FILTER_FORMATTING

Indicates whether IP PrintWay extended mode formats line or text data after a UNIX filter processes it. If this environment variable is set to any value, IP PrintWay extended mode does not format data when all these conditions are met:

- The data set to be processed contains either line data or text data.
- A UNIX filter is specified in the printer definition for the data format.
- The PAGEDEF or FORMDEF parameter is specified on the OUTPUT JCL statement for the data set. (A page or form definition specified in the printer definition does not meet this criterion.)

If this environment variable is not set, IP PrintWay extended mode always formats line or text data after a UNIX filter processes it. For

example, IP PrintWay processes carriage control characters and transparent data characters, and converts the data from the document code page to the printer code page.

Tips:

- Specify this environment variable if you use a UNIX filter that calls an AFP transform to convert line data to PCL, PDF, or PostScript format.
- This environment variable does not affect how IP PrintWay extended mode processes data when you specify a DLL filter in the printer definition. IP PrintWay extended mode does not modify data returned by DLL filters.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **suppress-post-unix-filter-formatting** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- This environment variable does not apply to IP PrintWay basic mode.

Default: IP PrintWay extended mode formats line or text data after a UNIX filter processes it.

Example:

```
AOP_SUPPRESS_POST_UNIX_FILTER_FORMATTING=YES
```

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.

Tips:

- 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

Example: AOP_WRITE_COMMUNITY=mypass

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: /usr/lpp/Printsrv/InfoprintCentral

Example: AOPCENTRAL=/usr/lpp/mylib/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

Example: AOPCENTRAL_CODEPAGE=IBM-297

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: /etc/Printsrv/aopd.conf

AOPHINVD_MAXTHREADTASKS

The MAXTHREADTASKS limit for the Historical Inventory daemon, **aophinvd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aophinvd** can have concurrently active. This environment variable is optional.

Tip: For additional information, see the “Tips” for the AOP_MAXTHREADTASKS environment variable.

Rules:

- If dynamic configuration is enabled, specify the equivalent **aophinvd-max-thread-tasks** attribute in the system configuration definition.
- If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

Default: The limit specified in the AOP_MAXTHREADTASKS environment variable is used. If that environment variable is not specified, the limit is 200.

AOPIPPD_MAXTHREADTASKS

The MAXTHREADTASKS limit for the Internet Printing Protocol (IPP) daemon, **aopippd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aopippd** can have concurrently active. This environment variable is optional.

Tip: For additional information, see the “Tips” for the AOP_MAXTHREADTASKS environment variable.

Rules:

- If dynamic configuration is enabled, specify the equivalent **aopipd-max-thread-tasks** attribute in the system configuration definition.
- If dynamic configuration is not enabled, specify this environment variable in the **aopstart EXEC**.

Default: The limit specified in the AOP_MAXTHREADTASKS environment variable is used. If that environment variable is not specified, the limit is 200.

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.

If you set a limit for print jobs in the AOPLIMIT_JOBS environment variable, the limit in AOPLIMIT does not apply to print jobs.

If you set a limit for documents in the AOPLIMIT_DOCS environment variable, the limit in AOPLIMIT does not apply to documents.

Tips:

- 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.
- When the HTTP Server times out, users might not see any search results, or might see partial search results or a "Page not found" error.

Default: 250

Example: AOPLIMIT=100

AOPLIMIT_DOCS

The maximum number of documents 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.

For additional information, see the AOPLIMIT environment variable.

Default: The limit specified in the AOPLIMIT environment variable is used. If that environment variable is not specified, the limit is 250.

Example: AOPLIMIT_DOCS=100

AOPLIMIT_JOBS

The maximum number of print jobs 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.

For additional information, see the AOPLIMIT environment variable.

Default: The limit specified in the AOPLIMIT environment variable is used. If that environment variable is not specified, the limit is 250.

Example: AOPLIMIT_JOBS=100

AOPLPD_MAXTHREADTASKS

The MAXTHREADTASKS limit for the Infoprint Server LPD, **aoplpd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aoplpd** can have concurrently active. This environment variable is optional.

Tip: For additional information, see the "Tips" for the AOP_MAXTHREADTASKS environment variable.

Rules:

- If dynamic configuration is enabled, specify the equivalent **aoplpd-max-thread-tasks** attribute in the system configuration definition.
- If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

Default: The limit specified in the AOP_MAXTHREADTASKS environment variable is used. If that environment variable is not specified, the limit is 200.

AOPMAILER Specifies the full path name of the z/OS UNIX **sendmail** command that IP PrintWay uses to send output to email destinations. This environment variable is optional. However, you must set it if you did not install the **sendmail** executable file in the default **/bin** directory.

Rules:

- Start the path name with a slash (/).
- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **mailer-path-name** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- If you run IP PrintWay basic mode, specify this environment variable in the IP PrintWay basic mode startup procedure.

Default: /bin/sendmail

Example: AOPMAILER=/mydirectory/sendmail

AOPMAILER_OPTIONS

Options for the z/OS UNIX **sendmail** command. IP PrintWay specifies these options in addition to the **-i** option when it sends

emails. You can specify this environment variable regardless of whether you specify the AOPMAILER environment variable. This environment variable is optional.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **mailer-options** attribute in the system configuration definition.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
- If you run IP PrintWay basic mode, specify this environment variable in the IP PrintWay basic mode startup procedure.

Default: The environment variable is not set.

Example: AOPMAILER_OPTIONS=-Am

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 this configuration file or if you created the configuration file in the default location, you do not need to set this environment variable.

Default: /etc/Printsrv/aopmsg.conf

AOPNETD_MAXTHREADTASKS

The MAXTHREADTASKS limit for the NetSpool daemon, **aopnetd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aopnetd** can have concurrently active. This environment variable is optional.

Tip: For additional information, see the “Tips” for the AOP_MAXTHREADTASKS environment variable.

Rules:

- If dynamic configuration is enabled, specify the equivalent **aopnetd-max-thread-tasks** attribute in the system configuration definition.
- If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

Default: The limit specified in the AOP_MAXTHREADTASKS environment variable is used. If that environment variable is not specified, the limit is 200.

AOPNETD_USE_FIXED_JOBID

If this environment variable is set to YES, NetSpool assigns the same job ID to all output data sets that it creates on the JES spool. The job ID is the value specified in the **job-prefix** attribute, which defaults to PS, followed by 000000. For example, PS000000.

If this environment variable is not set, and environment variable AOPNETD_USE_UNALTERED_JOBID is not set, NetSpool assigns a unique job ID to each data set that it creates. This ID is the same as the job ID that Infoprint Server assigns to each print job.

When Infoprint Central displays Infoprint Server print jobs, it displays the unique Job ID that Infoprint Server assigns to each print job. Therefore, if this environment variable is set to YES,

Infoprint Central functions that need to correlate JES print jobs with Infoprint Server print jobs do not work.

Tip: Set this environment variable only if you run JES3 and have a very large number of PSF FSAs or IP PrintWay basic mode FSAs. Contact IBM for more information.

Rules:

- If dynamic configuration is enabled, specify the equivalent **netspool-use-fixed-jobid** attribute in the system configuration definition.
- If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

Default: The environment variable is not set.

Example: AOPNETD_USE_FIXED_JOBID=YES

AOPNETD_USE_UNALTERED_JOBID

If this environment variable is set to YES, NetSpool does not assign a job ID to the output data sets that it creates on the JES spool. The job ID for each data set is the same as the job ID of the **aopnetd** daemon.

If this environment variable is not set, and environment variable AOPNETD_USE_FIXED_JOBID is not set, NetSpool assigns a unique job ID to each data set that it creates. This ID is the same as the job ID that Infoprint Server assigns to each print job.

When Infoprint Central displays Infoprint Server print jobs, it displays the unique job ID that Infoprint Server assigns to each print job. Therefore, if this environment variable is set to YES, Infoprint Central functions that need to correlate JES print jobs with Infoprint Server print jobs do not work.

Rules:

- If dynamic configuration is enabled, specify the equivalent **netspool-use-unaltered-jobid** attribute in the system configuration definition.
- If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

Default: The environment variable is not set.

Example: AOPNETD_USE_UNALTERED_JOBID=YES

AOPOPTIONS

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 AOPOPTIONS environment variable before the options you specify on the command line, a user can override the values of this variable. This variable is optional. However, you might want to set it to specify job attributes that are constant for all print jobs. For the format of job attributes and values, see *z/OS Infoprint Server User's Guide*.

AOPOUTD_MAXTHREADTASKS

The MAXTHREADTASKS limit for the IP PrintWay extended mode daemon, **aopoutd**. The MAXTHREADTASKS limit is the maximum

number of MVS tasks that **aopoutd** can have concurrently active. This environment variable is optional.

Tips:

- If you exceed the default limit (200) when a lot of printers are waiting for operator intervention at the same time, you might need to increase the limit in this environment variable or reduce the IP PrintWay response timeout value in the printer definitions. For information about setting the response timeout value, see *z/OS Infoprint Server Operation and Administration*.
- See the “Tips” for the AOP_MAXTHREADTASKS environment variable.

Rules:

- If dynamic configuration is enabled, specify the equivalent **aopoutd-max-thread-tasks** attribute in the system configuration definition.
- If dynamic configuration is not enabled, specify this environment variable in the **aopstart EXEC**.

Default: The limit specified in the AOP_MAXTHREADTASKS environment variable is used. If that environment variable is not specified, the limit is 200.

AOPPATH

Path where job attribute files, used by the **lp** command and AOPPRINT procedure, are located. This environment variable is optional. However, you must set it if you create attribute files for the **lp** command. For information about attribute files, see *z/OS Infoprint Server User's Guide*.

AOPRESUBMITUSER

If this environment variable is set to YES, the job submitter's z/OS user ID that IP PrintWay sends to the Infoprint Server LPD during the resubmit for filtering function is considered an authenticated z/OS user ID for the purpose of doing RACF checks. Infoprint Server uses the authenticated z/OS user ID when it checks the job submitter's RACF access to AFP user resource libraries, which are specified in the USERLIB JCL parameter or in the printer definition.

The z/OS system authenticates this z/OS user ID and the associated password during job submission. However, IP PrintWay sends it over a potentially unsecured network to the Infoprint Server LPD.

If this environment variable is not set, Infoprint Server considers the job submitter's z/OS user ID to be an *unauthenticated* user ID. This is because an untrusted source could send an unauthenticated user ID to the Infoprint Server LPD in the same manner as IP PrintWay basic mode does, and because the network is potentially unsecured. Instead of using the job submitter's user ID to do RACF checks, Infoprint Server uses ID *NONE*.

Tips:

- For a secure system, do not set this environment variable.
- Set this environment variable to YES only if these conditions exist:

- All users on your z/OS system are trusted, all users who submit print requests to the Infoprint Server LPD are trusted, and your network is protected.
- You receive this message for data sets submitted with JCL: AOP092E *NONE* IS NOT AUTHORIZED TO READ *resource library*. *NONE* in the message text indicates that Infoprint Server considers the user ID to be unauthenticated.
- Instead of specifying this environment variable, do one of these:
 - Run IP PrintWay extended mode. IP PrintWay extended mode transforms data without using the resubmit for filtering function.
 - Ask job submitters who receive message AOP092E to print using a job submission method that does not involve the resubmit for filtering function. For example, job submitters can use the **lp** command, the Print Interface subsystem, or the AOPPRINT JCL procedure.
 - Give RACF universal READ access to AFP user resource libraries that do not need to be read-protected.

Default: The environment variable is not set.

Example: AOPRESUBMITUSER=YES

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 this environment variable.

Default: /etc/Printsrv/aoprxf.conf

AOPSAPD_CONF

The full path name of the SAP Callback daemon configuration file. 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 this environment variable.

Default: /etc/Printsrv/aopsapd.conf

AOPSDBD_MAXTHREADTASKS

The MAXTHREADTASKS limit for the search database daemon, **aopsdbd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aopsdbd** can have concurrently active. This environment variable is optional.

Tip: For additional information, see the “Tips” for the AOP_MAXTHREADTASKS environment variable.

Rules:

- If dynamic configuration is enabled, specify the equivalent **aopsdbd-max-thread-tasks** attribute in the system configuration definition.
- If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

Default: The limit specified in the AOP_MAXTHREADTASKS environment variable is used. If that environment variable is not specified, the limit is 200.

AOPSSID_MAXTHREADTASKS

The MAXTHREADTASKS limit for the Infoprint Central daemon, **aopssid**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aopssid** can have concurrently active. This environment variable is optional.

Tip: For additional information, see the “Tips” for the AOP_MAXTHREADTASKS environment variable.

Rules:

- If dynamic configuration is enabled, specify the equivalent **aopssid-max-thread-tasks** attribute in the system configuration definition.
- If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

Default: The limit specified in the AOP_MAXTHREADTASKS environment variable is used. If that environment variable is not specified, the limit is 200.

AOPSUBD_MAXTHREADTASKS

The MAXTHREADTASKS limit for the Print Interface subsystem daemon, **aopsubd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aopsubd** can have concurrently active. This environment variable is optional.

Tip: For additional information, see the “Tips” for the AOP_MAXTHREADTASKS environment variable.

Rules:

- If dynamic configuration is enabled, specify the equivalent **aopsubd-max-thread-tasks** attribute in the system configuration definition.
- If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

Default: The limit specified in the AOP_MAXTHREADTASKS environment variable is used. If that environment variable is not specified, the limit is 200.

AOPVALIDATEDB

If this environment variable is 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: The environment variable is not set.

Example: AOPVALIDATEDB=YES

AOPWSMD_MAXTHREADTASKS

The MAXTHREADTASKS limit for the IP PrintWay extended mode daemon, **aopwsmd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aopwsmd** can have concurrently active. This environment variable is optional.

Tip: For additional information, see the “Tips” for the AOP_MAXTHREADTASKS environment variable.

Rules:

- If dynamic configuration is enabled, specify the equivalent **aopwsmd-max-thread-tasks** attribute in the system configuration definition.
- If dynamic configuration is not enabled, specify this environment variable in the **aopstart EXEC**.

Default: The limit specified in the AOP_MAXTHREADTASKS environment variable is used. If that environment variable is not specified, the limit is 200.

AOPXFD_CONF

The full path name of the transform configuration file, **aopxfd.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 this environment variable.

Default: /etc/Printsrv/aopxfd.conf

CLASSPATH

The full path names of Infoprint Server Java Archive (JAR) files. This environment variable is optional. However, you must set it in these situations:

- You start the IPP Server or Infoprint Central and you did not install Infoprint Server in the default directory (**/usr/lpp/Printsrv**).
- You start Infoprint Central and the CLASSPATH environment variable is set to other values in the HTTP Servers' environment variables file (**httpd.envvars**).

Infoprint Central: Add these values to any existing values:

- /*directory*/classes/lib/ipa.jar
- /*directory*/classes/lib/modelplugin.jar
- /*directory*/classes/lib/penguin.jar
- /*directory*/classes/lib/snmp.jar

IPP Server: Specify these values:

- /*directory*/classes/ipp.jar
- /*directory*/classes/ippreal.jar
- /*directory*/classes/ippserver.jar

directory is the directory where you installed Infoprint Server.

Rules:

- If you start the IPP Server, set the CLASSPATH environment variable in the **aopstart EXEC**.
- If you start Infoprint Central, set the CLASSPATH environment variable in the HTTP Server environment variables file, **httpd.envvars**.

Default:

IPP Server:

```
/usr/lpp/Printsrv/classes/ipp.jar:  
/usr/lpp/Printsrv/classes/ippreal.jar:  
/usr/lpp/Printsrv/classes/ippserver.jar
```

Infoprint Central:


```
/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 might need to set it if the default value is not suitable for the IPP Server or Infoprint Central. For the Java requirements, see:

- IPP Server: “Identifying software requirements for Print Interface” on page 116
- Infoprint Central: “Identifying software requirements for Infoprint Central” on page 328

Rules:

- IPP Server: Set the JAVA_HOME environment variable in the **aopstart** EXEC.
- Infoprint Central: Set the JAVA_HOME environment variable in the HTTP Server environment variables file, **httpd.envvars**.

Default: The IPP Server and Infoprint Central use Java files in these directories, in the order shown:

1. /usr/lpp/java/J7.0
2. /usr/lpp/java/J6.0
3. /usr/lpp/java/J6.0.1

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. This locale overrides the locale in the LC_TIME variable.

Rule: In the z/OS HTTP environment variables file (**httpd.envvars**), do not specify a value that is associated with a code page other than IBM-1047.

Examples:

aopstart EXEC: LC_ALL=Fr_FR.IBM-297

httpd.envvars file: LC_ALL=Fr_FR.IBM-1047

Default: C (also called POSIX)

LC_CTYPE The locale that determines the EBCDIC code page used to validate Infoprint Server attribute values.

Rules: In the z/OS HTTP Server environment variables file (**httpd.envvars**):

- Do not specify the LC_CTYPE environment variable.
- Specify the AOPCENTRAL_CODEPAGE environment variable with the same EBCDIC code page as specified in the LC_CTYPE environment variable. For example:
AOPCENTRAL_CODEPAGE=Fr_FR.IBM-297

Example: LC_CTYPE=Fr_FR.IBM-297

Default: C (also called POSIX). The default code page is IBM-1047.

- 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)
- LIBPATH** The path used to locate dynamic link libraries (DLLs). This environment variable is required.
- If you installed Infoprint Server in the default directory, add **/usr/lpp/Printsrv/lib** to any existing values. If you installed Infoprint Server in a different directory, add that directory to any existing values.
- If you run the SAP Output Management System (OMS), also add the directory that contains the SAP remote function calls library:
librfc.dll
- If you run Infoprint Central, also add the IBM XML Toolkit V1R10 libraries before the libraries for any other XML Toolkit versions in the z/OS HTTP Server environment variables file (**httpd.envvars**):
/usr/lpp/ixm/IBM/xml4c-5_7/lib
/usr/lpp/ixm/IBM/xslt4c-1_11/lib
/usr/lpp/Printsrv/lib
- Default:** /usr/lpp/Printsrv/lib
- This default is set in the **aopstart** EXEC. You must specify this environment variable in the **/etc/profile** file. If you run Infoprint Central, you must also specify this environment variable in the **httpd.envvars** file.
- 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.
- Note:** Add the new directory path *before* **/usr/man/%L** in the MANPATH environment variable, so that the Infoprint Server versions of the **lp**, **lpstat**, and **cancel** man pages are displayed.
- Default:** None.
- 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 is set in the **aopstart** EXEC. You must specify this environment variable in the **/etc/profile** file.
- 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 values in this variable. 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 is set in the `aopstart` EXEC. You must specify this environment variable in the `/etc/profile` file.

PRINTER or LPDEST

The name of the default printer for the `lp` command. The value in LPDEST overrides the value in PRINTER.

This variable is optional. You can, instead, use Infoprint Server ISPF panels to define the name of the default printer. For information, see *z/OS Infoprint Server Operation and Administration*. The printer named in either LPDEST or PRINTER overrides the printer named on the ISPF configuration panel.

STEPLIB

The names of any step libraries that are not in the system LNKLST, such as:

- Language Environment runtime libraries (*hlq*.SCEERUN and *hlq*.SCEERUN2)
- C++ runtime library (*hlq*.SCLBDLL)
- The library that contains optional IP PrintWay extended mode exits.
- SYS1.IMAGELIB if you want IP PrintWay extended mode to use FCBs to format data with carriage controls.

Rule: Specify these step libraries in the file specified by the STEPLIBLIST statement in the BPXPRMxx PARMLIB member. See “Updating the BPXPRMxx member of SYS1.PARMLIB (extended mode)” on page 207.

Example: If you use the standard high-level qualifiers, specify:

STEPLIB=CEE.SCEERUN:CEE.SCEERUN2:CBC.SCLBDLL:PW.EXITLIB:SYS1.IMAGELIB

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.

For information about the allowed values, see *z/OS UNIX System Services Command Reference*.

Default: The z/OS HTTP Server default is EST5EDT (Eastern U.S. time zone). The default for the Printer Inventory Manager is GMT0.

Related information: For information about how to access the Language Environment runtime libraries and improve the performance of runtime libraries, see *z/OS UNIX System Services Planning*.

Appendix D. Configuration attributes

This appendix summarizes the configuration attributes in the system configuration definition in the Printer Inventory. When dynamic configuration is enabled, you can specify these attributes using the Infoprint Server System Configuration ISPF panel or the Printer Inventory Definition Utility (PIDU).

Note: The examples for each attribute show how to specify the attribute when you use PIDU to edit the system configuration definition. If you use the ISPF panel to edit the system configuration definition, use the online help provided with the ISPF panel for information about how to specify values.

allow-all-characters-in-line-data = yes | no

This attribute determines how Infoprint Server detects the data format of documents that contain line data. Valid line data contains carriage control characters, table reference characters, binary zeroes, and printable characters. However some applications might create line data that contains unprintable characters.

yes Infoprint Server detects the data format as “line data” even if the input data contains unprintable characters.

no Infoprint Server detects the data format as “other” if the input data contains unprintable characters. Line data that Infoprint Server detects as “other” might not print correctly because Infoprint Server does not convert it to the printer's code page.

Tips:

- Job submitters can specify the data format in the **document-format** job attribute. For example, job submitters can specify **document-format=line** in the PRTATTRS parameter on the OUTPUT JCL statement. If the **document-format** job attribute is specified, Infoprint Server ignores this attribute because it does not detect the data format.
- Specify **allow-all-characters-in-line-data=yes** if Infoprint Server does not detect the line data format and job submitters cannot specify the **document-format=line** job attribute.
- This attribute does not affect how NetSpool checks SCS and 3270 data streams for validity. The input SCS and 3270 data streams must contain only valid characters.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify the equivalent AOP_ALLOW_ALL_CHARACTERS_IN_LINE_DATA environment variable in the **aopstart** EXEC.
- This attribute does not apply to IP PrintWay basic mode.

Default: allow-all-characters-in-line-data = no

Example: allow-all-characters-in-line-data = yes

aophinvd-max-thread-tasks = nnn

The MAXTHREADTASKS limit for the Historical Inventory daemon,

aophinvd. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aophinvd** can have concurrently active.

Tip: See the “Tips” for the **max-thread-tasks** attribute.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop all Infoprint Server daemons and restart Infoprint Server.
- If dynamic configuration is not enabled, specify the equivalent AOPHINVD_MAXTHREADTASKS environment variable in the **aopstart EXEC**.

Default: The limit specified in the **max-thread-tasks** attribute is used. If that attribute is not specified, the limit is 200.

aopippd-max-thread-tasks = nnn

The MAXTHREADTASKS limit for the Internet Printing Protocol (IPP) daemon, **aopippd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aopippd** can have concurrently active.

Tip: See the “Tips” for the **max-thread-tasks** attribute.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop **aopippd** and restart Infoprint Server.
- If dynamic configuration is not enabled, specify the equivalent AOIPPD_MAXTHREADTASKS environment variable in the **aopstart EXEC**.

Default: The limit specified in the **max-thread-tasks** attribute is used. If that attribute is not specified, the limit is 200.

aoplpd-max-thread-tasks = nnn

The MAXTHREADTASKS limit for the Infoprint Server LPD, **aoplpd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aoplpd** can have concurrently active.

Tip: See the “Tips” for the **max-thread-tasks** attribute.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop **aoplpd** and restart Infoprint Server.
- If dynamic configuration is not enabled, specify the equivalent AOPLPD_MAXTHREADTASKS environment variable in the **aopstart EXEC**.

Default: The limit specified in the **max-thread-tasks** attribute is used. If that attribute is not specified, the limit is 200.

aopnetd-max-thread-tasks = nnn

The MAXTHREADTASKS limit for the NetSpool daemon, **aopnetd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aopnetd** can have concurrently active.

Tip: See the “Tips” for the **max-thread-tasks** attribute.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop **aopnetd** and restart Infoprint Server.
- If dynamic configuration is not enabled, specify the equivalent AOPNETD_MAXTHREADTASKS environment variable in the **aopstart EXEC**.

Default: The limit specified in the **max-thread-tasks** attribute is used. If that attribute is not specified, the limit is 200.

aopoutd-max-thread-tasks = nnn

The MAXTHREADTASKS limit for the IP PrintWay extended mode daemon **aopoutd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aopoutd** can have concurrently active.

Tips:

- If you exceed the default limit (200) when many printers are simultaneously waiting for operator intervention, you might need to increase the limit in this attribute or reduce the IP PrintWay response timeout value in the printer definitions. For information about setting the response timeout value, see *z/OS Infoprint Server Operation and Administration*.
- Also see the “Tips” for the **max-thread-tasks** attribute.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop the IP PrintWay extended mode daemons (**aopoutd** and **aopwsmd**) and restart Infoprint Server.
- If dynamic configuration is not enabled, specify the equivalent AOPOUTD_MAXTHREADTASKS environment variable in the **aopstart EXEC**.

Default: The limit specified in the **max-thread-tasks** attribute is used. If that attribute is not specified, the limit is 200.

Example: In this example, the limit for the **aopoutd** daemon is 250:

