

z/OS



# Infoprint Server Customization

*Version 2 Release 2.0*

**Note**

Before using this information and the product it supports, read the information in “Notices” on page 493.

This edition applies to these IBM products:

- Version 2 Release 2 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, 2015.**

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

---

# Contents

<b>Figures . . . . .</b>	<b>ix</b>
--------------------------	-----------

<b>Tables . . . . .</b>	<b>xi</b>
-------------------------	-----------

<b>About this document . . . . .</b>	<b>xiii</b>
--------------------------------------	-------------

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

<b>How to send your comments to IBM . . . . .</b>	<b>xvii</b>
---------------------------------------------------	-------------

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

<b>Summary of changes . . . . .</b>	<b>xix</b>
-------------------------------------	------------

z/OS Version 2 Release 2 . . . . .	xix
z/OS Version 2 Release 1 . . . . .	xxii

<b>Chapter 1. Introducing Infoprint Server . . . . .</b>	<b>1</b>
----------------------------------------------------------	----------

Printer Inventory Manager. . . . .	3
Windows client . . . . .	6
Print Interface . . . . .	7
Print Interface with an SAP R/3 application server on the z/OS system . . . . .	11
Print Interface with an SAP R/3 application server on a remote system . . . . .	12
NetSpool . . . . .	14
IP PrintWay . . . . .	16
Transform Interface. . . . .	21
Transforms . . . . .	22
Infoprint Central . . . . .	23

<b>Chapter 2. Customization roadmap . . . . .</b>	<b>25</b>
---------------------------------------------------	-----------

<b>Chapter 3. Customizing the Printer Inventory Manager. . . . .</b>	<b>27</b>
--------------------------------------------------------------------------	-----------

Identifying software requirements for Printer Inventory Manager . . . . .	28
Enabling the Infoprint Server feature . . . . .	28
Updating the BPXPRMxx member of SYS1.PARMLIB . . . . .	28
Determining the Infoprint Server operating mode . . . . .	29
Setting the operating mode for z/OS V2R2       functions . . . . .	30
Making Language Environment and C++ runtime libraries available . . . . .	31
Creating Infoprint Server directories . . . . .	32
/etc/Printsrv directory . . . . .	33
/var/Printsrv directory . . . . .	33
Creating and editing the Infoprint Server configuration file (aopd.conf) . . . . .	37

Steps for creating the Infoprint Server configuration file (aopd.conf) . . . . .	44
Steps for editing the Infoprint Server configuration file (aopd.conf) . . . . .	45
Setting environment variables for the Printer Inventory Manager . . . . .	46
Setting up security for the Printer Inventory and operator commands . . . . .	52
AOP.ADMINISTRATOR profile. . . . .	53
AOP.CONFIGURATION profile. . . . .	54
AOPADMIN and AOPOPER groups . . . . .	54
AOPCONF group . . . . .	54
Steps for setting up security for the Printer Inventory, common message log, and operator commands. . . . .	55
Setting permissions for directories and executable files -- aopsetup . . . . .	59
Steps for running aopsetup . . . . .	59
Customizing startup and shutdown commands and procedures . . . . .	61
Editing the aopstart EXEC . . . . .	62
Creating an environment variables file ( <i>envvar</i> ) . . . . .	65
Creating a user ID to assign to the JCL startup and shutdown procedures . . . . .	65
Defining the JCL startup and shutdown procedures to RACF . . . . .	66
Editing JCL startup procedures . . . . .	67
Editing JCL shutdown procedures . . . . .	72
Starting Infoprint Server daemons automatically . . . . .	75
Editing the JCL status procedure . . . . .	76
Enabling Infoprint Server ISPF panels . . . . .	76
Defining libraries in the TSO logon procedure. . . . .	77
Displaying the Infoprint Server panel option . . . . .	77
Modifying the AOPINIT EXEC . . . . .	77
Enabling dynamic configuration . . . . .	78
Dynamic attributes . . . . .	79
Cross-system coupling facility (XCF) . . . . .	82
Steps for enabling dynamic configuration . . . . .	82
Steps for disabling dynamic configuration . . . . .	85
Steps for determining whether dynamic configuration is enabled . . . . .	86
Editing the system configuration definition for Printer Inventory Manager . . . . .	87
Steps for editing the system configuration definition . . . . .	91
Sending non-console messages to the hardcopy log Creating the message configuration file (aopmsg.conf) . . . . .	96
Editing the message configuration file (aopmsg.conf) . . . . .	97
Setting up the MVS system logger for the common message log . . . . .	98
Deciding whether to use the MVS system logger . . . . .	99
Deciding the type of log stream . . . . .	99
Deciding on a naming convention for log streams . . . . .	100