```
aopoutd-max-thread-tasks = 250
```

aopsdbd-max-thread-tasks = nnn

The MAXTHREADTASKS limit for the search database daemon, **aopsdbd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aopsdbd** can have concurrently active.

Tip: See the “Tips” for the **max-thread-tasks** attribute.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop all Infoprint Server daemons and restart Infoprint Server.
- If dynamic configuration is not enabled, specify the equivalent AOPSDBD_MAXTHREADTASKS environment variable in the **aopstart EXEC**.

Default: The limit specified in the **max-thread-tasks** attribute is used. If that attribute is not specified, the limit is 200.

aopssid-max-thread-tasks = nnn

The MAXTHREADTASKS limit for the Infoprint Central daemon, **aopssid**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aopssid** can have concurrently active.

Tip: See the “Tips” for the **max-thread-tasks** attribute.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop **aopssid** and restart Infoprint Server.
- If dynamic configuration is not enabled, specify the equivalent AOPSSID_MAXTHREADTASKS environment variable in the **aopstart EXEC**.

Default: The limit specified in the **max-thread-tasks** attribute is used. If that attribute is not specified, the limit is 200.

aopsubd-max-thread-tasks = nnn

The MAXTHREADTASKS limit for the Print Interface subsystem daemon, **aopsubd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aopsubd** can have concurrently active.

Tip: See the “Tips” for the **max-thread-tasks** attribute.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop **aopsubd** and restart Infoprint Server.
- If dynamic configuration is not enabled, specify the equivalent AOPSUBD_MAXTHREADTASKS environment variable in the **aopstart EXEC**.

Default: The limit specified in the **max-thread-tasks** attribute is used. If that attribute is not specified, the limit is 200.

aopwsmd-max-thread-tasks = nnn

The MAXTHREADTASKS limit for the IP PrintWay extended mode daemon **aopwsmd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aopwsmd** can have concurrently active.

Tip: See the “Tips” for the **max-thread-tasks** attribute.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop the IP PrintWay extended mode daemons (**aopoutd** and **aopwsmd**) and restart Infoprint Server.
- If dynamic configuration is not enabled, specify the equivalent AOPWSMD_MAXTHREADTASKS environment variable in the **aopstart EXEC**.

Default: The limit specified in the **max-thread-tasks** attribute is used. If that attribute is not specified, the limit is 200.

Example: In this example, the limit for the **aopwsmd** daemon is 400:

```
aopwsmd-max-thread-tasks = 400
```

applid = applid

The application program ID that IP PrintWay extended mode uses to establish a VTAM session with printers. This ID must match the name of

the APPL statement defined to VTAM in the SYS1.VTAMLST data set. This attribute is required to use IP PrintWay extended mode to print on VTAM-controlled printers.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent **AOP_APPLID** environment variable in the **aopstart** EXEC.
- This attribute does not apply to IP PrintWay basic mode.

Default: None

Example: applid = PWAPPL

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*.

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
- 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. For information about how to create conversion tables, see information about code set conversion utilities in *z/OS XL C/C++ Programming Guide*.

Infoprint Server uses the code page specified in this field in these ways:

- The Printer Inventory Manager displays this code page in the **Printer code page** field in ISPF panels for new IP PrintWay printer definitions that use the LPR, direct sockets, or IPP protocol to transmit data sets to the remote system.
- Print Interface uses this code page as the default source (document) code page when it converts data streams from ASCII to EBCDIC. A document code page specified in the printer definition or job attribute overrides this value.
- Print Interface uses this code page as the target (printer) code page when it converts PostScript documents that were created with a non-ASCII code page.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: ascii-codepage = IS08859-1

blank-truncation-classes = classes

The list of JES output classes for which IP PrintWay extended mode

removes blanks from the end of records or lines. If a data set is allocated in one of the listed classes, and the data set contains either line data or text data, IP PrintWay removes any blank characters that occur at the end of each record or line. If this attribute is not specified, IP PrintWay extended mode does not remove any blank characters.

Values: One or more letters or numbers. Specify multiple letters and numbers with no separation. An asterisk (*) indicates all JES output classes. Lowercase letters are converted to uppercase.

Tips:

- Specify this attribute only if line or text data prints with extra blank characters at the end of lines.
- This attribute can slow performance and cause IP PrintWay to use more system resources.
- Before giving data to IP PrintWay extended mode, JES always restores blanks that it truncated from data sets to save space on the JES spool. JES restores blanks regardless of how you set the JES2 BLNKTRNC and JES3 TRUNC parameters. Therefore, if you specify either BLNKTRNC=YES (JES2) or TRUNC=YES (JES3), you might need to specify this attribute if your printed output contains unwanted blank characters. For example, if you specify BLNKTRNC=YES for output classes C and D, you could specify AOP_BLANK_TRUNCATION_CLASSES=CD to request that IP PrintWay extended mode remove trailing blank characters for output classes C and D. (BLNKTRNC=YES and TRUNC=YES are the JES defaults.)

For information about specifying JES blank truncation, see one of these:

- *z/OS JES2 Initialization and Tuning Guide*
- *z/OS JES3 Initialization and Tuning Guide*

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOP_BLANK_TRUNCATION_CLASSES environment variable in the **aopstart** EXEC.
- This attribute does not apply to IP PrintWay basic mode.

Default: None.

Example: blank-truncation-classes = ABC123

console-name = *name*

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

Rules:

- The console name must be 2 – 8 alphanumeric or national (#, \$, and @) characters. The first character cannot be numeric. Enclose a name that contains national characters in single or double quotation marks.
- Do not use these reserved names: HC, INSTREAM, INTERNAL, OPERLOG, SYSIOSRS, SYSLOG, UNKNOWN.
- If you specify the Infoprint Central daemon (**aopssid**) in the **start-daemons** attribute, this console name must be unique among all active and inactive console names on all systems in a sysplex. To display the active and inactive console names, enter this MVS command:

DISPLAY EMCS,ST=L

For information about the DISPLAY EMCS command, see *z/OS MVS System Commands*.

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute while the Infoprint Central daemon (**aopssid**) is running, stop **aopssid** and restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

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

Example: console-name = PRINTCON

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*.

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
- 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. For information about how to create conversion tables, see information about code set conversion utilities in *z/OS XL C/C++ Programming Guide*.

Infoprint Server uses this code page in these ways:

- The Printer Inventory Manager displays this code page in the **Printer code page** field in ISPF panels for new PSF printer definitions, general printer definitions, and IP PrintWay printer definitions that use the VTAM or email protocol to transmit data sets to the remote system.
- NetSpool uses this code page as the default source (document) code page when it converts data streams from EBCDIC to ASCII. A document code page in the printer definition or job attribute overrides this value.
- IP PrintWay extended mode uses this code page as the default source (document) code page when it converts data streams from EBCDIC to ASCII. A document code page in the printer definition overrides this value.
- Print Interface uses this code page as the source (document) code page when Print Interface cannot determine the user locale for the **Ip** command.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is

running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server. Also restart the NetSpool started task.

Default: ebcdic-codepage = IBM-1047

hardcopy-messages = all | list | none

Specifies which additional messages Infoprint Server sends to the hardcopy log:

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.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopmsg.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: hardcopy-messages = none

Example: hardcopy-messages = list

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.

Tips:

- You do not need to specify messages that Infoprint Server sends to the console, such as messages with the API prefix, because these messages are automatically sent to the hardcopy log.
- If your installation uses the MPFLSTxx Parmlib member for message automation, you might want to specify the same message IDs in MPFLSTxx.
- If your installation uses an automated operations tool, the tool can process these messages.

Rules:

- If you specify this attribute, also specify **hardcopy-messages = list**.
- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopmsg.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: None.

Example: `hardcopy-message-list={AOP3614I AOP3803E}`

ignore-dcf-routing-errors = yes | no

Indicates whether IP PrintWay extended mode writes an error message when it cannot find a printer definition in the Printer Inventory that matches the specified DEST, CLASS, and FORMS JCL parameters:

- yes** IP PrintWay extended mode does not write an error message (AOP3201E) when it cannot find a printer definition in the Printer Inventory that matches the DEST, CLASS, and FORMS parameters specified on the OUTPUT (or DD) JCL statement for a print job. Also, it does not save information about the print job in the Printer Inventory.
- no** IP PrintWay extended mode writes an error message (AOP3201E) when it cannot find a printer definition that matches the DEST, CLASS, and FORMS parameters. Also, it saves information about the print job in the Printer Inventory so that Infoprint Central can display the print job and the associated AOP3201E message.

Tips:

- Set **ignore-dcf-routing-errors = yes** if your IP PrintWay job-selection rules cause IP PrintWay to select a large number of jobs whose DEST, CLASS, and FORMS parameters do not match printer definitions in the Printer Inventory. (In this situation, message AOP3201E occurs numerous times in the common message log.) Specifying **ignore-dcf-routing-errors = yes** can conserve storage and processing resources.
- If you set **ignore-dcf-routing-errors = yes**, it might be difficult to determine that IP PrintWay did not process a print job because its CLASS, DEST, and FORMS parameters do not match a printer definition.
- The text of message AOP3201E is:
AOP3201E The DEST, CLASS, and FORMS JCL keywords do not match a printer definition in the Printer Inventory.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOP_IGNORE_DCF_ROUTING_ERRORS environment variable in the **aopstart EXEC**.
- This attribute does not apply to IP PrintWay basic mode.

Default: `ignore-dcf-routing-errors = no`

Example: `ignore-dcf-routing-errors = yes`

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.

Rules:

- Make sure that the port is not used by any other service on the same TCP/IP stack.

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: ipp-port-number = 631

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.

The prefix must be two alphanumeric or national (@, \$, and #) 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 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.
- 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.

Tips:

- z/OS also assigns a unique z/OS job ID to each output data set. The z/OS job ID might be different from the job ID of the output data set on the JES spool. JES operator commands use this z/OS job ID.
- The job ID that Print Interface returns to the job submitter does not contain this prefix.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: job-prefix = PS

Example: job-prefix = 'P#'

log-retention = *days*

The number of days worth of messages that Infoprint Central displays for print jobs and printers. Also, the number of days for which Infoprint Central displays print jobs that have been removed from the JES spool.

If you do not use the MVS system logger for the common message log, this attribute also determines how many days worth of messages Infoprint

Server retains in the z/OS UNIX file system. If you use the system logger for messages, the retention period specified in the system logger log stream definition is used instead.

You can specify a value from 0 – 59. A value of 0 means that Infoprint Central cannot display any messages and cannot display information about print jobs that have been removed from the JES spool. In addition, if you do not use the system logger for messages, Infoprint Server does not write any messages in the z/OS UNIX file system.

Guidelines:

- If you do not use the system logger for messages, specify a value greater than 0 or use the default of 1 day, so that Infoprint Server writes messages in a z/OS UNIX file system.
- If you use the system logger for messages and also use Infoprint Central, the value should be less than or equal to the retention period specified in the log stream definition. If you specify a higher value than the retention period for the log stream, you might not be able to see the messages for all the historical print jobs that Infoprint Central displays.
- If you use the system logger but do not use Infoprint Central, you can specify 0 to save space in the z/OS UNIX file system.
- If you specify a value greater than 0 or use the default of 1 day, you might need to increase the amount of space available to the file system mounted at the `/var` or `/var/Printsrv` mount point because the common message log and Historical Inventory can contain a large amount of data. See “Calculating DASD space requirements for the `/var/Printsrv` directory” on page 32.
- Because the common message log and Historical Inventory can contain a large amount of data, start with a value of 1 day. Increase the value by 1 day if there is enough space in the file system and if users request to see more messages or historical information.
- If the size of the common message log in the z/OS UNIX file system grows too large, you can decrease its size. For more information, see “Reducing the size of the common message log in the z/OS UNIX file system” on page 111.
- If you use Infoprint Central to view messages, set the TZ and either the LC_TIME or LC_ALL environment variables. These variables affect the date and time displayed in messages. See “Setting environment variables for the Printer Inventory Manager” on page 43.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the `aopd.conf` file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: `log-retention = 1`

Example: `log-retention = 3`

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. 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.

Tips:

- Assign port 515 to the Print Interface LPD so that you can use all of the functions that Infoprint Server provides. These limitations apply if you assign a port other than port 515:
 - 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 Infoprint Port Monitor for Windows, however, is an exception. You can configure the Port Monitor to send print requests to any port.
 - IP PrintWay basic mode cannot transform data with the **resubmit for filtering** option. This is because IP PrintWay resubmits data 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.

Rules:

- Make sure that the port is not used by any other service on the same TCP/IP stack.
- This port number must not be reserved in the *hlq.PROFILE.TCPIP* data set. For more information about the PORT statement, see “Editing the *hlq.PROFILE.TCPIP* data set” on page 117.
- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: lpd-port-number = 515

logstream-name = *name*

The name of the log stream that is defined to the MVS system logger for Infoprint Server messages. If this attribute is not specified, Infoprint Server does not use the system logger and instead writes messages in the Infoprint Server base directory (default is */var/Printsrv*).

You can specify 1-26 characters. A valid log stream name consists of one or more qualifiers. Each qualifier can contain 1-8 letters, numbers, and national characters (\$, #, and @). The first character of each qualifier cannot be a number. Each qualifier must be separated by a period. Lowercase letters are converted to uppercase.

Rule: You can specify this attribute only in the system configuration definition. If you change the value, you do not need to restart Infoprint Server.

Default: None.

Example: logstream-name = AOP.MSG

mail-do-not-add-suffixes = yes | no

Indicates whether IP PrintWay extended mode appends a suffix to file names specified in the MAILFILE JCL parameter or the **mail-file-name** job attribute:

yes IP PrintWay does not append a suffix.

no IP PrintWay appends one of these suffixes for the data formats: afp, jpg, pcl, pdf, ps, sap, tif, txt, or octet-stream (for unrecognized data formats).

Tip: If **mail-do-not-add-suffixes = yes**, IP PrintWay ignores the **mail-preserve-suffixes** attribute.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOP_MAIL_DO_NOT_ADD_SUFFIX environment variable in the **aopstart EXEC**.
- If you run IP PrintWay basic mode, specify the equivalent AOP_MAIL_DO_NOT_ADD_SUFFIX environment variable in the IP PrintWay startup procedure.

Default: mail-do-not-add-suffixes = no

Example: mail-do-not-add-suffixes = yes

mail-preserve-suffixes = suffixes

The suffixes that IP PrintWay extended mode preserves in file names specified in the MAILFILE JCL parameter or in the **mail-file-name** job attribute. If the file name ends with one of the suffixes in this list, IP PrintWay does not append another suffix to the file name.

If no file name is specified in MAILFILE or **mail-file-name**, or if the suffix of the file name does not match a suffix in this attribute, IP PrintWay appends one of these suffixes for the data formats it recognizes: afp, jpg, pcl, pdf, ps, sap, tif, txt, or octet-stream (for unrecognized data formats).

Tips:

- Specify the suffixes that users might specify in file names. For example, you might specify the .doc suffix for Microsoft Word documents.
- If the **mail-do-not-add-suffixes = yes**, IP PrintWay ignores the **mail-preserve-suffixes** attribute.

Rules:

- Begin each suffix with a period.
- To specify more than one suffix, separate the suffixes with a blank and enclose the suffixes in single or double quotation marks.
- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOP_MAIL_PRESERVE_SUFFIXES environment variable in the **aopstart EXEC**.

- If you run IP PrintWay basic mode, specify the equivalent AOP_MAIL_PRESERVE_SUFFIXES environment variable in the IP PrintWay startup procedure.

Default: None.

Example: mail-preserve-suffixes = ".123 .xls .doc"

mail-use-first-address = yes | no

When the print job contains more than one document, indicates whether IP PrintWay extended mode sends each document to the email address or addresses specified for the first document in the print job.

yes IP PrintWay sends each document to the email address or addresses specified for the first document in the print job.

no IP PrintWay sends each document to the email address or addresses specified for that document.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOP_MAIL_USE_FIRST_ADDRESS environment variable in the **aopstart** EXEC.
- This attribute does not apply to IP PrintWay basic mode.

Tip: This attribute has no effect if the **Concatenate job** option is selected in the **Dataset grouping** field in the printer definition, or if the **Automatic dataset grouping** field is selected in the printer definition. When either concatenation option is selected, IP PrintWay extended mode sends all documents in a print job in one email to the email address or addresses specified for the first document in the print job.

Default: mail-use-first-address = yes

Example: mail-use-first-address = no

mailer-options = options

Options for the z/OS UNIX **sendmail** command. IP PrintWay extended mode specifies these options in addition to the **-i** option when it sends emails. You can specify this attribute regardless of whether you specify the **mailer-path-name** attribute. This attribute is optional.

If you specify more than one option, separate the options with a blank and enclose the entire value in single or double quotation marks.

Rules:

- To specify more than one option, separate the options with a blank and enclose the options in single or double quotation marks.
- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOPMAILER_OPTIONS environment variable in the **aopstart** EXEC.
- If you run IP PrintWay basic mode, specify the equivalent AOPMAILER_OPTIONS environment variable in the IP PrintWay startup procedure.

Default: None

Example: mailer-options = -Am

mailer-path-name = *pathname*

The full path name of the z/OS UNIX **sendmail** command that IP PrintWay extended mode uses to send output to email destinations. This attribute is optional. However, you must specify it if you did not install the **sendmail** executable file in the default directory, **/bin**.

Rules:

- Start the path name with a slash (/).
- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOPMAILER environment variable in the **aopstart** EXEC.
- If you run IP PrintWay basic mode, specify the equivalent AOPMAILER environment variable in the IP PrintWay startup procedure.

Default: mailer-path-name=/bin/sendmail

Example: mailer-path-name=/mydirectory/sendmail

max-historical-inventory-size = *megabytes*

The maximum size in megabytes (MBs) of the Historical Inventory, **/var/Printsrv/hinv**. The Historical Inventory contains information about print jobs and documents (output data sets) 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. However, the number of days specified in the **log-retention** attribute limits the amount of information that Infoprint Server keeps in the Historical Inventory.

Table 57 shows the number of documents one MB in the Historical Inventory can contain.

Table 57. Number of documents in the Historical Inventory

If print jobs on the JES spool contain:	One MB in the Historical Inventory contains:
1 document in each print job	1350 documents
3 documents in each print job	1750 documents
5 documents in each print job	1850 documents
10 documents in each print job	1950 documents

Guideline: Use this formula to estimate the maximum size of the Historical Inventory:

Maximum size of Historical Inventory =
(Documents processed in one day X Days to retain information) /
Documents in one MB

For example:

- Documents processed in one day = 100,000
- Days to retain information = 3 (**log-retention** = 3)

- Documents in one MB = 1350 documents

The maximum size of the Historical Inventory = (100,000 X 3) / 1350 = 223 MB (rounded up)

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: max-historical-inventory-size = 10

Example: max-historical-inventory-size = 300

max-thread-tasks = nnn

The MAXTHREADTASKS limit for these Infoprint Server daemons: **aophinvd, aopippdxp, aoplpd, aopnetd, aopoutd, aopsdbd, aopssid, aopsubd, and aopwsmd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that a single process (daemon) can have concurrently active.

If you set a MAXTHREADTASKS limit for a specific daemon in another attribute, the limit in this attribute does not apply for that daemon.

Tips:

- The default MAXTHREADTASKS limits that Infoprint Server uses are suitable for most installations. Consider setting a different limit only if you receive this message:

```
BPXI040I PROCESS LIMIT MAXTHREADTASKS HAS REACHED nn% OF ITS
CURRENT CAPACITY OF nnn FOR PID=nnnnnn IN JOB daemon
```

The BPXI040I message is an informational message and does not indicate that an error occurred. If the BPXI040I message indicates that less than 100% of capacity has been reached, threads run without delay. However, when the limit is reached, a newly created thread does not run until an existing thread ends.

The LIMMSG statement in the BPXPRMxx member of SYS1.PARMLIB controls whether z/OS UNIX issues message BPXI040I. The default is that z/OS UNIX does not issue BPXI040I messages. You can change the LIMMSG value dynamically using the SETOMVS command.

- Do not set a value less than 32.
- If you increase the MAXTHREADTASKS limit, you might need to increase the maximum region size because each concurrent task requires additional storage:
 - If you use the AOPSTART JCL procedure to start Infoprint Server, increase the region size on the EXEC statement in the procedure.
 - If you use the **aopstart** command to start Infoprint Server, increase the maximum address space in the MAXASSIZE parameter in the BPXPRMxx member of SYS1.PARMLIB or in the ASSIZEMAX RACF parameter for the user ID that starts Infoprint Server (ASSIZEMAX overrides MAXASSIZE).

- If you enter the **aopstart** command from the TSO/E OMVS command line, the SIZE parameter on the TSO/E LOGON panel determines the maximum region size of the address space.
- Do not set a very high limit because each concurrent task requires some storage below the 16 MB line. If the limit is too high, an abend can occur.
- If you exceed the default limit (200) when a lot of printers are waiting for operator intervention at the same time, you might need to increase the limit in the **aopoutd-max-thread-tasks** attribute or reduce the IP PrintWay response timeout value in the printer definition. For information about setting the response timeout value, see *z/OS Infoprint Server Operation and Administration*.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop all Infoprint Server daemons and restart Infoprint Server.
- If dynamic configuration is not enabled, specify this equivalent AOP_MAXTHREADTASKS environment variable in the **aopstart EXEC**.

Default: The limit is 200 for each daemon.

Example: In this example, the limit for the **aopoutd** daemon is 250, while the limit for all other daemons is 225:

```
aopoutd-max-thread-tasks = 250
max-thread-tasks = 225
```

netspool-use-fixed-jobid = yes | no

Indicates whether NetSpool assigns the same job ID to all output data sets that it creates on the JES spool.

- yes** NetSpool assigns the same job ID to all output data sets. The job ID is the value specified in the **job-prefix** attribute, which defaults to PS, followed by 000000. For example, PS000000
- no** NetSpool assigns a unique job ID to each data set.

When Infoprint Central displays Infoprint Server print jobs, it displays the unique job ID that Infoprint Server assigns to each print job. Therefore, if **netspool-use-fixed-jobid = yes**, Infoprint Central functions that need to correlate JES print jobs with Infoprint Server print jobs do not work.

Tip: Specify **netspool-use-fixed-jobid = yes** only if you run JES3 and have a very large number of PSF FSAs or IP PrintWay basic mode FSAs. Contact IBM for more information.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop **aopnetd** and restart Infoprint Server.
- If dynamic configuration is not enabled, specify the equivalent AOPNETD_USE_FIXED_JOBID environment variable in the **aopstart EXEC**.

Default: netspool-use-fixed-jobid = no

Example: netspool-use-fixed-jobid = no

netspool-use-unaltered-jobid = yes | no

Indicates whether NetSpool assigns a job ID to output data sets that it creates on the JES spool.

yes NetSpool does not assign a job ID to the output data sets. The job ID for each data set is the same as the job ID of the **aopnetd** daemon.

no NetSpool assigns a job ID to each data set.

When Infoprint Central displays Infoprint Server print jobs, it displays the unique job ID that Infoprint Server assigns to each print job. Therefore, if **netspool-use-unaltered-jobid = yes**, Infoprint Central functions that need to correlate JES print jobs with Infoprint Server print jobs do not work.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop **aopnetd** and restart Infoprint Server.
- If dynamic configuration is not enabled, specify the equivalent **AOPNETD_USE_UNALTERED_JOBID** environment variable in the **aopstart EXEC**.

Default: netspool-use-unaltered-jobid = no

Example: netspool-use-unaltered-jobid = no

select-work-from-hold-queue = yes | no

Indicates whether IP PrintWay extended mode selects new held output groups from the JES spool.

yes IP PrintWay extended mode selects new held output groups from the JES spool. This lets you use Infoprint Central to move the held output groups to different IP PrintWay printers before releasing them.

no IP PrintWay extended mode does not select new held output groups from the JES spool. This lets you use JES or SDSF commands to release individual data sets in an output group for printing.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute while IP PrintWay is running, stop the IP PrintWay daemons (**aopoutd** and **aopwsmd**) and restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent **AOP_SELECT_WORK_FROM_HOLD_QUEUE** environment variable in the **aopstart EXEC**.
- This attribute does not apply to IP PrintWay basic mode.

Default: select-work-from-hold-queue = yes

Example: select-work-from-hold-queue = no

smf-recording = yes | no

Indicates whether IP PrintWay extended mode writes SMF type 6 records for data sets that it sends to a printer or email destination. SMF type 6 records contain accounting information.

yes IP PrintWay extended mode writes SMF type 6 records.

no IP PrintWay extended mode does not write SMF type 6 records.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.
- This attribute does not apply to IP PrintWay basic mode.

Default: smf-recording = yes

Example: smf-recording = no

snmp-community = name | public

The name assigned to the SNMP community for making SNMP requests to the z/OS system. This name must match both of these:

- The community name provided to the z/OS SNMP agent
- The community name defined to the SNMP manager (Network Printer Manager)

If you do not provide a community name to the z/OS SNMP agent, the z/OS SNMP agent accepts requests from any SNMP manager with the community name of **public**. Specify 1–32 uppercase or lowercase letters, numbers, or special characters. The community name is case-sensitive. Note that the z/OS agent and the SNMP manager might have other restrictions on the length of the community name or the allowed characters.

Rules:

- If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute while the SNMP subagent (**aopsnmpd**) is running, stop the subagent and restart Infoprint Server.
- If dynamic configuration is not enabled, specify this attribute in the **aopd.conf** file. If you change this attribute while Infoprint Server is running, stop all Infoprint Server daemons and restart Infoprint Server. Be sure to wait for all daemons to stop before restarting Infoprint Server.

Default: snmp-community = public

suppress-post-unix-filter-formatting = yes | no

Indicates whether IP PrintWay extended mode formats line or text data after a UNIX filter processes it:

- yes** IP PrintWay extended mode does not format data when all these conditions are met:
- The data set to be processed contains either line data or text data.
 - A UNIX filter is specified in the printer definition for the data format.

- The PAGEDEF or FORMDEF parameter is specified on the OUTPUT JCL statement for the data set. (A page or form definition specified in the printer definition does not meet this criterion.)

no IP PrintWay extended mode always formats line or text data after a UNIX filter processes it. For example, IP PrintWay processes carriage control characters and transparent data characters, and converts the data from the document code page to the printer code page.

Tips:

- Specify **suppress-post-unix-filter-formatting=yes** if you use a UNIX filter that calls an AFP transform to convert line data to PCL, PDF, or PostScript format.
- This attribute does not affect how IP PrintWay extended mode processes data when you specify a DLL filter in the printer definition. IP PrintWay extended mode does not modify data returned by DLL filters.

Rules:

- If you run IP PrintWay extended mode and dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, you do not need to restart Infoprint Server.
- If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOP_SUPPRESS_POST_UNIX_FILTER_FORMATTING environment variable in the **aopstart** EXEC.
- This attribute does not apply to IP PrintWay basic mode.

Default: suppress-post-unix-filter-formatting = no

Example: suppress-post-unix-filter-formatting = yes

Related information: For information about:

- Fields on the Infoprint Server ISPF panel that you can use to specify attributes, see “Dynamic attributes” on page 69.
- How to edit the system configuration definition, see “Steps for enabling dynamic configuration” on page 72.
- How to use PIDU, see *z/OS Infoprint Server Operation and Administration*.

Appendix E. AOPBATCH program

This appendix describes the AOPBATCH utility program that Infoprint Server provides. AOPBATCH lets you use MVS job control language (JCL) to run a program that resides in a z/OS UNIX file.

IBM suggests that you use AOPBATCH instead of BPXBATCH to run programs that Infoprint Server provides because AOPBATCH sets default values for the PATH, LIBPATH, and NLSPATH environment variables that are suitable for installations that installed Infoprint Server files in default locations. Also, AOPBATCH lets **stdin** be read from a DD statement and lets **stdout** and **stderr** be written to a DD statement.

Format

JCL:

```
EXEC PGM=AOPBATCH,PARM='/program_name options'
```

Description

AOPBATCH lets you submit an MVS batch job to run an executable program that resides in a z/OS UNIX file.

AOPBATCH uses the **spawn()** function to create a process that runs in the same address space as AOPBATCH under these conditions:

- The set-user-ID and set-group-ID bits are not set for the program.
- Enough storage exists to spawn a process in the same address space.
- The share address space extended attribute is set for the program object. This attribute is set by default.

If these conditions are not met, AOPBATCH spawns the process in a separate address space.

With AOPBATCH you can specify these optional DD statements to set up the UNIX environment and to allocate **stderr**, **stdin**, and **stdout** files.

Note: The STDIN, STDOUT, and STDERR statements are suitable only for text data. Do not use STDIN, STDOUT, and STDERR to specify an MVS data set that contains non-text data. For example, do not specify an MVS data set in STDIN or STDOUT when you run a transform command because transform input and transformed output is not text data.

STDENV

Specifies environment variables for use by program *program_name*. You can specify the environment variables in-stream in the JCL, in a UNIX file, or in an MVS data set. Specify the environment variables in the format *variable=value*, with one environment variable per line or record.

If you omit the STDENV data definition (DD) or do not specify one of these 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/En_US/%N:/usr/lib/nls/msg/%L/%N

AOPBATCH also sets the HOME environment variable to the user's home directory, the LOGIN variable to the user ID, and the _BPX_SHAREAS variable.

Tip: The **lp** command and some transform commands return MVS return codes (0, 4, 8) instead of UNIX exit values (0, 1) when you set the AOP_MVS_RETURN_CODES=YES environment variable.

STDERR

Specifies a SYSOUT data set, UNIX file, or MVS data set.

STDIN

Specifies input to the program. You can specify input in-stream in JCL, a UNIX file, or an MVS data set.

STDOUT

Specifies a SYSOUT data set, UNIX file, or MVS data set.

Parameters

AOPBATCH accepts the name of the executable program file you want to run followed by options for that program.

- / 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++ runtime options. You *must* include the initial slash if any of the PARM data itself includes a slash. For example, if the transform name is /mylib/ps2afp, specify: PARM='/mylib/ps2afp ...'.

program_name

The name of an executable program file that resides in a z/OS UNIX file. The program name is case-sensitive. The file must reside in one of the directories specified in the PATH environment variable or you must specify the path. Use the STDENV data definition (DD) to set the PATH environment variable if the default value set by AOPBATCH is not suitable.

You can run these programs that Infoprint Server provides:

- **aopmig:** Printer Inventory migration program for PSF
- **aoplogu:** Writes messages in the common message log to a file
- **remotexf:** Sends data to InfoPrint Transform Manager for Linux
- **pidu:** Manages the Printer Inventory

You can also run transform programs. For example: **afpxpcl**, **afpxpdf**, **afpxps**, **pcl2afp**, **pdf2afp**, **ps2afp**.

options

Options accepted by program *program_name*. For a description of the allowed options, see the documentation for the program. For Infoprint Server programs, see one of these:

- For the **aoplogu** and **pidu** programs, see *z/OS Infoprint Server Operation and Administration*.
- For the **remotexf** program, see *z/OS Infoprint Server User's Guide*.
- For Infoprint transform programs, see the documentation for the transform.

Usage notes

- Sequence numbers in columns 73 - 80 in data specified with the STDIN DD statement are treated as part of the data. Therefore, if you use the ISPF editor to edit the data, consider turning sequence numbers off by typing NUMBER OFF on the command line before you type any data. If sequence numbers already exist, type UNNUM to remove them and set number mode off.
- Sequence numbers in columns 73 - 80 in data specified with the STDENV data definition (DD) are ignored and not treated as part of the data.
- AOPBATCH is installed in SYS1.LINKLIB. Therefore, you do not need to include a STEPLIB DD statement in your JCL.
- If you did not add the Language Environment runtime libraries (*hlq.SCEERUN* and *hlq.CEERUN2*) and the C++ runtime library (*hlq.SCLBDLL*) to the system LNKLST, concatenate them to a STEPLIB DD statement.
- You cannot use AOPBATCH to run a shell command or a shell script directly. To run a shell command or script, specify **sh** as the program name, for example: PARM='sh -c myscript' or PARM='/sh -c /mylib/myscript.
- When you run a 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 when you use AOPBATCH, the transform cannot read input from standard input and cannot write its output to standard output. To identify the transform input and transform output data set or file, you can specify either a DD statement name or a data set name. You must specify a DD statement name if you want the transform to write output to an MVS data set that does not already exist, and you must include the DD statement to allocate the output data set.
- When you specify the name of a DD statement in the PARM parameter, use this format:

//DD:DDname

The name of the DD statement is *DDname*.

- When you specify an MVS data set name in the PARM parameter, 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), C/C++ appends the TSO user prefix.
 - If you are running under MVS batch or IMS (batch or online), C/C++ appends the RACF user ID.
- If you use AOPBATCH to run the **lp** command and the printer definition specifies an installation-written UNIX filter, specify the **_BPX_SHAREAS=YES** environment variable. In all other cases, do not specify the **_BPX_SHAREAS** environment variable because AOPBATCH sets it appropriately.

If you do not set **_BPX_SHAREAS=YES** and Infoprint Server calls a UNIX filter, you can receive this error message:

```
lp: AOP003E cat: EDC5157I An internal error has occurred.
```

Exit values

AOPBATCH returns one of these values:

Value Meaning

- | | |
|-----------|--|
| 1 | An error occurred while AOPBATCH was trying to write to the STDOUT DD. |
| 2 | An error occurred while AOPBATCH was trying to write to the STDERR DD. |
| 4 | AOPBATCH could not spawn the process. |
| <i>nn</i> | The exit value from the spawned process. |

Some transform commands can return either MVS return codes (0, 4, 8) or UNIX exit values (0, 1) depending on the value of the AOP_MVS_RETURN_CODES environment variable. If the environment variable is not set, the transforms return UNIX exit values.

Examples -- AOPBATCH

These examples show how to use the AOPBATCH program. The environment variables shown in the first example are set to default values. If you installed and created Infoprint Server files in default directories, you do not need to specify these environment variables, and you can omit the STDENV data definition (DD).

1. This example shows how to use the AOPBATCH program to run the Infoprint Server **pidu** program.

```
//AOPBATCH JOB ...
//PIDU EXEC PGM=AOPBATCH,PARM='/pidu'
//STDIN DD *
list printer;
/*
//STDOUT DD SYSOUT=*
//STDERR DD SYSOUT=*
//STDENV DD *
PATH=/usr/lpp/Printsrv/bin:/bin:/usr/bin
LIBPATH=/usr/lpp/Printsrv/lib:/lib:/usr/lib
NLSPATH=/usr/lpp/Printsrv/En_US/%N:/usr/lib/nls/msg/%L/%N
/*
```

2. This example shows how to use the AOPBATCH program to run the Infoprint Server **ps2afp** command. In this example, input to the command is in file **/tmp/input.ps**, and output is written to MVS data set HLQ.OUTPUT.AFP.

```
//AOPBATCH JOB ...
//PRINT EXEC PGM=AOPBATCH,PARM='/ps2afp -o //DD:OUTPUT /tmp/input.ps'
//OUTPUT DD DSN=HLQ.OUTPUT.AFP,DISP=SHR
//STDERR DD SYSOUT=*
```

Tip: For more examples that show how to use AOPBATCH, see *z/OS Infoprint Server User's Guide*.

Sample jobs

IBM provides these sample jobs in SYS1.SAMPLIB, which use AOPBATCH:

- AOPPIDU, which runs the Printer Inventory Definition Utility, **pidu**
- AOPMIGRX, which runs the migration program for PSF, **aopmig**
- AOPLOGUS, which runs the **aoplogu** command

Appendix F. Application programming interface (API)

Infoprint Server provides an Application Programming Interface (API) that lets you code C or C++ applications to do these functions:

- Create an Infoprint Server print job on the z/OS Job Entry System (JES) spool.
- Cancel an Infoprint Server print job.
- Query the status of an Infoprint Server print job.
- Query the status of all Infoprint Server print jobs for a printer.
- Query information about all printers defined in the Infoprint Server Printer Inventory.
- Connect to a transform daemon.

This appendix describes these topics for application programmers who want to use the Infoprint Server API:

- “API files”
- “API data structures”
- “API functions” on page 462
- “Programming considerations” on page 476
- “Customization requirements” on page 478

API files

This table summarizes the files provided with the API. For each file, the table lists the default directory into which the file is installed.

File	Directory	Description
aopapi.dll	/usr/lpp/Printsrv/lib	Executable code
aopapi.h	/usr/lpp/Printsrv/samples	Include file containing data structures and function declarations
libaopapi.a	/usr/lpp/Printsrv/lib	Static library containing function stubs

API data structures

This table summarizes the data structures provided with the Infoprint Server API. After the table, each data structure is described more fully in alphabetical order.

Data structure	Include file	Description
DocumentInfo	aopapi.h	Information about a document in an Infoprint Server print job
JobInfo	aopapi.h	Information about an Infoprint Server print job and each document in the job
PrinterInfo	aopapi.h	Information about an Infoprint Server printer
JobToken	aopapi.h	A token that uniquely identifies an Infoprint Server print job

Data structure	Include file	Description
QueryToken	aopapi.h	A token that uniquely identifies an Infoprint Server query request
Transform	aopapi.h	Information about a connection between your transform program and a transform daemon.

DocumentInfo - information about a document in a job

The DocumentInfo data structure contains information about a document. It is included in the JobInfo data structure. The status and format fields are described by the DocumentStatus and DocumentFormat enumerations.

```
typedef struct
{
    char          status;          /* Current document status      */
    char          format;         /* Format of the document        */
    char          reserved[2];
    unsigned int  size;           /* low-order word of size in
                                octets of document on spool */
    char*         name;           /* Filename of document         */
    unsigned int  high_size;      /* high-order word of size in
                                octets of document on spool */
} DocumentInfo;
```

The size of a document is stored in two fields: size and high_size. You can calculate the total size and store it in field "total" as follows:

```
unsigned long long total = (high_size << 31) + size;
```

The high_size field is available only in API Version 1 and later versions. You can use the GetAPIVersionNumber function to obtain the API version number. For information, see "Determining the API version number" on page 462.

The DocumentStatus enumeration identifies the status of the print job on the JES spool. For an explanation of the job states in this data structure, see the description of the lpstat command in the *z/OS Infoprint Server User's Guide*. The job states in the DocumentStatus structure are the same as those returned by the lpstat command. (The selected state in DocumentStatus is the same as the processing state of the lpstat command.)

```
typedef enum
{
    status_completed = 0,
    status_pending   = 1,
    status_selected  = 2,
    status_held      = 3,
    status_failed    = 4,
    status_purged    = 5
} DocumentStatus;
```

The DocumentFormat enumeration identifies the format of the input data stream. It is included in the DocumentInfo data structure. Infoprint Server determines the format automatically unless the **document-format** job attribute was specified as an attribute to the CreatePrintFile function. For an explanation of the data formats in this data structure, see the description of the **document-format** job attribute in the *z/OS Infoprint Server User's Guide*.

```
typedef enum
{
    format_line      = 0,
    format_modca     = 1,
}
```

```

    format_other      = 2,
    format_pcl       = 3,
    format_postscript = 4,
    format_text      = 5,
    format_pdf       = 6,
    format_sap       = 7
} DocumentFormat;

```

JobInfo - information about an Infoprint Server print job and each document in the job

This data structure contains information about an Infoprint Server print job and each document in the print job. It is returned by the `GetJobInfo` and `EnumJobs` functions. The `documents` field is described by the `DocumentInfo` data structure.

```

typedef struct
{
    unsigned int  id;           /* Infoprint Server job number */
    char*        printer_name; /* The printer on which the job was
                               /* submitted.
    char*        user_name;    /* The owner of the job
    char*        name_text;    /* The value of the name-text attribute
    int          priority;     /* Priority at which job was submitted
    time_t       submitted;    /* The time the job was submitted
                               /* in seconds since epoch.
    unsigned int  ndocuments;  /* The number of documents contained
                               /* in the job.
    DocumentInfo** documents; /* One DocumentInfo structure for each
                               /* document in the job.
} JobInfo;

```

JobToken - identifies an Infoprint Server print job

This opaque data structure uniquely identifies a request to create an output data set on the JES spool. The `CreatePrintFile` function returns the `JobToken` data structure when it receives a request to create an output data set. It is then input to the `WritePrintFile`, `ClosePrintFile`, and `AbortPrintFile` functions. `JobToken` is no longer valid after the `ClosePrintFile` function is called to close the output data set or the `AbortPrintFile` function is called to terminate the request to create an output data set.

```

typedef struct { void* p[2]; } JobToken;

```

PrinterInfo - information about an Infoprint Server printer

This data structure contains information about an Infoprint Server printer that is described by a printer definition in the Infoprint Server Printer Inventory. It is returned by the `GetPrinterInfo` and `EnumPrinters` functions. The fields in this structure are:

name	The name of the printer definition
description	The description specified in the printer definition or a NULL value if no description is specified
location	The location specified in the printer definition or a NULL value if no location is specified
njobs	Number of Infoprint Server jobs for this printer definition that are in the pending, selected, and held states
status	Described by the <code>PrinterStatus</code> enumeration

```

typedef struct
{
    char*        name;           /* Printer definition name */
}

```

```

    char*      description; /* Printer description or NULL if not supplied */
    char*      location;   /* Printer location or NULL if not supplied */
    unsigned   njobs;      /* Number of jobs queued on this printer */
    char       status;     /* Current printer status */
} PrinterInfo;

```

The PrinterStatus enumeration identifies the status of the printer. It is included in the PrinterInfo data structure. Currently, the status of all printers is enabled.

```

typedef enum
{
    status_enabled   = 0, /* The printer is accepting new jobs */
    status_disabled  = 1, /* The printer is not accepting new jobs */
} PrinterStatus;

```

QueryToken - identifies an Infoprint Server query request

This opaque data structure uniquely identifies a query request for an enumeration of print jobs for a printer. It is returned by the BeginEnumJobs function. It is input to the EnumJobs and EndEnumJobs functions. It is no longer valid after your application calls the EndEnumJobs function to end the enumeration.

```

typedef struct { void* p[2]; } QueryToken;

```

Transform - information about a transform connection

This data structure describes a connection between your transform program and a transform daemon. It is returned by the RequestTransform function, and is input to the ReleaseTransform function.