Determining how long to retain messages . . . . .	100
Determining the size of log-stream data sets and interim storage . . . . .	101
Setting up security for log streams . . . . .	101
Defining log streams to the MVS system logger	104
Activating a log stream in Infoprint Server . . . . .	109
Switching from one log stream to another . . . . .	111
Deactivating a log stream in Infoprint Server	112
Handling system logger errors. . . . .	113
Viewing messages in the log stream . . . . .	113
Backing up and restoring the Printer Inventory . . . . .	114
Steps for backing up the Printer Inventory to an HFS or zFS file . . . . .	115
Steps for restoring the Printer Inventory from an HFS or zFS file . . . . .	116
Steps for backing up the Printer Inventory to a generation data group (GDG) . . . . .	117
Steps for restoring the Printer Inventory from a GDG . . . . .	118
Restoring the Printer Inventory with the loadprinterinventory command . . . . .	120
Saving messages in the common message log in the z/OS UNIX file system. . . . .	120
Reducing the size of the common message log in the z/OS UNIX file system. . . . .	122
Setting Workload Manager goals . . . . .	123

## Chapter 4. Customizing Print Interface 125

Identifying software requirements for Print Interface . . . . .	126
Common customization for Print Interface. . . . .	126
Customizing z/OS Communications Server . . . . .	127
Updating the BPXPRMxx member of SYS1.PARMLIB. . . . .	129
Working with JES spool limits. . . . .	129
Editing the aopd.conf file for Print Interface . . . . .	130
Writing a filter program . . . . .	131
Customizing JES to specify no blank truncation	137
Setting environment variables for Print Interface	137
Editing the system configuration definition for Print Interface . . . . .	138
Customizing the z/OS UNIX printing commands	140
Customizing the AOPPRINT JCL procedure . . . . .	141
Customizing the Print Interface LPD . . . . .	142
Customizing z/OS Communications Server (TCP/IP) . . . . .	143
Editing the aopd.conf file for the LPD . . . . .	145
Editing the system configuration definition for the LPD . . . . .	146
Using the LPD compatibility filter . . . . .	147
Customizing LPR clients . . . . .	147
Customizing the z/OS SMB server to use Print Interface . . . . .	149
Customizing the Print Interface IPP Server . . . . .	150
Ensuring sufficient memory is available to start the IPP Server . . . . .	151
Editing the aopd.conf file for the IPP Server . . . . .	151
Setting environment variables for the IPP Server	152
Editing the system configuration definition for the IPP Server . . . . .	153
Customizing the IPP workstation client. . . . .	153

Customizing SAP R/3 . . . . .	155
Defining an SAP user for the callback daemon	156
Creating the SAP callback daemon configuration file (aopsapd.conf). . . . .	157
Setting environment variables for the SAP OMS	159
Defining a ROMS and LOMS to the SAP R/3 application server . . . . .	159
Modifying the LOMS to specify job attributes	160
Creating an SAP R/3 output device . . . . .	161
Customizing the Print Interface subsystem . . . . .	163
Editing the aopd.conf file for the Print Interface subsystem . . . . .	164
Increasing the maximum number of active transform daemons . . . . .	165

## Chapter 5. Customizing Transform Interface . . . . . 167

Identifying software requirements for Transform Interface . . . . .	168
Editing the aopd.conf file for the Transform Manager . . . . .	168
Creating the transform configuration file (aopxfd.conf) . . . . .	169
Transform classes . . . . .	169
Format of an entry in the transform configuration file . . . . .	169
Example of a transform entry . . . . .	171
Steps for creating the transform configuration file . . . . .	172
Steps for editing the transform configuration file	174
Creating the remote transform configuration file (aoprxf.conf). . . . .	175
Format of the remote transform configuration file . . . . .	175
Steps for creating the remote transform configuration file . . . . .	175
Setting environment variables for Transform Interface . . . . .	176
Customizing the TCP/IP firewall. . . . .	178
Setting up security for transform resources . . . . .	178
Creating a code page conversion table . . . . .	178