```

typedef struct {
    const char* class_name; /* name of transformer class (63 bytes maximum) */
    const int   fd;         /* connection to transform daemon */
} Transform;

```

The fields in this structure are:

class_name The name of the transform and class in the format: *transformname[_transformclass]*. This value must match the name of a transform entry in the transform configuration file (**aopxfd.conf**). The format of *class_name* is: *transformname[_transformclass]*.

fd A UNIX file descriptor used to send input and receive output from the transform daemon.

API functions

These tables summarize the C library functions provided with the API. Each function is listed according to the type of function it does. After the tables, each function is described more fully in alphabetical order.

Determining the API version number

Function	Include file	Description
GetAPIVersionNumber	aopapi.h	Returns the version number of this API

Initializing and terminating the API

Function	Include file	Description
InitAPI	aopapi.h	Initializes the API

Function	Include file	Description
TermAPI	aopapi.h	Terminates the API

Creating a print job

Function	Include file	Description
CreatePrintFile	aopapi.h	Creates an Infoprint Server print job
WritePrintFile	aopapi.h	Writes data to an Infoprint Server print job
ClosePrintFile	aopapi.h	Closes an Infoprint Server print job
AbortPrintFile	aopapi.h	Terminates an Infoprint Server print job

Canceling a print job

Function	Include file	Description
CancelJob	aopapi.h	Cancels an Infoprint Server print job

Querying print jobs

Function	Include file	Description
GetJobInfo	aopapi.h	Returns status of an Infoprint Server print job
BeginEnumJobs	aopapi.h	Begins an enumeration of Infoprint Server print jobs for a printer
EnumJobs	aopapi.h	Returns an enumeration of Infoprint Server print jobs for a printer
EndEnumJobs	aopapi.h	Ends an enumeration of Infoprint Server print jobs for a printer

Querying printers

Function	Include file	Description
GetPrinterInfo	aopapi.h	Returns information about an Infoprint Server printer
BeginEnumPrinters	aopapi.h	Begins an enumeration of Infoprint Server printers
EnumPrinters	aopapi.h	Returns an enumeration of Infoprint Server printers
EndEnumPrinters	aopapi.h	Ends an enumeration of Infoprint Server printers

Connecting to the Infoprint Server Transform Manager

Function	Include file	Description
RequestTransform	aopapi.h	Establishes a connection with a transform daemon

Function	Include file	Description
ReleaseTransform	aopapi.h	Releases a connection with a transform daemon

Handling errors

Function	Include file	Description
SetTerminationHandler	aopapi.h	Registers a function to be called at program termination
ErrorString	aopapi.h	Returns a pointer to an API error message
ErrorNumber	aopapi.h	Returns the API error code
BufferSizeRequired	aopapi.h	Returns the required buffer size after a buffer overflow error

AbortPrintFile -- terminate an Infoprint Server print job

Format

```
#include <aopapi.h>
int AbortPrintFile( JobToken token );
```

Description

The AbortPrintFile function removes an output data set from the JES spool that has not yet been closed by the ClosePrintFile function. The *token* argument must be a valid job token returned by the CreatePrintFile function. After you call the AbortPrintFile function, this job token is no longer valid and cannot be used by any other functions.

Returned value

AbortPrintFile returns:

- Zero if the function was successful.
- A non-zero value if the job token is not valid or if the function was not successful for another reason. You can use the ErrorNumber function to obtain an error code, and you can use the ErrorString function to obtain a description of the error.

BeginEnumJobs -- begin an enumeration of Infoprint Server print jobs for a printer

Format

```
#include <aopapi.h>
int BeginEnumJobs( QueryToken* out_token,
                  const char* printer_name );
```

Description

The BeginEnumJobs function begins an enumeration of all Infoprint Server print jobs for a printer. After calling BeginEnumJobs, call these functions:

1. EnumJobs to obtain the enumeration of print jobs.
2. EndEnumJobs to end the enumeration.

Input parameters are:

<i>printer_name</i>	The name of a printer definition in the Infoprint Server Printer Inventory whose jobs you want to enumerate. Printer definition names are case-sensitive.
<i>out_token</i>	A query token that is returned by this function. You must pass this token as an argument to the EnumJobs and EndEnumJobs functions.

Returned value

BeginEnumJobs returns:

- Zero and sets *out_token* if the function was successful.
- A non-zero value if the function was not successful. You can use the ErrorNumber function to obtain an error code, and you can use the ErrorString function to obtain a description of the error.

BeginEnumPrinters -- begin an enumeration of Infoprint Server printers

Format

```
#include <aopapi.h>
int BeginEnumPrinters( QueryToken* out_token );
```

Description

The BeginEnumPrinters function begins an enumeration of all printers defined by printer definitions in the Infoprint Server Printer Inventory. After calling BeginEnumPrinters, call these functions:

1. EnumPrinters to obtain the enumeration of printers.
2. EndEnumPrinters to end the enumeration.

Argument *out_token* is a query token returned by this function. You must pass this token as an argument to the EnumPrinters and EndEnumPrinters functions.

Returned value

BeginEnumPrinters returns:

- Zero and sets *out_token* if the function was successful.
- A non-zero value if the function was not successful. You can use the ErrorNumber function to obtain an error code, and you can use the ErrorString function to obtain a description of the error.

BufferSizeRequired -- return the required size of the buffer after a buffer overflow error

Format

```
#include <aopapi.h>
unsigned int BufferSizeRequired();
```

Description

The BufferSizeRequired function returns the buffer size that is required to complete the last action that failed due to a buffer overflow on the current thread. Call this function when the system error code returned by the ErrorNumber function is API_BUFFER_OVERFLOW.

Returned value

BufferSizeRequired returns:

- Required buffer size if a buffer overflow occurred.

- An undefined value if a buffer overflow did not occur.

CancelJob -- cancel a print job

Format

```
#include <aopapi.h>
int CancelJob( unsigned int job_id,
               const char* user_name );
```

Description

The `CancelJob` function cancels a print job that has been closed on the JES spool but has not yet been selected for printing. It removes all output data sets for the job on the JES spool. Input parameters are:

<i>job_id</i>	The job ID returned by the <code>ClosePrintFile</code> function or the job ID returned by Infoprint Server if the job was created by some other method.
<i>user_name</i>	The user name passed to the <code>CreatePrintFile</code> function or the submitter name if the job was created by some other method.

Returned value

`CancelJob` returns:

- Zero if the job was successfully canceled.
- A non-zero value if the job was not successfully canceled. You can use the `ErrorNumber` function to obtain an error code, and you can use the `ErrorString` function to obtain a description of the error.

ClosePrintFile -- close an Infoprint Server print job

Format

```
#include <aopapi.h>
int ClosePrintFile( JobToken token,
                   unsigned int* job_id );
```

Description

The `ClosePrintFile` function closes an output data set on the JES spool. If the output data set cannot be closed but *token* refers to a valid job, the output data set is removed from the JES spool as if `AbortPrintFile` had been called. Input parameters are:

<i>token</i>	The job token returned by the <code>CreatePrintFile</code> function.
<i>job_id</i>	A job ID assigned by Infoprint Server and returned by this function. If you call the <code>GetJobInfo</code> or <code>CancelJob</code> functions, you must pass this job ID as an argument. This job ID is assigned to the output data set on the JES spool. The operator can display this job ID using System Display and Search Facility (SDSF) commands or a comparable product. (JES commands do not display this job ID.)

To limit the range of *job_id* to a number from 1 to 65535, define the environment variable `AOP_64K_JOBID` before calling the `CreatePrintFile` function.

Returned value

`ClosePrintFile` returns:

- Zero and sets *job_id* if the output data set was closed successfully.

- A non-zero value if the output data set was not closed successfully. You can use the `ErrorNumber` function to obtain an error code, and you can use the `ErrorString` function to obtain a description of the error.

CreatePrintFile -- create an Infoprint Server print job

Format

```
#include <aopapi.h>
int CreatePrintFile( const char* printer_name,
                    const char* user_name,
                    const char* file_name,
                    const char* attributes,
                    JobToken* out_token );
```

Description

The `CreatePrintFile` function creates an Infoprint Server print job for a printer that is defined in the Infoprint Server Printer Inventory. This function lets you create only one document in each job. After calling `CreatePrintFile`, call these functions:

1. `WritePrintFile` to write the data.
2. `ClosePrintFile` to close the output data set.

For each successful call to `CreatePrintFile`, you must call either `ClosePrintFile` or `AbortPrintFile`. Otherwise, storage is not freed, and the output data set remains open.

Input parameters are:

<i>printer_name</i>	The name of a printer definition in the Infoprint Server Printer Inventory. Printer definition names are case-sensitive.
<i>user_name</i>	The name of the owner of the print job. The name can be from 1 to 31 characters. If <i>user_name</i> is NULL, <code>CreatePrintFile</code> uses the current user login name returned by the <code>_getlogin1()</code> function. The user name becomes the job name of the output data set on the JES spool.
<i>file_name</i>	The name of the document. The name can be any length. If <i>file_name</i> is NULL, the job identifier is used.
<i>attributes</i>	A null-terminated string of job attributes. If <i>attributes</i> is NULL, values specified in the printer definition, or default values, are used for all job attributes. Specify attributes in the format <i>attribute=value</i> , separated by spaces. For more information about the valid job attributes and how to specify them, see the description of the <code>-o</code> option of the <code>lp</code> command and the <i>Using Job Attributes</i> section of the <i>z/OS Infoprint Server User's Guide</i> .
	Tip: If the data to be written is ASCII text data, specify an ASCII code page, such as IS08859-1 in the document-codepage job attribute.
<i>out_token</i>	A job token returned by this function. You must pass this job token as an argument to the <code>WritePrintFile</code> , <code>ClosePrintFile</code> , and <code>AbortPrintFile</code> functions. You can pass the returned job token to functions from threads other than the thread that called the <code>CreatePrintFile</code> function.

Returned value

CreatePrintFile returns:

- Zero and *out_token* if the print job was successfully created.
- A non-zero value if the function was not successful. You can use the `ErrorNumber` function to obtain an error code, and you can use the `ErrorString` function to obtain a description of the error.

EndEnumJobs -- end the enumeration of Infoprint Server print jobs for a printer

Format

```
#include <aopapi.h>
int EndEnumJobs ( QueryToken  token );
```

Description

`EndEnumJobs` ends an enumeration of print jobs. The *token* argument must be a valid token returned by the `BeginEnumJobs` function.

Returned value

`EndEnumJobs` returns:

- Zero if the function was successful.
- A non-zero value if the function was not successful. You can use the `ErrorNumber` function to obtain an error code, and you can use the `ErrorString` function to obtain a description of the error.

EndEnumPrinters -- end the enumeration of Infoprint Server printers

Format

```
#include <aopapi.h>
int EndEnumPrinters ( QueryToken  token );
```

Description

`EndEnumPrinters` ends an enumeration of Infoprint Server printers. The *token* argument must be a valid token returned by the `BeginEnumPrinters` function.

Returned value

`EndEnumPrinters` returns:

- Zero if the function was successful.
- A non-zero value if the function was not successful. You can use the `ErrorNumber` function to obtain an error code, and you can use the `ErrorString` function to obtain a description of the error.

EnumJobs -- enumerate Infoprint Server print jobs for a printer

Format

```
#include <aopapi.h>
int EnumJobs ( QueryToken  token,
              void*       buffer,
              unsigned int  buffer_size,
              unsigned int  max_count,
              JobInfo***   job_list );
```

Description

The EnumJobs function returns information about all print jobs created by Infoprint Server for a specified printer definition. Job information is returned in JobInfo data structures in the buffer you provide to this function. The JobInfo data structures are returned in ascending order of the binary value of the job names. The locale has no effect on the ordering.

You can call EnumJobs as many times as necessary to obtain information about all jobs for the printer definition. EnumJobs returns zero when no more jobs exist for the printer. You must call the EndEnumJobs function to end the enumeration of print jobs.

Input parameters are:

<i>token</i>	A valid query token returned by the BeginEnumJobs function.
<i>buffer</i>	A buffer in which this function can return the JobInfo data structures.
<i>buffer_size</i>	The number of bytes in the buffer.
<i>max_count</i>	The maximum number of jobs for which you want JobInfo structures returned.
<i>job_list</i>	A pointer to an array of pointers to JobInfo data structures. The array of pointers is in the buffer. Do not use the returned array of pointers after the buffer has been freed or modified. The returned value indicates the number of pointers in the array.

Returned value

EnumJobs returns:

- The number of JobInfo structures created if the function was successful. Zero indicates that there are no more jobs for this printer definition.
- A negative value if the function was not successful. You can use the ErrorNumber function to obtain an error code, and you can use the ErrorString function to obtain a description of the error.

EnumPrinters -- enumerate Infoprint Server printers

Format

```
#include <aopapi.h>
int EnumPrinters ( QueryToken    token,
                  void*         buffer,
                  unsigned int   buffer_size,
                  unsigned int   max_count,
                  PrinterInfo*** printer_list );
```

Description

The EnumPrinters function returns information about all of the printers defined by printer definitions in the Infoprint Server Printer Inventory. Printer information is returned in PrinterInfo data structures in the buffer you provide to this function. The PrinterInfo data structures are returned in ascending order of the binary value of the printer definition names. The locale has no effect on the ordering.

You can call EnumPrinters as many times as necessary to obtain information about all printers. EnumPrinters returns zero when information about all printers has been returned. You must call the EndEnumPrinters function to end the enumeration of printers.

Input parameters are:

<i>token</i>	A valid query token returned by the BeginEnumPrinters function.
<i>buffer</i>	A buffer in which this function can return the PrinterInfo data structures.
<i>buffer_size</i>	The number of bytes in the buffer.
<i>max_count</i>	The maximum number of printers for which you want PrinterInfo structures returned.
<i>printer_list</i>	A pointer returned by this function to an array of pointers to PrinterInfo data structures. The array of pointers is in the buffer. Do not use the returned array of pointers after the buffer has been freed or modified. The returned value indicates the number of pointers in the array.

Returned value

EnumPrinters returns:

- The number of PrinterInfo structures created if the function was successful. Zero indicates that information about all printers has been returned.
- A negative value if the function was not successful. You can use the ErrorNumber function to obtain an error code, and you can use the ErrorString function to obtain a description of the error.

ErrorNumber -- return the API error code

Format

```
#include <aopapi.h>
int ErrorNumber();
```

Description

The ErrorNumber function returns a number that indicates the last error returned by the API for the calling thread. This table lists names defined in the aopapi.h file for the error numbers and describes the error. This table does not list all possible error numbers returned by this function. For all errors, you can use the ErrorString function to obtain a description of the error.

Name	Description
AOP_NO_ERROR	No error occurred.
AOP_PRINTER_NOT_FOUND	The printer definition that was specified as an input argument to an API function does not exist in the Printer Inventory.
AOP_JOB_NOT_FOUND	The job ID that was specified as an input argument to an API function does not exist.
AOP_JOB_LIMIT_EXCEEDED	The CreatePrintFile function was called to create a new job, but no free job IDs exist to assign to the job.
AOP_CANCEL_NOT_PERMITTED	The specified user is not the owner of the print job. Only the owner is allowed to cancel a print job.
AOP_INV_CONNECT_ERROR	The Printer Inventory is not active.
AOP_NOT_ENABLED	The use of Infoprint Server is not enabled.
AOP_NOT_AUTHORIZED	The action on the specified object is not permitted.
AOP_INVALID_TOKEN	The job or query token that was specified as an argument to an API function is not valid.

Name	Description
AOP_BUFFER_OVERFLOW	The buffer provided to an API function was not big enough to complete the operation. You can obtain the required buffer size for the operation by calling the <code>BufferSizeRequired</code> function.
AOP_API_INIT_FAILED	The API could not be initialized. Make sure that <code>aopapi.dll</code> can be loaded from one of the directories specified in the <code>LIBPATH</code> environment variable and is accessible to this process.

Returned value

`ErrorNumber` returns a number that indicates the last error.

ErrorString -- return a pointer to the API error message

Format

```
#include <aopapi.h>
const char* ErrorString();
```

Description

The `ErrorString` function returns a pointer to a string that describes the last error returned by the API for the calling thread. The message is a multi-byte string in the language indicated by the current locale when the locale is supported or in English for other locales. These locales are supported: C, En_US, and Ja_JP.

Returned value

`ErrorString` returns:

- A pointer to an error string if an error occurred.
- An empty string if no error occurred.

GetJobInfo -- return the status of an Infoprint Server print job

Format

```
#include <aopapi.h>
JobInfo* GetJobInfo( unsigned int  job_id,
                    void*        buffer,
                    unsigned int  buffer_size );
```

Description

The `GetJobInfo` function returns information about the Infoprint Server print job identified by argument `job_id`. This function returns the `JobInfo` data structure, which describes the job and each document in the job. The `DocumentInfo` data structure, which is part of the `JobInfo` data structure describes each document in the job.

Input parameters are:

<i>job_id</i>	The job ID returned by the <code>ClosePrintFile</code> function or the job ID returned by Infoprint Server if the job was created by some other method.
<i>buffer</i>	A buffer in which this function returns the <code>JobInfo</code> data structure.
<i>buffer_size</i>	The number of bytes in the buffer.

Returned value

GetJobInfo returns:

- A pointer to a JobInfo structure if the function was successful. The address will be in the buffer identified by the *buffer* argument. Do not use this address after you free or modify the buffer.
- A null pointer if the function was not successful. You can use the ErrorNumber function to obtain an error code, and you can use the ErrorString function to obtain a description of the error.

Example

This example calls the GetJobInfo function to get information for the print job identified by job ID 152. It handles a buffer overflow error condition, prints the API error message if another error occurs, and if the GetJobInfo function was successful, prints information returned about the job and each document in the job.

```
#include <aopapi.h>
#include <time.h>
#include <stdio.h>
#include <stdlib.h>
#define NULL_TEST(x) ((x) ? (x) : "(null)")
int main()
{
    int          i;
    JobInfo*     job;
    unsigned int job_number = 152;
    int          buffer_size = 4096;
    void*        buffer = malloc(buffer_size);
    retry: job = GetJobInfo(job_number,
                           buffer,
                           buffer_size);

    if (job == 0) {
        if (ErrorNumber() == AOP_BUFFER_OVERFLOW) {
            buffer_size <<= 1;
            buffer = realloc(buffer, buffer_size);
            goto retry;
        } else {
            fprintf(stderr, "API error: %s\n", ErrorString());
            exit(1);
        }
    }

    printf("Job %-d:\n", job->id);
    printf("  user name: %s\n", job->user_name);
    printf("  printer: %s\n", job->printer_name);
    printf("  submitted: %s", ctime(>submitted));
    printf("  priority: %-d\n", job->priority);
    printf("  name text: %s\n", NULL_TEST(job->name_text));
    for (i=0; i<job->ndocuments; ++i) {
        printf("  document: %s\n", NULL_TEST(job->documents[i]->name));
        printf("    status: %d\n", job->documents[i]->status);
        printf("    size: %-d\n", job->documents[i]->size);
        printf("    format: %-d\n", job->documents[i]->format);
    }
}
```

GetPrinterInfo -- return information about an Infoprint Server printer

Format

```
#include <aopapi.h>
PrinterInfo* GetPrinterInfo( const char*  printer_name,
                             void*       buffer,
                             unsigned int  buffer_size );
```


Description

The `GetPrinterInfo` function returns information about the printer definition in the Infoprint Server Printer Inventory identified by argument *printer_name*. This function returns the `PrinterInfo` data structure.

Input parameters are:

<i>printer_name</i>	The name of the printer definition in the Infoprint Server Printer Inventory.
<i>buffer</i>	An buffer in which this function returns the <code>PrinterInfo</code> data structure.
<i>buffer_size</i>	The number of bytes in the buffer.

Returned value

`GetPrinterInfo` returns:

- A pointer to a `PrinterInfo` structure if the function was successful. The address will be in the buffer identified by the *buffer* argument. Do not use this address after you free or modify the buffer.
- A null pointer if the function was not successful. You can use the `ErrorNumber` function to obtain an error code, and you can use the `ErrorString` function to obtain a description of the error.

InitAPI -- initialize the API

Format

```
#include <aopapi.h>
int InitAPI();
```

Description

The `InitAPI` function initializes the API. You can call `InitAPI` multiple times from a single process and concurrently from multiple threads. You do not need to explicitly call `InitAPI` because it is called automatically on the first call to any other API function if it has not been called explicitly. Use this function if your application needs to determine if the API can be successfully initialized separately from other calls.

Returned value

`InitAPI` returns:

- Zero if the API was successfully initialized.
- A non-zero value if the DLL was not loadable or the API was not successfully initialized for another reason. You can use the `ErrorNumber` function to obtain an error code, and you can use the `ErrorString` function to obtain a description of the error.

ReleaseTransform -- release a connection with a transform daemon

Format

```
#include <aopapi.h>

typedef enum {
    TRANSFORM_COMPLETE = 2,
    TRANSFORM_ABORT    = 3
} ReleaseType;
int      ReleaseTransform( Transform*, ReleaseType release_type );
```

Description

The ReleaseTransform function releases the connection with the transform daemon, which was established by the RequestTransform function. Input parameters are:

- Transform*** A pointer to the Transform data structure returned by the RequestTransform function.
- release_type* The status of the transform:
- TRANSFORM_COMPLETE: This status indicates that the transform completed normally. Specify this status when the transform completes processing, whether successful or not, in a known good state.
 - TRANSFORM_ABORT: This status indicates that the transform completed abnormally. Specify this status when the status of the transform is uncertain, due to an unexpected transform failure, a communication failure, and so on.

Returned value

ReleaseTransform returns:

- Zero if the function was successful.
- A non-zero value if the function was not successful. You can use the ErrorNumber function to obtain an error code, and you can use the ErrorString function to obtain a description of the error.

RequestTransform -- establish a connection with a transform daemon

Format

```
#include <aopapi.h>
Transform* RequestTransform( const char* transform_class );
```

Description

The RequestTransform function establishes a connection with a transform daemon. Call this function only when you are writing a transform that is managed by the Infoprint Server Transform Manager. After the connection is established, you can use the identifier in field **fd** in the returned Transform data structure to receive data from and send data to the transform daemon.

transform_class must match the transform name and class specified in the **transform** attribute in the transform configuration file (**aopxfd.conf**). The format of *transform_class* is: *transformname[_transformclass]*. *transform_class* can be a maximum of 63 bytes in length. For information about the transform configuration file, see “Format of an entry in the transform configuration file” on page 160.

To release the transform connection, call the ReleaseTransform function.

Returned value

RequestTransform returns:

- A pointer to a Transform structure if the function was successful.
- A null pointer if the function was not successful.

SetTerminationHandler -- register a termination function

Format

```
#include <aopapi.h>
int SetTerminationHandler(TerminationHandler handler,
                          void* argument);
```

Description

The SetTerminationHandler function registers a function to handle termination of the API threads created in the current process (Language Environment enclave). If the API encounters a condition on one of the threads it manages that would cause enclave termination, the API calls the termination function pointed to by *handler* with argument *argument*. If the termination function returns without terminating the enclave, the API terminates the enclave.

To remove the termination handler function, call SetTerminationHandler with a null argument for *handler*.

Tip: For information about how to write a termination handler function, see “Writing a termination handler function” on page 477.

Returned value

SetTerminationHandler returns:

- Zero if the function was successful.
- A non-zero value if the function was not successful. You can use the ErrorNumber function to obtain an error code, and you can use the ErrorString function to obtain a description of the error.

TermAPI -- terminate the API

Format

```
#include <aopapi.h>
int TermAPI();
```

Description

The TermAPI function terminates the API. When the number of TermAPI calls equals the number of InitAPI calls, then all resources associated with the API are freed and the aopapi.dll is unloaded. You do not need to explicitly call TermAPI because this function is done automatically when your application terminates.

Returned value

TermAPI returns:

- Zero if the API was successfully terminated.
- A non-zero value if the API was not successfully terminated. You can use the ErrorNumber function to obtain an error code, and you can use the ErrorString function to obtain a description of the error.

WritePrintFile -- write data to an Infoprint Server print job

Format

```
#include <aopapi.h>
int WritePrintFile( JobToken    token,
                   const char*  buffer,
                   unsigned int  buffer_size );
```

Description

The `WritePrintFile` function writes data sequentially to an output data set on the JES spool. Infoprint Server determines the data format automatically unless the **document-format** job attribute is passed to the `CreatePrintFile` function. If the printer definition in the Printer Inventory specifies a filter for the data format, Infoprint Server calls that filter to transform the data before writing it to the JES spool.

If the data format is text data, and if a printer code page is specified in the printer definition, Infoprint Server converts the data from the document code page to the printer code page before writing it to the JES spool. Because the default document code page is determined by the locale and is usually an EBCDIC code page, to print an ASCII document, specify an ASCII code page in the **document-codepage** job attribute when you call the `CreatePrintFile` function. Call the `WritePrintFile` function as many times as necessary to write all the data. After all data has been successfully written, you must call the `ClosePrintFile` function to close the output data set on the JES spool or the `AbortPrintFile` function to remove the data set from the JES spool.

If `WritePrintFile` is not successful for a valid *token*, the spooling operation associated with *token* is terminated and the output data set is removed from the JES spool as if the `AbortPrintFile` function had been called.

Input parameters are:

<i>token</i>	A valid job token returned by the <code>CreatePrintFile</code> function.
<i>buffer</i>	A buffer containing the data to be written to the output data set on the JES spool.
<i>buffer_size</i>	The number of bytes in the buffer.

Returned value

`WritePrintFile` returns:

- Zero if the data is written successfully.
- A non-zero value if data is not successfully written.

Programming considerations

This section describes considerations for writing and link-editing your C or C++ applications that use the API.

Redefining API function names

The names of the API functions are generic, for example `InitAPI` and `TermAPI`. These function names are mapped (using the `map` pragma) to names that are prefixed with `AOP` to reduce the chance of conflicts with your applications. For example:

```
#pragma map(InitAPI, "AOPINIT")
```

If the name of an API function conflicts with a function name defined in the same scope, you can redefine the API function name before including the API header file (`aopapi.h`). For example, you could redefine `InitAPI` to `AOP_InitAPI`, as shown in this example:

```
#define InitAPI AOP_InitAPI
#include <aopapi.h>
```

Linking to API functions

To link to the functions in this API, include the static library `libaopapi.a` in the link-edit for your application. The functions in `libaopapi.a` are stubs that call the actual code, which is in executable file `aopapi.dll`.

APF-authorizing your application

Your application must run APF-authorized to use the API. If your application is in a z/OS UNIX file, use the z/OS UNIX command `extattr` to APF-authorize it. Also, link your program with the `AC=1` specification.

Writing a termination handler function

If your application needs to do cleanup before an API-created thread terminates abnormally, you can write your own termination handler function. The API calls your termination handler when an abnormal-termination situation occurs.

If you write a termination handler function, you must call the `SetTerminationHandler` function to register your termination handler.

Your termination handler receives three parameters:

<i>handler_argument</i>	The argument that was passed to the <code>SetTerminationHandler</code> function.
<i>cleanup_handler</i>	A pointer to a cleanup handler function that your termination handler must call.
<i>cleanup_argument</i>	An argument that the termination handler must pass to the cleanup handler.

Your termination handler function should do these functions:

- Produce a dump or equivalent diagnostic of the process to help diagnose problems in your application and the API.
- Do any desired cleanup actions. All actions permitted by an Language Environment-condition handler are allowed in the termination handler.
- Call the cleanup handler with *cleanup_argument*. This call must be the last operation in the termination handler.
- Either (1) call the `exit` function to terminate the process or (2) simply return, which causes the function that initiated the API thread to fail but lets the process continue.

Compile the executable code that contains the termination handler function with either the `NODLL` or `DLL(CBA)` option. The `CBA` sub-option is required if the code is a `DLL` because the cleanup handler function pointer passed to the termination handler is compiled with the `NODLL` option.

The `aopapi.h` header file contains these definitions for the termination handler and cleanup handler functions:

```
typedef void (*CleanupHandler) (void* cleanup_argument );
typedef void (*TerminationHandler) (void* handler_argument,
                                     CleanupHandler cleanup_handler,
                                     void* cleanup_argument );
```

Customization requirements

Before you run an application that uses the API, you must do these tasks to customize Infoprint Server:

- Customize the Printer Inventory Manager component. See Chapter 3, “Customizing the Printer Inventory Manager,” on page 27.
- Customize the Print Interface component of Infoprint Server. You need to do only the general customization tasks for the Print Interface component. See Chapter 4, “Customizing Print Interface,” on page 115.
- (Optional) Customize the Transform Interface and a transform product. See Chapter 5, “Customizing Transform Interface,” on page 157.

In addition, do these customization task related to the API:

- (Optional) Define environment variable `AOP_64K_JOBID`. If this environment variable is defined, the `CreatePrintFile` function limits the job ID of a new 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.

Tip: This variable is defined for use by the z/OS SMB server. Other applications should not define this variable because system performance might be degraded.

After customizing Infoprint Server, do these tasks described in *z/OS Infoprint Server Operation and Administration*:

1. Start the Printer Inventory Manager daemon. If you want to transform data from one format to another on the z/OS system, also start the Infoprint Server Transform Manager daemon.

2. Create a printer definition in the Infoprint Server Printer Inventory for each target printer and specify the printer attributes required by the Print Interface component. If you want to transform data, also specify the names of the transform filters in the printer definitions.

Appendix G. Migrating from IP PrintWay basic mode to extended mode

This section describes how to move all processing from IP PrintWay basic mode to IP PrintWay extended mode. For complete information about how to customize IP PrintWay extended mode, see Chapter 7, “Customizing IP PrintWay,” on page 199.

Steps for migrating from IP PrintWay basic mode to extended mode:

1. Customize IP PrintWay extended mode:

- In the Infoprint Server configuration file, **aopd.conf**, specify the **outd** and **ssid** values in the **start-daemons** attribute to start the IP PrintWay extended mode daemons and the Infoprint Central daemon. This change takes effect only when you restart Infoprint Server in a later step.
- If dynamic configuration is enabled, edit the system configuration definition in the Printer Inventory to specify these attributes:
 - **applid**: If the **Applid** field in an IP PrintWay basic mode FSS definition in the Printer Inventory contains a value, specify the same value in the **applid** attribute in the system configuration definition. IP PrintWay basic mode and extended mode can use the same application program ID if you do not run both modes at the same time.
 - **ebcdic-codepage**: Specify the same code page that is currently specified in the **Document code page** field in the IP PrintWay basic mode FSS definition in the Printer Inventory.
 - **log-retention**: Specify how many days worth of historical print job information to display in Infoprint Central. If you do not use the MVS system logger for Infoprint Server messages, this attribute also specifies many days worth of messages Infoprint Server retains in the z/OS UNIX file system.
 - **mail-do-not-add-suffixes**: If the **AOP_MAIL_DO_NOT_ADD_SUFFIX** environment variable is specified in the IP PrintWay basic mode startup procedure, specify the same value in this attribute.
 - **mail-preserve-suffixes**: If the **AOP_MAIL_PRESERVE_SUFFIXES** environment variable is specified in the IP PrintWay basic mode startup procedure, specify the same value in this attribute.
 - **mailer-options**: If the **AOPMAILER_OPTIONS** environment variable is specified in the IP PrintWay basic mode startup procedure, specify the same value in this attribute.
 - **mailer-path-name**: If the **AOPMAILER** environment variable is specified in the IP PrintWay basic mode startup procedure, specify the same value in this attribute.

Tip: If dynamic configuration is not enabled, you can specify the equivalent attributes in the **aopd.conf** file and the equivalent environment variables in the **aopstart** EXEC. For information about the equivalent attributes and environment variables, see Table 11 on page 70.

For information about how to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 80.

- Increase the space available to the **/var/Printsrv** directory if necessary. See “Increasing the space available in **/var/Printsrv** directory (extended mode)” on page 207.

- Adjust the values for the MAXPROCSYS and MAXPROCUSER parameters in the BPXPRMxx parmlib member if necessary. See “Updating the BPXPRMxx member of SYS1.PARMLIB (extended mode)” on page 207.
- Create IP PrintWay extended mode job-selection rules in the Printer Inventory. In the job-selection rules, specify the same work-selection criteria that you currently specify for the IP PrintWay basic mode functional subsystem applications (FSAs). IP PrintWay extended mode does not process any print jobs until you create work-selection rules. See “Creating IP PrintWay job-selection rules (extended mode)” on page 208.
- Give IP PrintWay extended mode RACF authorization to select output data sets from the JES spool. See “Authorizing IP PrintWay to select data sets from the JES spool (extended mode)” on page 214.
- Give IP PrintWay extended mode RACF authorization to write SMF type 6 records. See “Authorizing IP PrintWay to write SMF type 6 records (extended mode)” on page 215.
- Edit the **aopstart** EXEC to specify these environment variables if necessary:
 - **_BPXK_SETIBMOPT_TRANSPORT**: If you run more than one TCP/IP stack, specify the name of the z/OS UNIX stack. IP PrintWay extended mode must use the z/OS UNIX stack.
 - **STEPLIB**: Specify the STEPLIB environment variable if the IP PrintWay exit library or the FCB image library is not in the system LNKLIST. Also add the library to the STEPLIBLIST facility of SYS1.PARMLIB.

Tip: If dynamic configuration is not enabled, you might need to specify these environment variables: AOP_APPLID, AOP_DO_NOT_ADD_SUFFIX, AOP_PRESERVE_SUFFIXES, AOPMAILER, and AOPMAILER_OPTIONS, See “Setting environment variables for IP PrintWay (extended mode)” on page 215.
- Recompile the IP PrintWay SMF exit.
- Review other IP PrintWay exits to see if changes are required. See “Comparing IP PrintWay basic mode and extended mode exits” on page 238.
- If you specify the **concatenation-separators=document** attribute in your IP PrintWay FSS definitions, and your IP PrintWay Begin Data Set or End Data Set exit adds printer instructions in records, change the exit to specify the printer instructions in the ANFUEXTP control block instead of in records. See “Begin Data Set exit” on page 258.

2. Customize Infoprint Central. See Chapter 8, “Customizing Infoprint Central,” on page 327.

3. Restart the z/OS HTTP Server to pick up changes. For instructions, see *z/OS HTTP Server Planning, Installing, and Using*.

4. Change IP PrintWay printer definitions in the Printer Inventory if necessary. Because IP PrintWay extended mode and IP PrintWay basic mode share the same printer definitions, in most cases, no changes are required. However, you might need to change printer definitions that specify any of these fields:

- **Data format**
- **Double-byte translate table**

- **Line termination**
- **PostScript header** (Landscape, Always landscape, or Add option)
- **SOSI mode**
- **Translation dataset qualifier**

For information about how to change printer definitions for IP PrintWay extended mode, see *z/OS Infoprint Server Operation and Administration*.

5. If you currently specify a default font in the JES definitions for IP PrintWay basic mode FSAs and also use the AFP to PDF, AFP to PostScript, or AFP to PCL transform, specify the same default font in the printer definitions for the IP PrintWay printers. Or, if you want the default font to apply to all IP PrintWay printers, specify the font in the AOP_CHARS variable in the transform configuration file, **aopxfd.conf**. IP PrintWay extended mode does not use a default font supplied by JES like IP PrintWay basic mode does.

6. If you currently specify a default FCB in the JES definitions for IP PrintWay basic mode FSAs, ask job submitters to specify the same FCB in the FCB JCL parameter or **forms-control-buffer** job attribute to use this FCB. IP PrintWay extended mode does not use a default FCB supplied by JES like IP PrintWay basic mode does.

7. Review the SMF type 6 record for IP PrintWay extended mode to see if changes affect your accounting programs. For the format of the SMF6 record, see *z/OS MVS System Management Facilities (SMF)*. For information about the differences between the records that IP PrintWay basic mode and extended mode write, see *z/OS Infoprint Server Operation and Administration*.

8. Change the manual and automatic procedures you use to search for IP PrintWay messages. IP PrintWay extended mode writes its messages to the Infoprint Server common message log. IP PrintWay extended mode messages have different IDs and message texts than IP PrintWay basic mode. You can use the **aoplogu** command to retrieve messages from the common message log. Operators can use Infoprint Central to see all messages in the common message log for specific printers and print jobs. For information about the **aoplogu** command, see *z/OS Infoprint Server Operation and Administration*.

9. Stop any IP PrintWay basic mode FSAs that select the same print jobs from the JES spool as IP PrintWay extended mode does.

Tip: The work-selection criteria for IP PrintWay basic mode FSAs determine which print jobs IP PrintWay basic mode selects. The job-selection rules in the Printer Inventory determine which print jobs IP PrintWay extended mode selects.

10. To start IP PrintWay extended mode and Infoprint Central, stop all Infoprint Server daemons and restart Infoprint Server. For information about how to stop and start daemons, see *z/OS Infoprint Server Operation and Administration*.

After you start IP PrintWay extended mode, you can submit print jobs that meet the criteria in the job-selection rules.

To work with print jobs on the JES queue, control IP PrintWay printers, and start and stop IP PrintWay job-selection rules, you can use Infoprint Central. For information about how to use Infoprint Central, see *z/OS Infoprint Server Operation and Administration*. For detailed usage information, see the Infoprint Central online help system.

To diagnose problems, you can use Infoprint Central to look at messages for selected print jobs and printers, or you can use the **aoplogu** command to see all messages in the common message log.

When you are sure that you no longer need to run IP PrintWay basic mode, you can delete:

- FSS and FSA definitions for IP PrintWay basic mode from the Printer Inventory.
- JES initialization statements for IP PrintWay basic mode FSSs and FSAs.
- IP PrintWay basic mode startup procedures.
- IP PrintWay message data set (default name is ANF.MSGFILE) unless NetSpool writes messages to this data set.
- IP PrintWay transmission queue data set (default name is ANF.QUEUE).
- IP PrintWay Message (ANFUXMSG) exit (IP PrintWay extended mode does not support this exit).

Note: If you want to run IP PrintWay basic mode and extended mode at the same time, IP PrintWay extended mode must select different output data sets from the JES spool than basic mode. Also, if IP PrintWay sends print jobs to VTAM-controlled printers, IP PrintWay extended mode must use a different VTAM APPL resource definition than basic mode. To accomplish this:

- Create IP PrintWay extended mode job-selection rules that cause IP PrintWay extended mode to select different output data sets from the JES spool than IP PrintWay basic mode selects. For example, if IP PrintWay basic mode selects output data sets in class P, extended mode could select output data sets in class Q. (See “Creating job-selection rules in the Printer Inventory (extended mode)” on page 211.)
- If you print to VTAM-controlled printers, create a new VTAM APPL statement for IP PrintWay extended mode and specify it in the AOP_APPLID environment variable in the **aopstart** EXEC or, if dynamic configuration is enabled, in the **applid** attribute in the system configuration definition. (See “Creating a VTAM APPL resource definition for IP PrintWay” on page 236.)

Appendix H. Exploiting new functions in z/OS V2R1

This section describes how to exploit the Infoprint Server functions introduced in z/OS V2R1.

Note: All installations must perform some actions when migrating to z/OS V2R1. For information about the actions that installations must take to maintain the same functional behavior in z/OS V2R1 as in previous releases, see *z/OS Migration*.

Dynamic configuration

The dynamic configuration function lets you change many Infoprint Server configuration attributes while Infoprint Server is running, without stopping and restarting Infoprint Server. These attributes are stored in the system configuration definition in the Printer Inventory. By default, dynamic configuration is not enabled.

Some advantages of exploiting dynamic configuration are:

- Authorized administrators can use the Infoprint Server ISPF panels to view and change the dynamic attributes.
- If you change an attribute in the system configuration definition, with a few exceptions, you do not need to stop and restart Infoprint Server for the change to take effect.
- You can exploit new functions in Infoprint Server that require dynamic configuration. For example, you can exploit the MVS system logger function.

Note: Enable dynamic configuration only after you are sure that you will not return to a release before z/OS V2R1. This is because if you change attributes in the system configuration definition in z/OS V2R1, the changes are lost when you return to a previous release.

Related information: For information about to enable dynamic configuration, see “Enabling dynamic configuration” on page 68.

MVS system logger

In z/OS V2R1, Infoprint Server can write messages to a log stream that the MVS system logger manages. If you don't set up the MVS system logger for messages, Infoprint Server instead writes messages in the Infoprint Server base directory (default is `/var/Printsrv`).

Some advantages of exploiting the MVS system logger are:

- You can retain Infoprint Server messages for a longer time because the system logger can automatically offload log records to auxiliary storage when the log stream is nearing capacity. If you do not use the system logger, you can run out of space in the file system where Infoprint Server writes messages.
- If Infoprint Server runs on more than one system in a sysplex, all instances of Infoprint Server can write messages to the same log stream (a coupling-facility log stream is required). The Infoprint Server **aoplogu** command lets you view the messages merged from all systems.

For information about how to exploit the MVS system logger, see “Setting up the MVS system logger for the common message log” on page 88.

SMF type 6 record enhancements

In z/OS V2R1, IP PrintWay extended mode includes accounting information from the JOB JCL statement in a new field in the SMF type 6 record. You can modify your accounting applications to use this additional job information.

For information about the new SMF6ACCT field in the SMF6 record, see *z/OS MVS System Management Facilities (SMF)*.

Appendix I. Accessibility

Accessible publications for this product are offered through the z/OS Information Center, which is available at <http://www.ibm.com/systems/z/os/zos/bkserv/>. If you experience any accessibility problems with the z/OS Information Center, send an email to mhvrcfs@us.ibm.com or write to the following address:

IBM Corporation
Attention: MHVRCFS Reader Comments
Department H6MA, Building 707
2455 South Road
Poughkeepsie, NY 12601-5400
US

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully. The major accessibility features in z/OS enable users to:

- Use assistive technologies such as screen readers and screen magnifier software
- Operate specific or equivalent features using only the keyboard
- Customize display attributes such as color, contrast, and font size.

Using assistive technologies

Assistive technology products, such as screen readers, function with the user interfaces found in z/OS. Consult the assistive technology documentation for specific information when using such products to access z/OS interfaces.

Keyboard navigation of the user interface

Users can access z/OS user interfaces using TSO/E or ISPF. Refer to *z/OS TSO/E Primer*, *z/OS TSO/E User's Guide*, and *z/OS ISPF User's Guide Vol I* for information about accessing TSO/E and ISPF interfaces. These guides describe how to use TSO/E and ISPF, including the use of keyboard shortcuts or function keys (PF keys). Each guide includes the default settings for the PF keys and explains how to modify their functions.

Notices

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

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

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not 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

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

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

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

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.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

Site Counsel
IBM Corporation
2455 South Road
Poughkeepsie, NY 12601-5400
USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

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

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Policy for unsupported hardware

Various z/OS elements, such as DFSMS, HCD, JES2, JES3, and MVS, contain code that supports specific hardware servers or devices. In some cases, this device-related element support remains in the product even after the hardware devices pass their announced End of Service date. z/OS may continue to service element code; however, it will not provide service related to unsupported hardware devices. Software problems related to these devices will not be accepted for service, and current service activity will cease if a problem is determined to be associated with out-of-support devices. In such cases, fixes will not be issued.

Minimum supported hardware

The minimum supported hardware for z/OS releases identified in z/OS announcements can subsequently change when service for particular servers or devices is withdrawn. Likewise, the levels of other software products supported on a particular release of z/OS are subject to the service support lifecycle of those products. Therefore, z/OS and its product publications (for example, panels, samples, messages, and product documentation) can include references to hardware and software that is no longer supported.

- For information about software support lifecycle, see: <http://www.ibm.com/software/support/systemsz/lifecycle/>
- For information about currently-supported IBM hardware, contact your IBM representative.

Programming interfaces

This information documents programming interfaces intended to allow you to write programs to obtain the services of Infoprint Server.

Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com) are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol ([®] or [™]), these symbols indicate U.S. 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 “Copyright and trademark information” at <http://www.ibm.com/legal/copytrade.shtml>.

Adobe and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

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

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

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

Glossary

This glossary defines technical terms and abbreviations used in Infoprint Server documentation. If you do not find the term you are looking for, see the index or see the IBM Terminology website at <http://www.ibm.com/software/globalization/terminology/>.

These cross-references are used in this glossary:

Contrast with. Refers to a term that has an opposite or substantively different meaning.

See. Refers to multiple-word terms in which this term appears.

See also. Refers to related terms that have similar, but not synonymous, meanings.

Synonym for. Appears in the commentary of a less desirable or less specific term and identifies the preferred term that has the same meaning.

Synonymous with. Appears in the commentary of a preferred term and identifies less desirable or less specific terms that have the same meaning.

A

access method control block. A control block that links an application program to Virtual Storage Access Method (VSAM) or Virtual Telecommunications Access Method (VTAM).

ACIF. See AFP Conversion and Indexing Facility.

Advanced Function Presentation (AFP). A set of licensed programs, together with user applications, that use the all-points-addressable concept to print data on a wide variety of printers or to display data on a variety of display devices. AFP includes creating, formatting, archiving, retrieving, viewing, distributing, and printing information.

Advanced Interactive eXecutive. See AIX.

AFP. See Advanced Function Presentation.

AFP Conversion and Indexing Facility (ACIF). An Advanced Function Presentation (AFP) program that converts a print file into a Mixed Object Document Content Architecture-Presentation (MO:DCA-P) document, creates an index file for later retrieval and viewing, and retrieves resources used by an AFP document into a separate file.

AFP printer driver. A printer driver that runs on a Windows workstation and creates output in Advanced Function Presentation (AFP) format for printing on AFP printers.

AFP Viewer plug-in. A program that runs on a Windows workstation and is used for viewing files in Advanced Function Presentation (AFP) format from a Web browser.

AIX (Advanced Interactive eXecutive). A UNIX operating system developed by IBM that is designed and optimized to run on POWER[®] microprocessor-based hardware such as servers, workstations, and blades.

alphanumeric character. A lowercase or uppercase letter, number, or special symbol.

API. See application programming interface.

application program identifier (APPLID). The name by which a logical unit is known in a VTAM network.

application programming interface (API). An interface that allows an application program to use specific data or functions of the operating system or another program.

APPLID. See application program identifier.

B

banner page. A page printed before the data set is printed.

binary data. A type of data consisting of numeric values stored in bit patterns of 0s and 1s.

bind request. A request to establish a connection between systems or logical units.

broadcast. The simultaneous transmission of data to more than one destination.

buffer. An area of storage that compensates for the different speeds of data flow or timings of events by temporarily holding a block of data that is waiting to be processed or written to an I/O device.

burst. To separate continuous-forms paper into individual sheets.

C

carriage control character. A character that is used to specify a write, space, or skip operation.

carriage return (CR). The movement of the printing position or display position to the first position on the same line.

catalog. (1) A directory of files and libraries, with reference to their locations. (2) To enter information about a data set or a library into a catalog. (3) A data set that contains information about other data sets.

CICS (Customer Information Control System). An IBM licensed program that provides online transaction-processing services and management for business applications.

client/server. Pertaining to the model of interaction in distributed data processing in which a program on one computer sends a request to a program on another computer and awaits a response. The requesting program is called a client; the answering program is called a server.

code point. A unique bit pattern that represents a character in a code page.

coexistence. The ability of two or more entities to function in the same system or network.

common message log. A log that can contain messages from several Infoprint Server components, including Print Interface, NetSpool, and IP PrintWay extended mode. Infoprint Central and the aoplogu command can display messages in the common message log.

connection. In data communication, an association established between entities for conveying information.

cookie. Information that a server stores on a client machine and accesses during subsequent sessions. Cookies allow servers to remember specific information about clients.

copy group. (1) An internal object in a form definition or a print data set that controls such items as modifications to a form, page placement, and overlays. (2) One or more copies of a sheet of paper or form. Each copy can have modifications, such as text suppression, page position, forms flash, overlays, paper-source, and duplex printing.

CR. See carriage return.

Customer Information Control System. See CICS.

D

daemon. A program that runs unattended to perform continuous or periodic functions, such as network control.

data set. The major unit of data storage and retrieval, consisting of a collection of data in one of several

prescribed arrangements and described by control information to which the system has access.

data stream. The commands, control codes, data, or structured fields that are transmitted between an application program and a device such as printer or nonprogrammable display station.

DBCS. See double-byte character set. See also single-byte character set.

default. Pertaining to an attribute, value, or option that is assumed when none is explicitly specified.

DLL filter. A filter 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. See also filter, UNIX filter.

dotted decimal notation. The syntactical representation for a 32-bit integer that consists of four 8-bit numbers written in base 10 and separated by dots. IP addresses are represented in dotted decimal notation.

double-byte character set (DBCS). A set of characters in which each character is represented by two bytes. These character sets are commonly used by national languages, such as Japanese and Chinese, that have more symbols than can be represented by a single byte. See also single-byte character set.

download. To transfer data from a computer to a connected device, such as a workstation or personal computer.

drain. An operator action to halt the flow of jobs to a printer, usually to stop the printer or to change print options.

E

EBCDIC. See Extended Binary Coded Decimal Interchange Code.

encryption. In computer security, the process of transforming data into an unintelligible form in such a way that the original data either cannot be obtained or can be obtained only by using a decryption process.

environment variable. A variable that specifies the operating environment for a process. For example, environment variables can describe the home directory, the command search path, the terminal in use, and the current time zone.

Extended Binary Coded Decimal Interchange Code (EBCDIC). A coded character set of 256 8-bit characters developed for the representation of textual data.

F

filter. In Infoprint Server, a program that can add, delete, or modify input data. Infoprint Server provides support for two types of filter programs: DLL filters and UNIX filters. See also DLL filter.

font. (1) A particular type style (for example, Bodoni or Times Roman) that contains definitions of character sets, marker sets, and pattern sets. (2) A family or assortment of characters of a given size and style, for example, 9-point Bodoni modern.

form definition. An Advanced Function Presentation (AFP) resource object that defines the characteristics of the form or printed media, including: overlays to be used, duplex printing, text suppression, the position of composed-text data on the form, and the number and modifications of a page.

FSA. See functional subsystem application.

FSS. See functional subsystem.

functional subsystem (FSS). An extension of JES that runs in an address space separate from the JES address space. An FSS provides support for a function peripheral to JES processing, such as a peripheral device or other component.

functional subsystem application (FSA). An application that uses the support facilities of the functional subsystem (FSS) to communicate with JES.

H

half-session. In SNA, one of the locations in a logical connection in a network.

HFS file. An object that exists in a mountable file system.

hiperspace. A high-performance, virtual-storage space of up to 2 gigabytes (GB). Unlike an address space, a hiperspace contains only user data and does not contain system control blocks or common areas; code does not run in a hiperspace. Unlike a data space, data in a hiperspace cannot be referenced directly; data must be moved to an address space in blocks of 4 KB before being processed.

Historical Inventory. A set of files that contain information about Infoprint Server print jobs that are no longer on the JES spool. These print jobs finished processing or were deleted from the JES spool. Infoprint Central can display information about print jobs in the Historical Inventory.

home directory. The current directory associated with the user at the time of login.

host. A computer that is connected to a network and provides an access point to that network. The host can be a client, a server, or both a client and server simultaneously.

host address. See IP address.

host name. In Internet communication, the name given to a computer. Sometimes, host name is used to mean the fully qualified domain name; other times, it is used to mean the most specific subname of a fully qualified domain name. For example, if mycomputer.city.company.com is the fully qualified domain name, either of the following host names can be used: mycomputer.city.company.com or mycomputer.

I

IMS/VS. See Information Management System/Virtual Storage.

Infoprint Server. An element of z/OS that supports printing on local printers and remote printers in a TCP/IP or SNA network. With Infoprint Server, users can submit print requests from remote workstations in a TCP/IP network, from z/OS UNIX System Services applications, from batch applications, from VTAM applications (such as CICS and IMS), and from SAP R/3.

Information Management System/Virtual Storage (IMS/VS). A database/data communication (DB/DC) system that can manage complex databases and networks.

inline resource. A resource contained in a print file or a print data set.

internal copy group. A copy group within a print data set instead of within a form definition.

Internet Printing Protocol (IPP). An application-level protocol that supports distributed printing on the Internet. IPP uses a client/server architecture and defines the interactions between IPP clients (typically workstations) and IPP servers.

Internet Protocol address. See IP address.

IP address (Internet Protocol address). A unique address for a device or logical unit on a network that uses the Internet Protocol standard.

IPP. See Internet Printing Protocol.

IP PrintWay (PrintWay). A component of Infoprint Server that transmits output data sets from the JES spool to printers in a TCP/IP and SNA network or to email destinations.

J

JCL. See job control language.

JES. See Job Entry Subsystem. See also JES2, JES3.

JES2. An MVS subsystem that receives jobs into the system, converts them to internal format, selects them for execution, processes their output, and purges them from the system. In an installation with more than one processor, each JES2 processor independently controls its job input, scheduling, and output processing. See also Job Entry Subsystem, JES3.

JES3. An MVS subsystem that receives jobs into the system, converts them to internal format, selects them for execution, processes their output, and purges them from the system. In complexes that have several loosely coupled processing units, the JES3 program manages processors so that the global processor exercises centralized control over the local processors and distributes jobs to them using a common job queue. See also Job Entry Subsystem, JES2.

job control language (JCL). A command language that identifies a job to an operating system and describes the job's requirements.

Job Entry Subsystem (JES). An IBM licensed program that receives jobs into the system and processes all output data that is produced by jobs. See also JES2, JES3.