## Chapter 6. Customizing NetSpool. . . . . 179

Identifying software requirements for NetSpool	180
Updating the BPXPRMxx member of SYS1.PARMLIB. . . . .	180
Editing the aopd.conf file for NetSpool. . . . .	181
Customizing the NetSpool message log. . . . .	182
Customizing the NetSpool startup procedure. . . . .	182
Determining the number of NetSpool tasks to start . . . . .	183
Using the Japanese version. . . . .	183
Using an extended MCS console . . . . .	184
Creating a user ID to assign to the NetSpool startup procedure . . . . .	184
Defining the NetSpool startup procedure to RACF . . . . .	185
Editing the NetSpool startup procedure . . . . .	185
Working with JES spool limits. . . . .	188
Changing Language Environment runtime options	189

Writing NetSpool exit and filter programs . . . . .	189
Programming considerations . . . . .	190
APIPP1X2 control block . . . . .	191
Beginning of File exit for SCS data streams (APIPPTD1) . . . . .	192
Beginning of File exit for 3270 data streams (APIUBF3) . . . . .	194
Transparent Data Control exit (APIPPTD2) . . . . .	197
Graphic Escape exit (APIUGEX) . . . . .	199
Sample exits and macro . . . . .	201
Installing the exits . . . . .	201
Modifying the exits . . . . .	202
Setting environment variables for NetSpool . . . . .	202
Editing the system configuration definition for NetSpool . . . . .	203
 <b>Chapter 7. Customizing IP PrintWay</b>	<b>207</b>
Comparing IP PrintWay extended mode to basic mode . . . . .	210
Identifying software requirements for IP PrintWay . . . . .	213
Editing the aopd.conf file for IP PrintWay (extended mode) . . . . .	213
Increasing the space available in /var/Printsrv directory (extended mode) . . . . .	215
Updating the BPXPRMxx member of SYS1.PARMLIB (extended mode) . . . . .	215
Creating IP PrintWay job selection rules (extended mode) . . . . .	216
Creating job selection rules in the Printer Inventory (extended mode) . . . . .	220
Starting and stopping job selection rules (extended mode) . . . . .	221
Authorizing IP PrintWay to select data sets from the JES spool (extended mode) . . . . .	222
Authorizing IP PrintWay to write SMF type 6 records (extended mode) . . . . .	223
Setting environment variables for IP PrintWay (extended mode) . . . . .	224
Editing the system configuration definition for IP PrintWay (extended mode) . . . . .	232
APF-authorizing SYS1.IMAGELIB (extended mode) . . . . .	240
Customizing from AFP transforms for IP PrintWay (extended mode) . . . . .	240
Defining printer host names in the DNS . . . . .	240
Customizing z/OS Communications Server . . . . .	241
Customizing the TCPIP.DATA data set . . . . .	241
Customizing the PROFILE.TCPIP data set . . . . .	243
Customizing z/OS UNIX sendmail . . . . .	243
Creating a VTAM APPL resource definition for IP PrintWay . . . . .	245
Writing IP PrintWay exit and filter programs . . . . .	246
Comparing IP PrintWay basic mode and extended mode exits . . . . .	248
Resubmit for filtering considerations (basic mode) . . . . .	250
Programming considerations . . . . .	251
Common control blocks . . . . .	252
Routing exit (ANFUXRTG) . . . . .	256
Begin Data Set exit . . . . .	267
Record exit . . . . .	279
End Data Set exit . . . . .	287

SMF exit (ANFUXSMF) . . . . .	296
Message exit (ANFUXMSG) — basic mode . . . . .	302
Response Notification exit (ANFUXRSP) . . . . .	304
Using the Block Letter program (ANFUBLK) . . . . .	310
IP PrintWay sample exits . . . . .	312
Installing IP PrintWay exits . . . . .	312
Modifying IP PrintWay exits . . . . .	313
Updating the BPXPRMxx member of SYS1.PARMLIB (basic mode) . . . . .	313
APF-authorizing SYS1.IMAGELIB (basic mode) . . . . .	314
Defining the IP PrintWay functional subsystem (basic mode) . . . . .	314
Determining how many functional subsystems to define (basic mode) . . . . .	314
Determining how many functional subsystem applications to define (basic mode) . . . . .	315
Determining work-selection criteria (basic mode) . . . . .	316
Specifying a default font and page definition for data transforms (basic mode) . . . . .	317
Defining the IP PrintWay functional subsystem in JES2 (basic mode) . . . . .	318
Defining the IP PrintWay functional subsystem in JES3 (basic mode) . . . . .	321
Creating FSS and FSA definitions in the Printer Inventory (basic mode) . . . . .	325
Creating FSS definitions (basic mode) . . . . .	325
Creating FSA definitions (basic mode) . . . . .	328
Creating the transmission-queue data set (basic mode) . . . . .	328
Sharing the transmission-queue data set (basic mode) . . . . .	329
Allocating and initializing the transmission-queue data set (basic mode) . . . . .	330
Restricting access to the transmission-queue data set (basic mode) . . . . .	331
Creating the IP PrintWay message-log data set (basic mode) . . . . .	331
Customizing the IP PrintWay startup procedure (basic mode) . . . . .	332
Creating a user ID to assign to the IP PrintWay startup procedure (basic mode) . . . . .	333
Defining the IP PrintWay startup procedure to RACF (basic mode) . . . . .	333
Editing the ANFWPROC startup procedure (basic mode) . . . . .	333
Changing Language Environment runtime options (basic mode) . . . . .	337
Customizing PSF for sharing network printers . . . . .	337
Customizing AOPCMND for starting and stopping printers . . . . .	337

 <b>Chapter 8. Customizing Infoprint Central</b>	<b>339</b>
Identifying software requirements for Infoprint Central . . . . .	340
Editing the aopd.conf file for Infoprint Central . . . . .	341
Customizing the IBM HTTP Server - Powered by Apache . . . . .	343
Configuring the IBM HTTP Server - Powered by Apache . . . . .	344

	Editing the configuration file for the IBM HTTP	
	Server - Powered by Apache (conf/httpd.conf)	345
	Customizing the environment variables file for	
	the IBM HTTP Server - Powered by Apache	
	(bin/envvars)	346
	Creating a started task to run the IBM HTTP	
	Server - Powered by Apache	351
	Setting up security for Infoprint Central	352
	Authorizing the AOOPER group to RACF	
	profiles	353
	Defining Infoprint Central users as z/OS UNIX	
	users	355
	Activating the PRINTSRV class	355
	Authorizing users to read the Printer Inventory	355
	Setting up security for printers	356
	Setting up security for NetSpool logical units	
	(LUs)	364
	Setting up security for IP PrintWay job selection	
	rules	365
	Setting up security for print jobs	367
	Setting up security for Infoprint Server daemons	373
	Setting up security for printer and printer pool	
	definitions	373
	Requesting RACF notification and auditing	373
	Editing the system configuration definition for	
	Infoprint Central	374
	Customizing web browsers	376
	Logging on to Infoprint Central from the web	377
	Customizing JES3 console standards (CONSTD	
	statement)	378
	Re-creating the search database	378

## Chapter 9. Installing and customizing the Infoprint Server Windows client. . . 381

## Chapter 10. Customizing PSF for z/OS to use the Printer Inventory . . . 385

Identifying PSF software requirements	385
Creating PSF FSS and FSA definitions in the	
Printer Inventory	385
Customizing PSF for the Printer Inventory	386
Customizing PSF for z/OS to write messages to	
the common message log	386

## Appendix A. Checklist of Infoprint Server customization tasks . . . 389

## Appendix B. Infoprint Server files. . . 395

aopd.conf -- Infoprint Server configuration file	395
Purpose	395
Format	395
Usage notes	406
aopmsg.conf -- Infoprint Server message	
configuration file	406
Purpose	406
Format	406
Usage notes	408
aoprxf.conf -- Infoprint Server remote transform	
configuration file	408

Purpose	408
Format	408
Usage notes	409
aopxrd.conf -- Infoprint Server transform	
configuration file	409
Purpose	409
Example	409
Usage notes	409
aopsapd.conf -- Infoprint Server SAP callback	
daemon configuration file	410
Purpose	410
Format	410
Example	410
Usage notes	411

## Appendix C. Environment variables 413

## Appendix D. Configuration attributes 437

## Appendix E. AOPBATCH program . . 457

Format	457
Description	457
Parameters	458
Usage notes	459
Exit values	460
Examples	460
Sample jobs	460