job-selection rule. A method that determines which print jobs IP PrintWay extended mode selects to print. The administrator defines job-selection rules in the Printer Inventory.

K

Kanji. A graphic character set consisting of symbols used in Japanese ideographic alphabets. Each character is represented by 2 bytes.

L

LAN. See local area network.

line data. Data prepared for printing on a line printer. Line data can contain carriage-control characters and table-reference characters (TRC) for spacing and font selections.

line printer daemon (LPD). The printer server that allows other hosts to access its printer.

line printer requester (LPR). A client that lets the local host submit a file for printing on a remote printer server.

local area network (LAN). A network that connects several devices in a limited area (such as a single building or campus) and that can be connected to a larger network.

locale. A setting that identifies language or geography and determines formatting conventions such as collation, case conversion, character classification, the language of messages, date and time representation, and numeric representation.

logical printer. In NetSpool, the target of the VTAM print data, which acts as the secondary logical unit (SLU) on the session.

logical unit (LU). An access point through which a user or application program accesses the SNA network to communicate with another user or application program.

logon mode. In VTAM, a subset of session parameters specified in a logon mode table for communication with a logical unit (LU).

logon mode table. In VTAM programs, a set of entries for one or more logon modes.

LPD. See line printer daemon.

LPR. See line printer requester.

LU. See logical unit.

LU type. The classification of a logical unit in terms of the specific subset of SNA protocols and options that it supports for a given session. LU type specifies the values allowed in the session activation request, the usage of data stream controls, function management headers, request unit parameters, sense data values, and presentation services protocols.

M

Management Information Base (MIB). In the Simple Network Management Protocol (SNMP), a database of objects that can be queried or set by a network management system.

MB (Mbyte). See megabyte.

Mbyte (MB). See megabyte.

megabyte (Mbyte, MB). For processor storage, real and virtual storage, and channel volume, 2 to 20th power or 1,048,576 bytes. For disk storage capacity and communications volume, 1 000 000 bytes.

MIB. See Management Information Base.

migration. Activities that relate to the installation of a new version or release of a program to replace an earlier level. Completion of these activities ensures that

the applications and resources on your system will function correctly at the new level.

MO:DCA-P data. Print data that has been composed into pages. Text-formatting programs (such as DCF) can produce composed text data consisting entirely of structured fields.

N

NCP. See network control program.

NetSpool. A component of Infoprint Server that provides the capability for an installation to automatically direct VTAM application data targeted for a network printer to the JES spool without changing the VTAM applications. From the JES spool, the data can be printed on a JES or PSF printer or sent to another location for printing.

network control program (NCP). A program that controls the operation of a communication controller.

Network Print Facility (NPF). A feature that routes VTAM, JES2, or JES3 printer output to printers in a TCP/IP network.

Network Printer Manager (NPM). An application that lets network administrators monitor, control, and configure network printers. NPM also lets network administrators monitor some aspects of printers controlled by PSF and other manufacturers' network printers that comply with RFC 1759.

NPF. See Network Print Facility.

NPM. See Network Printer Manager.

O

output data set. A file that contains the results of processing.

output group. A group of output data sets that share certain characteristics, such as class and destination.

output writer. A part of the Job Entry Subsystem (JES) that receives job output from the system spool.

P

page definition. An Advanced Function Presentation (AFP) resource that defines the rules for transforming line data and XML data into Mixed Object Document Content Architecture-Presentation (MO:DCA-P) data and text controls.

partitioned data set (PDS). A data set on direct access storage that is divided into partitions, called members, each of which can contain a program, part of a program, or data. See also sequential data set.

path name. A name that specifies all directories leading to a file plus the file name itself.

PDS. See partitioned data set. See also sequential data set.

picture element (pixel). The smallest printable or displayable unit that can be displayed. A common measurement of device resolution is picture elements per inch.

PIDU. See Printer Inventory Definition Utility.

pixel. See picture element.

PLU. See primary logical unit. See also secondary logical unit.

PostScript. A page description language developed by Adobe Systems, Incorporated, that describes how text and graphics are presented on printers and display devices.

primary logical unit (PLU). In SNA, the logical unit that contains the primary half-session for a particular logical unit-to-logical unit (LU-to-LU) session. See also secondary logical unit.

Print Interface. A component of Infoprint Server that accepts input from remote TCP/IP workstations or from UNIX System Services printing commands. It creates output data sets on the JES spool.

print job. One or more documents submitted in the same job to be printed on the same printer.

print queue. A list of print jobs waiting to be printed.

Print Services Facility™ (PSF). An IBM licensed program that manages and controls the input data stream and output data stream required by supported page printers.

printer definition. An object in the Printer Inventory that contains information about a printer or set of printers that share the same characteristics. A printer definition contains information that is used to print files.

Printer Inventory. In Infoprint Server, a set of files that contain information about printers. The Printer Inventory includes such objects as printer definitions, functional subsystem (FSS) definitions, and job-selection rules for IP PrintWay.

Printer Inventory Definition Utility (PIDU). In Infoprint Server, a utility program that creates objects in the Printer Inventory.

PrintWay. See IP PrintWay.

PSF. See Print Services Facility.

R

RACF. See Resource Access Control Facility.

Request for Comments (RFC). In Internet communication, one of a series of numbered documents that describe Internet communication protocols.

request unit (RU). In SNA, a message unit that contains data, control information, or both (for example, data and indicators).

resource. A collection of printing instructions used by Print Services Facility (PSF), in addition to the print data set, to produce the printed output.

Resource Access Control Facility (RACF). An IBM licensed program that provides access control by identifying users to the system; verifying users of the system; authorizing access to protected resources; logging unauthorized attempts to enter the system; and logging accesses to protected resources.

response unit (RU). A message unit that acknowledges a request unit. It can contain prefix information received in a request unit.

Restructured Extended Executor (REXX). A general-purpose, high-level, programming language, particularly suitable for EXEC procedures or programs for personal computing.

retain time. In IP PrintWay, the length of time to keep a data set on the JES spool after a transmission.

retry limit. In IP PrintWay, the maximum number of retries that IP PrintWay is to attempt.

retry time. In IP PrintWay, the time between two attempts to send the data set to its destination.

REXX. See Restructured Extended Executor.

RFC. See Request for Comments.

root user. A system user who operates without restrictions. A root user has the special rights and privileges needed to perform administrative tasks.

RU. (1) See request unit. (2) See response unit.

S

SBCS. See single-byte character set. See also double-byte character set.

SCS. See SNA character string.

SDSF. See System Display and Search Facility.

secondary logical unit (SLU). In SNA, the logical unit (LU) that contains the secondary half-session for one logical unit-to-logical unit (LU-to-LU) session. See also primary logical unit.

sense code. A value sent or received, or a negative response to indicate what error occurred.

sequential data set. A data set whose records are organized on the basis of their successive physical positions, such as on magnetic tape. See also partitioned data set.

Server Message Block. A protocol that manages requests and responses in a client/server environment so that clients on a network can share files, directories, and devices.

session. A logical or virtual connection between two stations, programs, or devices on a network that allows the two elements to communicate and exchange data.

session parameters. In SNA, the parameters that specify or constrain the protocols (such as bracket protocol and pacing) for a session between two network accessible units (NAUs).

shell script. A program, or script, that is interpreted by the shell of an operating system.

Simple Network Management Protocol (SNMP). A set of protocols for monitoring systems and devices in complex networks. Information about managed devices is defined and stored in a Management Information Base (MIB).

single-byte character set (SBCS). A coded character set in which each character is represented by a 1-byte code. A 1-byte code point allows representation of up to 256 characters. See also double-byte character set.

SLU. See secondary logical unit. See also primary logical unit.

SME. See System Management Facility.

SNA. See Systems Network Architecture.

SNA character string (SCS). In SNA, a string of EBCDIC control characters carried within a request/response unit (RU); the string can also contain user data.

SNMP. See Simple Network Management Protocol.

SNMP agent. A server process that resides on a network node and is responsible for communicating with managers regarding that node. The node is represented as a managed object, which has various fields or variables that are defined in the appropriate MIB.

SNMP manager. A managing system that runs a managing application or suite of applications. These

applications depend on Management Information Base (MIB) objects for information that resides on the managed system. The SNMP manager generates requests for this MIB information, and an SNMP agent on the managed system responds to these requests.

spool. The system function of putting files or jobs into disk storage for later processing or printing.

startup procedure. A procedure used to start an application and to specify initialization parameters and libraries containing system resources.

superuser. See root user.

syntax. The rules for the construction of a command or statement.

System Display and Search Facility (SDSF). An IBM-licensed program that provides a menu-driven full-screen interface that is used to obtain detailed information about jobs and resources in a system.

System Management Facility (SMF). A z/OS facility that collects and records a variety of system and job-related information.

Systems Network Architecture (SNA). The description of the logical structure, formats, protocols, and operational sequences for transmitting information through and controlling the configuration and operation of networks.

T

table reference character (TRC). A numeric character corresponding to the order in which font character sets have been specified. The TRC is used to select a font character set during printing.

Telnet. In TCP/IP, a protocol that provides remote-terminal connection service. It allows users of one host to log on to a remote host and interact as if they were directly attached terminal users of that host.

Time Sharing Option (TSO). A base element of the z/OS operating system with which users can interactively work with the system.

trace. A record of the processing of a computer program or transaction. The information collected from a trace can be used to assess problems and performance.

transform. A program that converts a data stream from one format to another, for example, from PCL to AFP or PDF to AFP. Transforms provided by IBM are implemented as dynamic link library (DLL) filters.

transmission-queue data set. In IP PrintWay, a data set containing an entry for each data set that IP

PrintWay is to transmit to the remote system or that IP PrintWay is retaining on the Job Entry Subsystem (JES) spool.

transparent data. (1) Data that can contain any hexadecimal value. (2) A set of code points that is interpreted as presentable data, not as control data. In the AFP architecture, transparent data is identified with a control byte and a count of the number of bytes that follow.

trap. A message that reports a problem or a significant event.

TRC. See table reference character.

TSO. See Time Sharing Option.

U

UCS. See universal character set.

universal character set (UCS). A printer feature that permits the use of a variety of character arrays.

UNIX file. An object that exists in a hierarchical file system. Examples of UNIX files are files in an HFS, ZFS, NFS, and TFS.

UNIX filter. In UNIX operating systems, a program that obtains data from standard input (STDIN) and returns the results to standard output (STDOUT). See also DLL filter.

UNIX System Services. An element of z/OS that creates a UNIX environment that conforms to XPG4 UNIX 1995 specifications and that provides two open-system interfaces on the z/OS operating system: an application programming interface (API) and an interactive shell interface.

user port. In IP PrintWay, a port address that is outside the range of predefined addresses.

V

Virtual Telecommunications Access Method (VTAM). An IBM licensed program that controls communication and the flow of data in an SNA network.

VTAM. See Virtual Telecommunications Access Method.

VTAM application program. A program that has opened an access method control block (ACB) to identify itself to VTAM and that can therefore issue VTAM macroinstructions.

Z

zFS. See z/OS File System.

z/OS. An operating system for the IBM eServer™ product line that uses 64-bit real storage.

z/OS File System (zFS). A file system that can be used by z/OS UNIX System Services. For information, see *z/OS Distributed File Service zFS Administration*

Index

Special characters

- `_BPX_JOBNAME` environment variable 161, 405
- `_BPX_SHAREAS` environment variable 412
- `_BPX_UNLIMITED_OUTPUT` environment variable 412
- `_BPXK_SETIBMOPT_TRANSPORT` environment variable 133, 215, 412
- `_ICONV_MODE` environment variable 412
- `/etc/Printsrv` directory 30
- `/etc/Printsrv/aopd.conf`, creating 34
- `/etc/Printsrv/aopmsg.conf`, creating 86
- `/etc/Printsrv/aopsapd.conf`, creating 146
- `/etc/profile`, setting environment variables in 43
- `/etc/rc` file
 - `setstack` command 66
 - setting `STEPLIB` variable 29
 - starting Infoprint Server daemons 65
 - starting `sendmail` 234
- `/usr/lpp/Printsrv/bin`, specifying in `PATH` variable 46, 432
- `/var/Printsrv` directory 31
- `%` operator in filter options 122
- `&` in configuration files 42, 43, 148, 400, 406, 408
- `#` in configuration files 42, 43, 148, 400, 406, 408

Numerics

- 161 and 162 ports, unrestricting in TCP/IP data set 118
- 3270 data stream support in NetSpool 14
- 3270 data streams, NetSpool exits for 181
- 33E abend 57
- 515 port number
 - default 135
 - specified as default 135
 - unrestricting in TCP/IP data set 118
- 631 port number
 - specified as default 141, 143
 - unrestricting in TCP/IP data set 118

A

- abend D37 174, 319
- `AbortPrintFile` function 464
- accessibility 485
 - contact IBM 485
 - features 485
- address spaces, running NetSpool in multiple 175
- `AddType` directives, in HTTP Server 334
- administrator group, for security 50
- AFP Download Plus
 - Printer Inventory 4, 379
- AFP Printer Driver for Windows, overview 5
- AFP to PCL transform, customizing for PJJ functions 230
- AFP to PostScript transform, customizing for PJJ functions 230
- AFP Viewer plug-in for Windows, overview 6
- AIX, increasing time-out value 138
- aliases file, `sendmail` 234
- `allow-all-attributes-in-line-data` attribute 128, 195, 223, 435
- ANFDEAL sample JCL 318
- ANFMFILE program 174, 319
- ANFMIJCL sample JCL 319

- ANFMMSG DD statement 323
- ANFQINIT sample JCL 318
- ANFQUEUE DD statement 323
- ANFUBTBL, tables for block letters 298
- ANFUEXTP control block 248, 259, 270, 277, 285, 291
- ANFULINK sample JCL 300
- ANFUXRSP exit 292
- ANFUXxxx exits, sample 299
- ANFWPROC startup procedure 321
- AOP_64K_JOBID 412
- AOP_64K_JOBID environment variable 478
- AOP_ALLOW_ALL_CHARACTERS_IN_LINE_DATA environment variable 127, 194, 216, 413
- AOP_APPLID environment variable 217, 413
- AOP_BLANK_TRUNCATION_CLASSES environment variable 217
- AOP_IGNORE_DCF_ROUTING_ERRORS environment variable 210, 218, 414
- AOP_MAIL_DO_NOT_ADD_SUFFIX environment variable 219, 415
- AOP_MAIL_PRESERVE_SUFFIXES environment variable 219, 416
- AOP_MAIL_USE_FIRST_ADDRESS environment variable 220, 416
- AOP_MAXTHREADTASKS environment variable 417
- AOP_MVS_RETURN_CODES environment variable 418
- AOP_PJL environment variable 230
- AOP_READ_COMMUNITY environment variable 336, 419
- AOP_SELECT_WORK_FROM_HOLD_QUEUE environment variable 220, 420
- AOP_SUPPRESS_POST_UNIX_FILTER_FORMATTING environment variable 125, 221, 420
- AOP_WRITE_COMMUNITY environment variable 336, 421
- AOP.ADMINISTRATOR profile 48, 51, 344
- AOP.CONFIGURATION profile 49, 51
- AOP.DAEMON profile 362
- AOP.NETSPOOL profile 353
- AOP.PRIORITY profile 361
- AOP3201E, suppressing 210, 218
- AOPADMIN group 49, 50, 54
- AOPAPI 459
- AOPBATCH 455
 - examples 458
 - exit values 457
 - format 455
- AOPCENTRAL 421
- AOPCENTRAL environment variable 337
- AOPCENTRAL_CODEPAGE 422
- AOPCENTRAL_CODEPAGE environment variable 337
- AOPCONF environment variable 45, 337, 422
- AOPCONF group 49, 51
- AOPCPETC sample job 41, 86, 163, 166
- `aopd.conf`
 - attributes 389
 - creating 34
 - editing 42
 - naming in AOPCONF environment variable 45
- `aopfilt.so` 126
- AOPHINVD_MAXTHREADTASKS environment variable 422
- `aophinvd-max-thread-tasks` 435
- AOPINIT EXEC 67

AOPIPPD_MAXTHREADTASKS environment variable 422
 aopipdd-max-thread-tasks 436
 AOPLIMIT environment variable 337, 423
 AOPLIMIT_DOCS environment variable 337, 423
 AOPLIMIT_JOBS environment variable 338, 423
 AOPLPD_MAXTHREADTASKS environment variable 424
 aoplpd-max-thread-tasks 436
 AOPMSG_CONF environment variable 45, 425
 aopmsg.conf
 creating 86
 format 401
 AOPNETD_MAXTHREADTASKS environment variable 425
 AOPNETD_USE_FIXED_JOBID environment variable 425
 AOPNETD_USE_UNALTERED_JOBID environment variable 426
 aopnetd-max-thread-tasks 436
 aopoms command 149
 AOOPER group 49, 50, 54
 AOOPTIONS environment variable 130
 AOPOUTD_MAXTHREADTASKS environment variable 426
 aopoutd-max-thread-tasks 437
 AOPPATH environment variable 427
 AOPPATH variable 130
 AOPPRINT JCL procedure, customizing 130
 AOPRACF 50, 60, 61
 AOPRESUBMITUSER environment variable 427
 AOPRXF_CONF environment variable 167, 428
 aoprxf.conf
 creating 165
 format 165, 403
 aoprxf.so 126
 AOPSAPD_CONF environment variable 148, 428
 aopsapd.conf
 client attribute 147, 407
 creating 146
 example 148
 format 147, 407
 initial-target attribute 147, 407
 language attribute 147, 407
 naming in environment variable 148
 password attribute 147, 407
 syntax rules 148
 system attribute 147, 407
 userid attribute 147, 407
 AOPSDBD_MAXTHREADTASKS environment variable 428
 aopsdbd-max-thread-tasks 437
 aopsetup shell script 54
 AOPSSID_MAXTHREADTASKS environment variable 429
 aopssid-max-thread-tasks 438
 aopssid, starting 329
 aopstart EXEC, editing 57
 AOPSTART procedure 61
 AOPSTOP procedure 63
 AOPSUBD_MAXTHREADTASKS environment variable 429
 aopsubd-max-thread-tasks 438
 AOPTRACEON environment variable, setting in AOPSTART procedure 61
 AOPVALIDATEDB environment variable 45
 AOPVALIDATEDB environment variable, setting in AOPSTART procedure 61
 AOPWSMD_MAXTHREADTASKS environment variable 429
 aopwsmd-max-thread-tasks 438
 AOPXFD_CONF environment variable 167, 430
 aopxfd.conf
 creating 159
 environment attribute 160, 404
 example 162
 aopxfd.conf (*continued*)
 format 160, 404
 maximum-active attribute 161, 405
 maximum-idle-time attribute 161, 405
 minimum-active attribute 161, 405
 start-command attribute 160, 404
 start-daemons attribute 158
 transform attribute 160, 404
 APF-authorizing C++ applications that use API 477
 APF-authorizing NetSpool exits 182
 APF-authorizing SYS1.IMAGELIB 230, 301
 API 459
 APIJPJCL startup procedure for NetSpool 178
 APIJPTD1 193
 APIJPTD2 193
 APIMIJCL sample JCL 174
 APIMMSG DD statement 179
 APIPP1X2 control block 183
 APIPP1X2 macro 193
 APIPPTD1 exit, NetSpool 185
 APIPPTD2 exit, NetSpool 189
 APIUBF3 exit, NetSpool 187
 APIUGEX exit, NetSpool 192
 APPL ID, specifying for IP PrintWay basic mode 312
 APPL statement for IP PrintWay, creating 236
 Application Programming Interface 459
 applid attribute 224, 438
 ascii-codepage attribute 37, 76, 439
 assistive technologies 485
 attributes, job, specifying in environment variable 130
 AUDIT parameter, RACF 362
 AUTH parameter 237
 authorizing users to Printer Inventory, common message log, and operator commands 47
 AUTOLOG statement in TCP/IP 118
 automation, message 84
 AUTOSTOP parameter (JES2) 306

B

backend option, AIX 138
 backing up the Printer Inventory 105
 banner page, adding in exit 258, 276
 banner page, controlling in sample exit 300
 base-directory attribute 34
 Begin Data Set exit, IP PrintWay
 examples 268
 input 259
 installing 300
 modifying 301
 output 265
 overview 258
 programming considerations 242
 sample 299
 BeginEnumJobs function 464
 BeginEnumPrinters function 465
 Beginning of File exit for 3270 data, NetSpool (APIUBF3)
 description 187
 examples 189
 input 188
 installing 194
 modifying 194
 output 189
 Beginning of File exit for SCS data, NetSpool (APIPPTD1)
 description 185
 examples 187
 input 186

Beginning of File exit for SCS data, NetSpool (APIPTD1)
(continued)
 installing 194
 modifying 194
 output 186
 programming considerations 182
 sample 193

binary data support in NetSpool 14

blank truncation 127
 IP PrintWay extended mode 217

blank-truncation-classes attribute 224, 439

BLNKTRNC parameter (OUTCLASS) 127

Block Letter routine
 description of 298
 input 298
 link-editing with exit 300
 output 299
 sample exit for 300
 tables (ANFUBTBL) 298

BPX.SMF profile 215

BPXPRMxx member, updating
 for IP PrintWay basic mode 301
 for IPP Server 140
 for NetSpool 172
 for Print Interface 119
 MAXPROCUSER statement for IP PrintWay 207
 MAXSOCKETS parameter 28
 STEPLIBLIST facility, for SCEERUN and SCLBDLL
 libraries 29

broadcasting data 14

browsers, customizing for Infoprint Central 365

BufferSizeRequired function 465

C

C++ runtime options, changing in IP PrintWay 324

C++ runtime options, changing in NetSpool 181

callable services 459

CancelJob function 466

carriage control characters
 input to Begin Data Set exit 262
 input to End Data Set exit 280
 input to Routing exit 251

CARRIAGE parameter 304

cfilter.c 123

checklist of customization tasks 383

CICS, customization required to print from 25

class
 changing in exit 247
 CLASS parameter (JES2) 307
 QUEUE parameter (JES2) 307
 specifying as JES2 work-selection criteria 307, 308
 specifying as JES3 work-selection criteria 311
 WC parameter (JES3) 311

classes for transforms 159

CLASSPATH environment variable 142, 338, 430

client attribute 147, 407

ClosePrintFile function 466

Coaxial Printer Support 205

code pages
 custom code pages, specifying in configuration file 37, 38,
 76, 77
 default ASCII code page in configuration file 37, 76
 default EBCDIC code page in configuration file 38, 77
 defaults for IP PrintWay basic mode 313
 for Infoprint Central 337
 Japanese ISPF panels 66

coexistence, regarding Printer Inventory 27

COMMNDxx parmlib member 66

common message log
 allocating space 33
 MVS system logger 88
 NetSpool 174
 reducing size of 111
 saving 110
 security for 47

common message log, overview 4

community name for Infoprint Central 336, 421

community name for SNMP 375, 376

compatibility filter 136

configuration file aopd.conf
 attributes 389
 creating 34
 editing 42
 naming in AOPCONF environment variable 45

configuration file aopmsg.conf
 creating 86
 format 401

configuration file aoprxf.conf
 creating 165
 format 165, 403

configuration file aopsapd.conf
 client attribute 147, 407
 creating 146
 example 148
 format 147, 407
 initial-target attribute 147, 407
 language attribute 147, 407
 naming in environment variable 148
 password attribute 147, 407
 syntax rules 148
 system attribute 147, 407
 userid attribute 147, 407

configuration file aopxfd.conf
 creating 159
 environment attribute 160, 404
 example 162
 format 160, 404
 maximum-active attribute 161, 405
 maximum-idle-time attribute 161, 405
 minimum-active attribute 161, 405
 start-command attribute 160, 404
 start-daemons attribute 158
 transform attribute 160, 404

CONSNAME parameter, for NetSpool 179

console-name attribute 329, 363, 440

consoles, for NetSpool 176, 179

CONSTD statement 366

control file first, sending with LPR 137

COPIES parameter, changing in exit 247

cp command, example of 146

CreatePrintFile function 467

customizing
 AOPPRINT JCL procedure 130
 Infoprint Central 327
 Infoprint Port Monitor for Windows 369
 IP PrintWay 199
 LPD 132
 NetSpool 171
 Print Interface 115
 Print Interface IPP Server 140
 Print Interface SAP R/3 OMS 144
 Printer Inventory Manager 27
 PSF 379

- customizing (*continued*)
 - SNMP manager 377
 - SNMP subagent 373
 - Transform Interface 157
 - Transform Manager 157
 - Windows client 369
 - z/OS SNMP agent 374

D

- D37 abend 174, 319
- daemons to start 37, 134, 141, 154, 158, 173, 206, 375
- DATASETPREFIX statement 233
- DBCS, customizing for
 - default ASCII code page, specifying 37, 76
 - default EBCDIC code page, specifying 38, 77
- default printer, specifying 130, 433
- destination
 - changing in exit 247
 - DGROUP parameter (JES3) 310
 - ROUTECDDE parameter (JES2) 308
 - specifying as JES2 work-selection criteria 308
 - specifying as JES3 work-selection criteria 311
- DEVICE statement (JES3) 310
- DGROUP parameter (JES3) 310
- diagram of system flow 6
- directories for Infoprint Server 30
- directory name for files, specifying in configuration file 34
- DLLs, specifying directory for 46, 148, 432
- DNS
 - defining printers 231
 - email protocol 234
 - lookup order 232
 - resolve-printway-printers attribute 205
- document code page, changing default 37, 76
- document header
 - Begin data set exit 258
 - example 268
 - XTP_DOC_HEADER_LENGTH field 265
 - XTP_DOC_HEADER_PTR field 264
 - XTP_TRANS_DOC_HEADER field 262
- document trailer
 - Begin data set exit 258
 - example 268
 - XTP_DOC_TRAILER_LENGTH field 265
 - XTP_DOC_TRAILER_PTR field 265
 - XTP_TRANS_DOC_TRAILER field 262
- DocumentInfo data structure 460
- domain name server (DNS)
 - defining printers 231
 - email protocol 234
 - lookup order 232
 - resolve-printway-printers attribute 205
- downloading Infoprint Port Monitor 369
- DTYPE parameter (JES3) 310
- dynamic attributes 69
- dynamic configuration, enabling 68
- dynamic enablement of Infoprint Server 28
- DYNAMIC parameter (JES3) 310
- dynamic-configuration attribute 34

E

- ebcdic-codepage attribute 38, 77, 441
- EC6 abend 57
- echo command, example of 44

- email addresses for multi-document print jobs 220, 416
- email protocol, IP PrintWay
 - changing email address in Routing exit 257
 - customization tasks specific to protocol 201
 - environment variables 323
 - sendmail customization 234
- enabling Infoprint Server 28
- End Data Set exit, IP PrintWay
 - examples 284
 - input 277
 - installing 300
 - modifying 301
 - output 283
 - overview 276
 - programming considerations 242
 - sample 299
- EndEnumJobs function 468
- EndEnumPrinters function 468
- EnumJobs function 468
- EnumPrinters function 469
- environment attribute 160, 404
- environment variables
 - _BPX_JOBNAME 161, 405
 - _BPX_SHAREAS variable 412
 - _BPX_UNLIMITED_OUTPUT variable 412
 - _BPXK_SETIBMOPT_TRANSPORT variable 133, 215, 412
 - _ICONV_MODE variable 412
 - AOP_64K_JOBID variable 412, 478
 - AOP_ALLOW_ALL_CHARACTERS_IN_LINE_DATA variable 127, 194, 216, 413
 - AOP_APPLID variable 217, 413
 - AOP_BLANK_TRUNCATION_CLASSES variable 217
 - AOP_IGNORE_DCF_ROUTING_ERRORS variable 218, 414
 - AOP_MAIL_DO_NOT_ADD_SUFFIX variable 219, 415
 - AOP_MAIL_PRESERVE_SUFFIXES variable 219, 416
 - AOP_MAIL_USE_FIRST_ADDRESS variable 220, 416
 - AOP_MAXTHREADTASKS variable 417
 - AOP_MVS_RETURN_CODES variable 418
 - AOP_READ_COMMUNITY variable 336, 419
 - AOP_SELECT_WORK_FROM_HOLD_QUEUE variable 220, 420
 - AOP_SUPPRESS_POST_UNIX_FILTER_FORMATTING variable 221, 420
 - AOP_WRITE_COMMUNITY variable 336, 421
 - AOPCENTRAL variable 337, 421
 - AOPCENTRAL_CODEPAGE variable 337, 422
 - AOPCONF variable 45, 337, 422
 - AOPHINVD_MAXTHREADTASKS variable 422
 - AOPIPPD_MAXTHREADTASKS variable 422
 - AOPLIMIT variable 337, 423
 - AOPLIMIT_DOCS variable 337, 423
 - AOPLIMIT_JOBS variable 338, 423
 - AOPLPD_MAXTHREADTASKS variable 424
 - AOPMAILER variable 221, 424
 - AOPMAILER_OPTIONS variable 222, 424
 - AOPMSG_CONF variable 45, 425
 - AOPNETD_MAXTHREADTASKS variable 425
 - AOPNETD_USE_FIXED_JOBID variable 425
 - AOPNETD_USE_UNALTERED_JOBID variable 426
 - AOPTIONS variable 130, 426
 - AOPOUTD_MAXTHREADTASKS variable 426
 - AOPPATH variable 130, 427
 - AOPRESUBMITUSER variable 427
 - AOPRFX_CONF variable 167, 428
 - AOPSAPD_CONF variable 148, 428
 - AOPSDBD_MAXTHREADTASKS variable 428

- environment variables (*continued*)
 - AOPSSID_MAXTHREADTASKS variable 429
 - AOPSUBD_MAXTHREADTASKS variable 429
 - AOPVALIDATEDB variable 45
 - AOPWSMD_MAXTHREADTASKS variable 429
 - AOPXFD_CONF variable 167, 430
 - CLASSPATH variable 142, 338, 430
 - dynamic attributes, equivalent 70
 - example of displaying with echo command 44
 - examples of setting with export command 44
 - for IP PrintWay 323
 - for IP PrintWay extended mode 215
 - for IPP Server 142
 - for NetSpool 194
 - for Print Interface 127
 - for Print Interface IPP Server 142
 - for Printer Inventory Manager 43
 - for SAP OMS 148
 - for Transform Interface 167
 - for transforms 160, 404
 - JAVA_HOME variable 142, 338, 431
 - LANG variable 45, 339, 431
 - LC_ALL variable 45, 339, 431
 - LC_CTYPE variable 45, 431
 - LC_TIME variable 46, 339, 432
 - LIBPATH variable 46, 148, 339, 432
 - LPDEST variable 130, 433
 - MANPATH variable 46, 432
 - NLSPATH variable 46, 339, 432
 - PATH variable 46, 432
 - PRINTER variable 130, 433
 - setting in aopstart EXEC 57
 - STEPLIB variable 29, 216, 433
 - summary of 409
 - TZ variable 47, 340, 433
- ErrorNumber function 470
- ErrorString function 471
- examples
 - Begin Data Set exit 268
 - End Data Set exit 284
 - JES2 initialization 306
 - JES3 initialization 309
 - Message exit 292
 - Record exit 275
 - Routing exit 257
 - SAP callback daemon configuration file 148
 - SMF exit 290
 - transform configuration file 162
- exit programs 181, 237
 - APF-authorizing NetSpool exits 182
 - APIPP1X2 control block 183
 - filter programs 182, 238
 - installing IP PrintWay exits 300
 - installing NetSpool exits 194
 - IP PrintWay Begin Data Set exit 258
 - IP PrintWay End Data Set exit 276
 - IP PrintWay Message exit 290
 - IP PrintWay Record exit 269
 - IP PrintWay Response Notification exit 292
 - IP PrintWay Routing exit 247
 - IP PrintWay SMF exit 285
 - modifying IP PrintWay exits 301
 - modifying NetSpool exits 194
 - NetSpool Beginning of File exit for 3270 data streams 187
 - NetSpool Beginning of File exit for SCS data 185
 - NetSpool Graphic Escape exit 192
 - NetSpool Transparent Data Control exit 189

- exit programs (*continued*)
 - order of, IP PrintWay 238
 - Print Interface filters 121
 - programming considerations, IP PrintWay 242
 - programming considerations, NetSpool 182
 - sample IP PrintWay exits 299
 - sample NetSpool exits 193
 - STEPLIB for NetSpool exits 179
- exploiting new functions 483
- export command, example of 44
- exporting the Printer Inventory 105

F

- FCB parameter (JES2) 307
- FCB work-selection criterion 311
- FCB, APF authorization for SYS1.IMAGELIB 230, 301
- FCB, default for IP PrintWay (JES3) 311
- filter programs
 - examples of % operator 122
 - Infoprint Server filters 126
 - input 123, 124
 - LPD compatibility 136
 - modifying 126
 - options for 122
 - output 124, 125
 - programming considerations 125
 - REXX examples 122
 - sample 126
 - source code, sample 123
 - testing 126
 - writing a DLL filter 123
 - writing a UNIX filter 124
- firewall for remote transforms 168
- firewalls, email considerations 234
- fixed job ID, NetSpool 425, 451
- font, default for IP PrintWay (JES2) 304, 308
- font, default for IP PrintWay (JES3) 304, 311
- form name
 - changing in exit 247
 - FORMS parameter (JES2) 307
 - specifying as JES2 work-selection criteria 308
 - specifying as JES3 work-selection criteria 311
- FSA definitions for IP PrintWay 315
- FSS and FSA definitions, IP PrintWay 312
- FSS definitions for IP PrintWay 312
- FSS initialization statement (JES2) 306
- FSS initialization statement (JES3) 309
- FSSNAME parameter (JES3) 309, 310
- functional subsystem (FSS), IP PrintWay
 - defining multiple FSAs 303
 - defining multiple FSSs 302
 - JES2 initialization statements 306
 - JES3 initialization 309
 - name, specifying in JES2 306
 - name, specifying in JES3 309
 - planning 302
- functional subsystem application (FSA), IP PrintWay
 - determining work-selection criteria 303
 - how many IP PrintWay FSAs to create 303
 - IP PrintWay configuration information 315
 - name, specifying in JES2 306
 - name, specifying in JES3 310

G

- GetJobInfo function 471
- GetPrinterInfo function 472
- GF10 default font (JES2) 304, 308
- Graphic Exit, NetSpool (APIUGEX)
 - description 192
 - examples 193
 - input 192
 - installing 194
 - modifying 194
 - output 193
 - sample 193
- GS10 default font (JES3) 304

H

- hardcopy log 84
- hardcopy-message-list attribute 84, 401, 442
- hardcopy-messages attribute 84, 401, 442
- header, document
 - Begin data set exit 258
 - example 268
 - XTP_DOC_HEADER_LENGTH field 265
 - XTP_DOC_HEADER_PTR field 264
 - XTP_TRANS_DOC_HEADER field 262
- HFS directories 30
- hiperspace blocks for IP PrintWay basic mode 313
- Historical Inventory, overview 5
- HTTP Server, customizing 332

I

- IAZJSPA control block 243
- IBM-1047, specified as default 38, 77
- IDCAMS parameters to allocate transmission-queue data sets 318
- IFAPRDxx member of SYS1.PARMLIB 28
- ignore-dcf-routing-errors attribute 225, 443
- IMS, customization required to print from 25
- Infoprint Central
 - authorizing AOOPER group to JES commands 341
 - CONSTD statement 366
 - customizing 327
 - customizing PSF for z/OS 366
 - customizing web browsers 365
 - customizing z/OS HTTP Server 332
 - defining users to RACF 344
 - initializing the search database 331
 - RACF notification 362
 - security 341
 - setting up security for actions on PSF printers 349
 - setting up security for changing print job priority 361
 - setting up security for daemons 362
 - setting up security for IP PrintWay job-selection rules 354
 - setting up security for NetSpool logical units 353
 - setting up security for print jobs 356
 - setting up security for printer and printer pool definitions 362
 - setting up security for printers 345
 - software requirements 328
 - starting from the web 366
- Infoprint Central, overview 21
- Infoprint Port Monitor 369
- Infoprint Port Monitor compared to other printing protocols 369
- Infoprint Port Monitor for Windows, overview 5

- Infoprint Server configuration file aopd.conf
 - attributes 389
 - creating 34
 - editing 42
 - naming in AOPCONF environment variable 45
- InfoPrint Transform Manager for Linux
 - code page conversion table 169
 - configuration file 165
 - customization tasks 157
 - firewall 168
 - resource security 168
 - software requirement 158
- InitAPI function 473
- initial-target attribute 147, 407
- input data streams, types supported 13
- interfaces, programming 121
- INTERVAL parameter 118
- INV parameter in IP PrintWay startup procedure 323
- INV parameter in NetSpool startup procedure 179
- inventory attribute 35, 154
- Inventory, Printer
 - authorizing users 47
 - backing up 105
 - FSS and FSA definitions, IP PrintWay 312
 - naming in IP PrintWay startup procedure 323
 - naming in NetSpool startup procedure 179
 - restoring 105
 - sharing 27
- IP address, changing in exit 247
- IP PrintWay basic mode
 - APF-authorizing SYS1.IMAGELIB 301
 - APPL statement 236
 - authorizing to select data sets 214
 - comparing to extended mode 202
 - customizing 199
 - exits, writing 237
 - filters, writing 238
 - FSA definitions 315
 - FSA, defining to JES 302
 - FSS definitions 312
 - FSS, defining to JES 302
 - increasing allowed number of processes 301
 - Language Environment runtime options 324
 - migrating from basic mode to extended mode 479
 - sendmail, customizing 234
 - sharing AFP printers with PSF 324
 - software requirements 205
 - space requirements in hiperspace 312
 - startup procedure for 320
 - TCP/IP:DATA data set, customizing for 232
 - transmission-queue data set, allocating 316
- IP PrintWay extended mode
 - APF-authorizing SYS1.IMAGELIB 230
 - APPL statement 236
 - authorization to write SMF type 6 records 215
 - comparing to basic mode 202
 - customizing 199
 - environment variables 215
 - exits, writing 237
 - filters, writing 238
 - increasing allowed number of processes 207
 - job-selection rules 208
 - sendmail, customizing 234
 - setting up the DNS 231
 - sharing AFP printers with PSF 324
 - software requirements 205
 - space required in /var/Printsrv 32

- IP PrintWay extended mode (*continued*)
 - TCPIP.DATA data set, customizing for 232
 - temporary files in /var/Printsrv 32
- IP PrintWay, overview 15
- IPP protocol, IP PrintWay
 - customization tasks specific to protocol 201
 - environment variables 323
- IPP Server in Print Interface, customizing 140
 - environment variables for 142
 - IPP workstation client, customizing 143
 - specifying port number for 141, 143
 - starting daemon for 141
 - URI used by IPP workstation client 143
- ipp-port-number attribute 141, 143, 443, 445
- ISO8859-1, specified as default 37, 76
- ISPF panels, Infoprint Server
 - defining libraries in TSO logon procedure 66
 - displaying Infoprint Server option 67
 - enabling 66
 - modifying AOPINIT EXEC 67

J

- Japanese messages for IP PrintWay basic mode 313
- Japanese, ISPF panels 66
- Japanese, messages 66, 175
- Java software requirements 116, 328
- JAVA_HOME environment variable 142, 338, 431
- JES work-selection criteria
 - determining 303
 - specifying in JES2 308
 - specifying in JES3 311
- jes-name attribute 35
- JES2
 - FSS statement example 306
 - initialization statements 305
 - PRTnnnnn statement 306
- JES3
 - DEVICE statement 310
 - FSS statement 309
 - initialization statements 309
- JESNEWS data set
 - indicator in ANFUEXTP control block 247
 - suppressing SMF type 6 record for 285, 300
- JNAME parameter (JES3) 310
- job attributes, specifying in environment variable 130
- job control language (JCL)
 - sample to link-edit exit 300
 - startup, for Infoprint Server daemons 61
 - startup, IP PrintWay 322
 - startup, NetSpool 174
- job ID, NetSpool 425, 426, 451, 452
- job name for transforms, changing 161, 405
- job separator page data area (IAZJSPA) 243
- job-prefix attribute 120, 129, 173, 196, 444
- job-selection rules
 - creating 208
 - security 354
- JobInfo data structure 461
- JobToken data structure 461
- JSPA control block 243

K

- keyboard
 - navigation 485

- keyboard (*continued*)
 - PF keys 485
 - shortcut keys 485

L

- LANG environment variable 45, 339, 431
- language attribute 147, 407
- Language Environment runtime libraries 29
- Language Environment runtime options, changing in IP PrintWay 324
- Language Environment runtime options, changing in NetSpool 181
- language for IP PrintWay basic mode messages 312
- LANGUAGE parameter, in NetSpool startup procedure 179
- large files, sending to LPD 137
- LC_ALL environment variable 45, 339, 431
- LC_CTYPE environment variable 431
- LC_TIME environment variable 46, 339, 432
- LIBPATH environment variable 46, 148, 339, 432
- libraries, defining in TSO logon procedure (Print Interface) 66
- limit of objects Infoprint Central displays 337, 423
- limits on spooled output 119, 180
- Linux, Infoprint Transform Manager
 - code page conversion table 169
 - configuration file 165
 - customization tasks 157
 - firewall 168
 - resource security 168
 - software requirement 158
- log stream
 - deciding on naming convention 90
 - deciding to use 89
 - deciding type 89
 - defining coupling-facility type 96
 - defining DASD-only type 94
 - defining system logger 94
 - determining retention period 90
 - disabling 102
 - enabling 99
 - handling errors 103
 - parameters 95
 - security for 91
 - size of data sets 91
 - switching log streams 101
 - viewing messages 104
- log-retention attribute 39, 78, 330, 364, 444
- log.v2db files, deleting 112
- logon procedure for ISPF panels 66
- logstream-name attribute 445
- LOMS for SAP R/3
 - defining 148
 - modifying 149
 - selecting for an output device 150
- LOOKUP statement 232
- LPD compatibility filter 136
- LPD control file first, sending with LPR 137
- lpd_compat.so 136
- lpd-port-number attribute 135
- LPD, customizing 132
 - command codes supported in 137
 - running Communications Server LPD on same system 133
 - specifying port number for 135
 - starting daemon 134
- LPDEST environment variable 130, 433
- lprafp command, downloading 7

LUCCLASS parameter in NetSpool startup procedure 178

M

mail-do-not-add-suffixes attribute 226, 447
mail-preserve-suffixes attribute 226, 447
mail-use-first-address attribute 227, 448
mailer-options attribute 227, 448
mailer-path-name attribute 228, 449
man pages, specifying directory for 46, 432
MANPATH environment variable 46, 432
max-historical-inventory-size 40, 79, 449
max-thread-tasks 450
MAXASSIZE parameter 140
maximum-active attribute 161, 405
maximum-idle-time attribute 161, 405
MAXPROCSYS parameter 119, 172, 207, 301
MAXPROCUSER parameter 119, 172, 207, 301
MAXSOCKETS parameter 28
MAXTHREADTASKS limit
 AOP_MAXTHREADTASKS environment variable 417
 AOPHINVD_MAXTHREADTASKS environment variable 422
 AOPIPPD_MAXTHREADTASKS environment variable 422
 AOPLPD_MAXTHREADTASKS environment variable 424
 AOPNETD_MAXTHREADTASKS environment variable 425
 AOPOUTD_MAXTHREADTASKS environment variable 426
 AOPSDBD_MAXTHREADTASKS environment variable 428
 AOPSSID_MAXTHREADTASKS environment variable 429
 AOPSUBD_MAXTHREADTASKS environment variable 429
 AOPWSMD_MAXTHREADTASKS environment variable 429
MCS console 176
message automation 84
message configuration file, aopmsg.conf
 creating 86
 format 401
Message exit, IP PrintWay
 examples 292
 input 291
 installing 300
 modifying 301
 output 291
 overview 290
 programming considerations 242
 sample 299
message log, common
 allocating space 33
 MVS system logger 88
 NetSpool 174
 reducing size of 111
 saving 110
 security for 47
message-log data set, IP PrintWay
 allocating space 319
 defining in startup procedure 323
 initializing 319
message-log data set, NetSpool 174
messages, sending to hardcopy log 84
migration information xiv
migration program, overview 5

minimum-active attribute 161, 405
MODE parameter
 in DEVICE statement (JES3) 311
 in PRTnnnnn statement (JES2) 307
MSGDEST parameter (JES3) 309
multiple TCP/IP stacks 133
MVS system logger
 deciding on naming convention 90
 deciding to use 89
 deciding type 89
 deciding type of log stream 89
 defining coupling-facility log stream 96
 defining coupling-facility type 96
 defining DASD-only log stream 94
 defining DASD-only type 94
 defining log streams 94
 defining system logger 94
 determining retention period 90
 disabling 102
 enabling 99
 handling errors 103
 log-stream parameters 95
 migrating to 483
 parameters 95
 security for 91
 setting up 88
 size of data sets 91
 structure parameters 98
 switching log streams 101
 viewing messages 104

N

navigation
 keyboard 485
NetSpool
 customizing 171
 environment variables 194
 exits for transparent data, writing 181
 filters, writing 182
 increasing allowed number of processes 172
 job ID, fixed 425, 451
 job ID, unaltered 426, 452
 Language Environment runtime options 181
 LU classes, specified in startup procedure 178
 message logs 174
 startup procedure for 174
netspool-use-fixed-jobid 451
netspool-use-unaltered-jobid 452
NetSpool, overview 12
Network Printer Manager (NPM) for the Web,
 customizing 377
network printers, customizing PSF for 324
NETWORK statement 28
new functions 483
NIFCB parameter (JES2) 304, 306
NIUCS parameter (JES2) 304, 306
NLSPATH environment variable 46, 339, 432
Notices 487
NOTIFY parameter, RACF 362
NPM for the Web, customizing 377

O

operator group, for security 50
OPERLOG messages 84

- output devices for SAP R/3, creating 150
- OutputTimeout directive, in HTTP Server 335
- overview of Infoprint Server components 1
 - AFP Printer Driver for Windows 5
 - AFP Viewer plug-in for Windows 6
 - Infoprint Central 21
 - Infoprint Port Monitor for Windows 5
 - IP PrintWay 15
 - NetSpool 12
 - Print Interface 6
 - Printer Inventory 3
 - Printer Inventory Manager 3
 - Transform Interface 19
 - transforms 20
 - Windows client 5

P

- page definition default for IP PrintWay (JES2) 304, 307
- page definition default for IP PrintWay (JES3) 304, 311
- panels, ISPF
 - defining libraries in TSO logon procedure 66
 - displaying Infoprint Server option 67
 - enabling 66
 - modifying AOPINIT EXEC 67
- parameters
 - JES2 initialization statements 305
 - JES3 initialization statements 309
 - LANGUAGE parameter 179
 - TRACE parameter 179
- Pass directives, in HTTP Server 333
- password attribute 147, 407
- PATH environment variable 46, 432
- PDEFAULT parameter (JES3) 304, 311
- performance and the TRKCELL parameter (JES2) 308
- performance, improving in IP PrintWay extended mode 232
- permissions for directories and daemons 54
- PIDU, used to back up and restore Printer Inventory 105
- PJL functions in IP PrintWay
 - customizing Infoprint transforms 230
 - JOB command
 - AOP_PJL environment variable 230
- PNAME parameter (JES3) 309
- port number
 - for IPP Server 141, 143
 - for Print Interface LPD 135
 - used by LPR 137
- PORT statement in TCP/IP 118
- PORTRANGE statement in TCP/IP 118
- ports, low 118
- prefix for job ID 120
- prefix for job identifier 129, 173, 196
- PRESELECT parameter (JES2) 307
- Print Interface
 - AOPPRINT JCL procedure, customizing 130
 - common customization required for all functions 116
 - configuration attributes 134
 - customizing 115
 - environment variables 127
 - environment variables for z/OS UNIX commands 130
 - filters, writing 121
 - increasing allowed number of processes 119
 - IPP Server customization 140
 - JES customization 127
 - LPD customization 132
 - SAP customization 144
 - software requirements 116
- Print Interface (*continued*)
 - subsystem customization 153
 - TCP/IP customization 117
- Print Interface, overview 6
- print queue, changing in exit 247
- PRINTDEF statement (JES2) 304, 306
- printer code page, changing default 38, 77
- PRINTER environment variable 130, 433
- Printer Inventory
 - authorizing users 47
 - backing up 105
 - FSS and FSA definitions, IP PrintWay 312
 - naming in IP PrintWay startup procedure 323
 - naming in NetSpool startup procedure 179
 - restoring 105
 - sharing 27
- Printer Inventory Manager
 - backing up and restoring Printer Inventory 105
 - creating aopd.conf file 34
 - creating aopmsg.conf file 86
 - creating startup and shutdown procedures for 56
 - customizing 27
 - directories 30
 - dynamic enablement of Infoprint Server 28
 - enabling ISPF panels for 66
 - establishing security for 47
 - making runtime libraries available for 29
 - saving the common message log 110
 - setting environment variables for 43
 - setting Workload Manager goals 112
 - verifying software requirements 28
- Printer Inventory Manager, overview 3
- Printer Inventory, overview 3
 - printer setup string
 - adding in a filter program 121
 - printer setup string, adding in exit 258
 - printer sharing, customizing PSF for 324
 - printer, setting a default 130, 433
 - PrinterInfo data structure 461
 - PrintWay, overview 15
 - PROC parameter (JES2) 306
 - procedure, startup
 - for Infoprint Server daemons 61
 - IP PrintWay 320
 - sample 322
 - specifying name of in JES2 306
 - specifying name of in JES3 309
 - NetSpool 174
 - sample 177
 - profile, TCP/IP 117
 - PROFILE.TCPIP data set, modifying for SNMP subagent 374
 - programming considerations for filter programs 125
 - programming considerations for IP PrintWay exits 242
 - programming considerations for NetSpool exit programs 182
 - programming interfaces 121
 - Protect directive, in HTTP Server 334
 - PRTnnnnn initialization statement (JES2) 306
 - PRTOPTNS parameter, changing in exit 247
 - PRTQUEUE parameter, changing in exit 247
 - PS, default job prefix 173
 - PSF for z/OS
 - AFP printer driver for Windows for creating AFP data 5
 - AFP viewer for viewing AFP data 6
 - customizing for printer sharing 324
 - customizing for use with Infoprint Central 366
 - customizing SNMP subagent to monitor PSF printers 373
 - customizing to use Printer Inventory 379

- PSF for z/OS (*continued*)
 - enabling SNMP reporting 376
 - Infoprint Central functions 21
 - Infoprint Server components to customize 25
 - migration program 5
 - monitoring printers with SNMP subagent 22
 - Printer Inventory 4, 379
 - roadmap of components to customize 25
 - setting up security for PSF printers 345
 - transforms from AFP on z/OS 20
 - transforms to AFP on z/OS 20
- PSF bucket xiv
- PW.SRC file 374

Q

- QueryToken data structure 462
- QUEUE parameter (JES2) 307

R

- RACF
 - AOP.ADMINISTRATOR profile 48, 344
 - AOP.CONFIGURATION profile 49
 - AOP.DAEMON profile 362
 - AOP.NETSPOOL profile 353
 - AOP.PRIORITY profile 361
 - AOPADMIN group 49
 - AOPCONF group 49
 - AOPOPER group 49
 - aopsetup shell script 54
 - auditing, requesting 362
 - notification, requesting 362
 - PRINTSRV class 48, 49
 - security for actions on PSF printers 349
 - security for changing print job priority 361
 - security for common message log 47
 - security for daemons 362
 - security for Infoprint Central 341
 - security for IP PrintWay job-selection rules 354
 - security for IP PrintWay transmission-queue data set 319
 - security for NetSpool logical units (LUs) 353
 - security for operator commands 47
 - security for print jobs 356
 - security for printer and printer pool definitions 362
 - security for Printer Inventory 47
 - security for printers 345
- Record exit, IP PrintWay
 - examples 275
 - input 270
 - installing 300
 - modifying 301
 - output 274
 - overview 269
 - sample 299
- record format
 - input to Begin Data Set exit 262
 - input to End Data Set exit 280
 - input to Routing exit 251
- region size, IP PrintWay 323
- region size, NetSpool 178
- registers
 - for Begin Data Set exit 259
 - for Beginning of File exit 186, 188, 192
 - for Block-Letter program 298
 - for End Data Set exit 277
- registers (*continued*)
 - for Message exit 291
 - for Record exit 270
 - for SMF exit 285
 - for the Routing exit 248
 - for Transparent Data exit 190
- ReleaseTransform function 473
- rembak command, AIX 138
- remote transform configuration file, aopxfd.conf
 - creating 165
 - format 165, 403
- remote-transform-manager-ip-address attribute 165, 403
- remote-transform-manager-port-number attribute 166, 403
- RequestTransform function 474
- requirements, software
 - Infoprint Central 328
 - IP PrintWay 205
 - NetSpool 172
 - Print Interface 116
 - Printer Inventory Manager 28
 - PSF for z/OS 379
 - SNMP subagent 373
 - Transform Interface 158
- resolve-printway-printers attribute 205
- Resource Access Control Facility (RACF)
 - AOP.ADMINISTRATOR profile 48, 344
 - AOP.CONFIGURATION profile 49
 - AOP.DAEMON profile 362
 - AOP.NETSPOOL profile 353
 - AOP.PRIORITY profile 361
 - AOPADMIN group 49
 - AOPCONF group 49
 - AOPOPER group 49
 - aopsetup shell script 54
 - auditing, requesting 362
 - notification, requesting 362
 - PRINTSRV class 48, 49
 - security for actions on PSF printers 349
 - security for changing print job priority 361
 - security for common message log 47
 - security for daemons 362
 - security for Infoprint Central 341
 - security for IP PrintWay job-selection rules 354
 - security for IP PrintWay transmission-queue data set 319
 - security for NetSpool logical units (LUs) 353
 - security for operator commands 47
 - security for print jobs 356
 - security for printer and printer pool definitions 362
 - security for Printer Inventory 47
 - security for printers 345
- resources, security for 168
- Response Notification exit 292
- restoring the Printer Inventory 105
- restrict ports 118
- resubmit for filtering
 - AOPRESUBMITUSER environment variable 427
 - customizing IP PrintWay to use font and page definition in printer definition 305
 - exit processing, description 240
 - specifying for transforms 158
 - suppressing SMF record 285
- RETAINF parameter, changing in exit 247
- RETAINS parameter, changing in exit 247
- RETRYL parameter, changing in exit 247
- RETRYT parameter, changing in exit 247
- REXX examples, for filters 122
- RFC 1179, LPD command codes supported 137

- roadmap of components to customize 25
- ROMS for SAP R/3, defining 148
- ROUTECD parameter (JES2) 308
- routing criteria, changing in exit 247
- Routing exit, IP PrintWay
 - examples 257
 - input 248
 - installing 300
 - modifying 301
 - output 254
 - overview 247
 - programming considerations 242
 - sample 299
- runtime libraries for Language Environment 29
- runtime options for C++ Language Environment , changing in IP PrintWay 324
- runtime options for C++ Language Environment , changing in NetSpool 181

S

- S2STG DSECT 183
- SACONFIG statement 374
- SAF (system authorization facility)
 - AOP.ADMINISTRATOR profile 48, 344
 - AOP.CONFIGURATION profile 49
 - AOP.DAEMON profile 362
 - AOP.NETSPOOL profile 353
 - AOP.PRIORITY profile 361
 - AOPADMIN group 49
 - AOPCONF group 49
 - AOOPER group 49
 - aopsetup shell script 54
 - auditing, requesting 362
 - notification, requesting 362
 - PRINTSRV class 48, 49
 - security for actions on PSF printers 349
 - security for changing print job priority 361
 - security for common message log 47
 - security for daemons 362
 - security for Infoprint Central 341
 - security for IP PrintWay job-selection rules 354
 - security for IP PrintWay transmission-queue data set 319
 - security for NetSpool logical units (LUs) 353
 - security for operator commands 47
 - security for print jobs 356
 - security for printer and printer pool definitions 362
 - security for Printer Inventory 47
 - security for printers 345
- sample exits 299
- SAP Callback daemon configuration file, aopsapd.conf
 - client attribute 147, 407
 - creating 146
 - example 148
 - format 147, 407
 - initial-target attribute 147, 407
 - language attribute 147, 407
 - naming in environment variable 148
 - password attribute 147, 407
 - syntax rules 148
 - system attribute 147, 407
 - userid attribute 147, 407
- SAP R/3 printing
 - configuration file 146
 - configuration file for Callback daemon 146
 - creating an SAP R/3 output device 150
 - customization tasks 144
 - SAP R/3 printing (*continued*)
 - environment variable for configuration file 148
 - ROMS and LOMS for 148
 - software requirements 116
 - user for Callback daemon 146
 - SAP R/3 printing, overview 9, 11
 - SCEERUN library 29
 - SCEERUN2 library 29
 - scheduler work block text unit (SWBTU) 244
 - SCLBDLL library 29
 - SCS data stream support in NetSpool 14
 - SCS data streams, NetSpool exits for 181
 - sdbu command 331
 - search database, initializing 331
 - search database, recovering 331
 - security, establishing
 - AOP.ADMINISTRATOR profile 48, 344
 - AOP.CONFIGURATION profile 49
 - AOP.DAEMON profile 362
 - AOP.NETSPOOL profile 353
 - AOP.PRIORITY profile 361
 - AOPADMIN group 49
 - AOPCONF group 49
 - AOOPER group 49
 - aopsetup shell script 54
 - auditing, requesting 362
 - notification, requesting 362
 - PRINTSRV class 48, 49
 - security for actions on PSF printers 349
 - security for changing print job priority 361
 - security for common message log 47
 - security for daemons 362
 - security for Infoprint Central 341
 - security for IP PrintWay job-selection rules 354
 - security for IP PrintWay transmission-queue data set 319
 - security for NetSpool logical units (LUs) 353
 - security for operator commands 47
 - security for print jobs 356
 - security for printer and printer pool definitions 362
 - security for Printer Inventory 47
 - security for printers 345
 - select-work-from-hold-queue attribute 228, 452
 - sendmail
 - creating an alias file 234
 - customizing 234
 - environment variables (basic mode) 324
 - environment variables (extended mode) 222
 - firewalls 234
 - specifying alternate directory 234
 - specifying options 235
 - separator page, adding in exit 258, 276
 - Server Message Block (SMB), customizing Print Interface support for 138
 - Server Message Block compared to other printing protocols 369
 - ServerInit directive, in HTTP Server 333
 - ServerInit directives, in HTTP Server 333
 - Service directives, in HTTP Server 333
 - setstack command 66
 - SetTerminationHandler function 475
 - SETUP parameter (JES2) 307
 - setup string, adding in exit 258
 - setup strings for printers
 - adding in a filter program 121
 - shared printers, SMB 139
 - sharing IP PrintWay transmission-queue data set 316
 - sharing of printers, customizing PSF for 324

- shortcut keys 485
- shutdown procedure, for Infoprint Server daemons 63
- SMB compared to other printing protocols 369
- SMB support, software requirements 116
- SMB, customizing Print Interface support for 138
- SMF exit, IP PrintWay
 - examples 290
 - input 285
 - installing 300
 - modifying 301
 - output 289
 - overview 285
 - programming considerations 242
 - sample 299
- SMF record
 - replacing in exit 285
 - suppressing in exit 285
- SMF type 6 record enhancements 484
- smf-recording attribute 206, 228, 452
- SMTP 234
- SNMP
 - read community name for Infoprint Central 336
 - subagent, customizing 373
 - write community name for Infoprint Central 336
 - z/OS agent, customizing 374
- SNMP subagent, overview 22
- SNMP write community name 421
- snmp-community attribute 375, 376, 453
- sockets, increasing maximum 28
- software requirements
 - Infoprint Central 328
 - IP PrintWay 205
 - NetSpool 172
 - Print Interface 116
 - Printer Inventory Manager 28
 - PSF for z/OS 379
 - SNMP subagent 373
 - Transform Interface 158
- special characters in configuration files 42, 43, 148, 400, 406, 408
- SPOOLDEF statement (JES2) 304, 305
- stacks, multiple TCP/IP 133
- START parameter (JES2) 308
- start-command attribute 160, 404
- start-daemons attribute 37, 134, 141, 154, 158, 173, 206, 329, 375
- starting daemons, specifying names in configuration file 37, 134, 141, 154, 158
- startup procedure
 - for Infoprint Server daemons 61
 - for IP PrintWay 320
 - sample 322
 - specifying name of in JES2 306
 - specifying name of in JES3 309
 - for NetSpool 174
 - sample 177
 - Printer Inventory Manager 61
- STDENV data definition (DD) 62, 323, 409
- STEPLIB environment variable 29, 216, 300, 433
- STEPLIBLIST facility 29, 207, 216
- structure parameters 98
- subsystem, Print Interface
 - customizing 153
- suffixes for email attachments 219, 226, 415, 416, 447
- suppress-post-unix-filter-formatting attribute 229, 453
- SWBTU control block 244
- syntax rules
 - for SAP Callback configuration file 148
- SYS1.IMAGELIB 216, 230, 301
- SYS1.PARMLIB 28
- SYS1.PARMLIB, updating
 - for IP PrintWay basic mode 301
 - for IPP Server 140
 - for NetSpool 172
 - for Print Interface 119
 - MAXPROCUSER statement for IP PrintWay 207
 - MAXSOCKETS parameter 28
 - STEPLIBLIST facility, for SCEERUN and SCLBDLL libraries 29
- SYSLOG messages 84
- SYSOUT statement (JES3) 309
- sysplex considerations
 - IP PrintWay extended mode 203
 - JES3 limitation 210
 - Printer Inventory 27
- SYSTCPD DD statement 323
- system attribute 147, 407
- system authorization facility (SAF)
 - AOP.ADMINISTRATOR profile 48, 344
 - AOP.CONFIGURATION profile 49
 - AOP.DAEMON profile 362
 - AOP.NETSPOOL profile 353
 - AOP.PRIORITY profile 361
 - AOPADMIN group 49
 - AOPCONF group 49
 - AOPOPER group 49
 - aopsetup shell script 54
 - auditing, requesting 362
 - notification, requesting 362
 - PRINTSRV class 48, 49
 - security for actions on PSF printers 349
 - security for changing print job priority 361
 - security for common message log 47
 - security for daemons 362
 - security for Infoprint Central 341
 - security for IP PrintWay job-selection rules 354
 - security for IP PrintWay transmission-queue data set 319
 - security for NetSpool logical units (LUs) 353
 - security for operator commands 47
 - security for print jobs 356
 - security for printer and printer pool definitions 362
 - security for Printer Inventory 47
 - security for printers 345
- system configuration definition
 - attributes 435
 - editing 80
 - ISPF panel for 80
- system flow diagram 6
- system logger
 - deciding on naming convention 90
 - deciding to use 89
 - deciding type of log stream 89
 - defining coupling-facility log stream 96
 - defining DASD-only log stream 94
 - defining log streams 94
 - determining retention period 90
 - disabling 102
 - enabling 99
 - handling errors 103
 - log-stream parameters 95
 - migrating to 483
 - security for 91
 - setting up 88

system logger (*continued*)
size of data sets 91
structure parameters 98
switching log streams 101
viewing messages 104
SYSTEM parameter (JES3) 309

T

TCP/IP
customizing sendmail 234
error when initialization is not finished 65
modifying the hlq.PROFILE.TCPIP data set 117
modifying the TCPIP.DATA data set 232
multiple stacks, configuring 133
specifying the name of 232, 313
standard translation table, IP PrintWay basic mode 313
SYSTCPD DD statement 323
TCPCONFIG statement in TCP/IP 117
TCPIP.DATA data set
DATASETPREFIX statement 233
for IP PrintWay basic mode 233
search order used by IP PrintWay basic mode 233
specifying name in IP PrintWay basic mode 321
SYSTCPD DD statement 323
TCPIPJOBNAME statement 232
TCPIPJOBNAME statement 232
TCPRCVBUFRSIZE parameter 118
TCPSENDBFRSIZE parameter 118
temporary files written by IP PrintWay basic mode 313
temporary files written by IP PrintWay extended 32
TERM parameter (JES3) 309
TermAPI function 475
termination handler, writing 477
time zone, setting 47, 340, 433
time-out value, increasing with sending large files to
LPD 138
TRACE parameter, in NetSpool startup procedure 179
trace, specifying for IP PrintWay basic mode 315
TRACEOPT DD statement 179, 180
tracing parameters for IP PrintWay basic mode 313
tracing, turning on in AOPSTART procedure 61
trailer, document
Begin data set exit 258
example 268
XTP_DOC_TRAILER_LENGTH field 265
XTP_DOC_TRAILER_PTR field 265
XTP_TRANS_DOC_TRAILER field 262
transform attribute 160, 404
transform classes 159
transform configuration file, aopxfd.conf
creating 159
environment attribute 160, 404
example 162
format 160, 404
maximum-active attribute 161, 405
maximum-idle-time attribute 161, 405
minimum-active attribute 161, 405
start-command attribute 160, 404
start-daemons attribute 158
transform attribute 160, 404
Transform data structure 462
Transform Interface
customizing 157
Transform Interface, overview 19
Transform Manager for Linux
code page conversion table 169

Transform Manager for Linux (*continued*)
configuration file 165
customization tasks 157
firewall 168
resource security 168
software requirement 158
Transform Manager, Infoprint Server
configuration file 159
transforms
components to customize 25
transforms, overview 20
translation table, standard 313
transmission-queue data set, IP PrintWay
allocating and initializing 317
defining in startup procedure 323
initializing 318
overview 316
restricting access 319
sharing between IP PrintWay FSSs 302, 316
Transparent Data Control exit, NetSpool
description 189
examples 191
input 190
installing 194
modifying 194
output 191
programming considerations 182
sample 193
transparent data exits in NetSpool, writing 181
TRKCELL parameter
JES2 308
SPOOLDEF 306
TRUNC parameter 127
TSO logon procedure, defining libraries in 66
TYPE parameter (JES3) 310
types of LU sessions supported 13
TZ environment variable 47, 340, 433

U

UCS parameter (JES2) 308
unaltered job ID, NetSpool 452
unaltered job ID, NetSpool 426
URI for printer 143
URL for printer 143
user interface
ISPF 485
TSO/E 485
userid attribute 147, 407

V

VTAM APPL statement for IP PrintWay, creating 236
VTAM applications, customization required to print from 25
VTAM protocol, IP PrintWay
AOP_APPLID variable 217
APPL ID, specifying in basic mode 312
APPL statement, creating in VTAM 236
customization tasks specific to protocol 201
FSS definition, creating 312
software required 205

W

WC parameter (JES3) 311
web browsers, customizing for Infoprint Central 365

- Windows client, customizing 369
- Windows client, overview 5
- Windows printing, customization required 25
- Workload Manager 112
- WritePrintFile function 475
- WS parameter
 - in DEVICE statement (JES3) 311
 - in PRTnnnnn statement (JES2) 308

X

- xcf-group-qualifier attribute 37
- XML Toolkit
 - level required for Infoprint Central 328
 - LIBPATH environment variable 339

Z

- z/OS HTTP Server, customizing 332
- zFS directories 30



Product Number: 5650-ZOS 5655-N62

Printed in USA

SA38-0691-00