## Appendix F. Application programming interface (API) . . . 461

API files	461
API data structures	461
DocumentInfo - information about a document	
in a job	462
JobInfo - information about an Infoprint Server	
print job and each document in the job	463
JobToken - identifies an Infoprint Server print	
job	463
PrinterInfo - information about an Infoprint	
Server printer	463
QueryToken - identifies an Infoprint Server	
query request	464
Transform - information about a transform	
connection	464
API functions	465
AbortPrintFile -- terminate an Infoprint Server	
print job	466
BeginEnumJobs -- begin an enumeration of	
Infoprint Server print jobs for a printer	466
BeginEnumPrinters -- begin an enumeration of	
Infoprint Server printers	467
BufferSizeRequired -- return the required size of	
the buffer after a buffer overflow error	467
CancelJob -- cancel a print job	468
ClosePrintFile -- close an Infoprint Server print	
job	468
CreatePrintFile -- create an Infoprint Server	
print job	469
EndEnumJobs -- end the enumeration of	
Infoprint Server print jobs for a printer	470

EndEnumPrinters -- end the enumeration of Infoprint Server printers. . . . .	470	Serviceability enhancements . . . . .	487
EnumJobs -- enumerate Infoprint Server print jobs for a printer . . . . .	470	Start and stop Infoprint Server daemons . . . . .	487
EnumPrinters -- enumerate Infoprint Server printers . . . . .	471	TSO/E command for controlling printers . . . . .	487
ErrorNumber -- return the API error code . . . . .	472	Variable inline email message . . . . .	488
ErrorString -- return a pointer to the API error message . . . . .	473	<b>Appendix I. Accessibility . . . . .</b>	<b>489</b>
GetAPIVersionNumber -- return the current API version number . . . . .	473	Accessibility features . . . . .	489
GetJobInfo -- return the status of an Infoprint Server print job. . . . .	474	Consult assistive technologies . . . . .	489
GetPrinterInfo -- return information about an Infoprint Server printer . . . . .	475	Keyboard navigation of the user interface . . . . .	489
InitAPI -- initialize the API. . . . .	476	Dotted decimal syntax diagrams . . . . .	489
ReleaseTransform -- release a connection with a transform daemon. . . . .	476	<b>Notices . . . . .</b>	<b>493</b>
RequestTransform -- establish a connection with a transform daemon . . . . .	477	Policy for unsupported hardware. . . . .	494
SetTerminationHandler -- register a termination function . . . . .	477	Minimum supported hardware . . . . .	495
TermAPI -- terminate the API . . . . .	478	Programming interfaces . . . . .	495
WritePrintFile -- write data to an Infoprint Server print job. . . . .	478	Trademarks . . . . .	495
Programming considerations . . . . .	479	<b>Glossary . . . . .</b>	<b>497</b>
Customization requirements . . . . .	480	<b>Index . . . . .</b>	<b>507</b>
 <b>Appendix G. Migrating from IP PrintWay basic mode to extended mode . . . . . 483</b>			
 <b>Appendix H. Benefiting from new functions in z/OS V2R2 . . . . . 487</b>			
Infoprint Central enhancements . . . . . 487			





## Figures

1.	Infoprint Server system diagram . . . . .	2	17.	Sample JCL to define a coupling-facility structure in the CFRM policy . . . . .	108
2.	Printer Inventory Manager system diagram . . . . .	4	18.	Sample JCL to define a coupling-facility log stream to the system logger. . . . .	109
3.	Print Interface system diagram . . . . .	8	19.	Sample UNIX filter . . . . .	134
4.	Printing SAP R/3 documents with an SAP R/3 application server that runs on a z/OS system . . . . .	11	20.	AOPPRINT JCL procedure in SYS1.IBM.PROCLIB . . . . .	141
5.	Printing SAP R/3 documents with an SAP R/3 application server that runs on a non-z/OS system . . . . .	13	21.	NetSpool startup procedure — SYS1.IBM.PROCLIB(APIJPJCL). . . . .	186
6.	NetSpool system diagram. . . . .	14	22.	Language Environment runtime options for NetSpool . . . . .	189
7.	IP PrintWay system diagram . . . . .	18	23.	Sample APPL statement for IP PrintWay extended mode . . . . .	246
8.	Transform Interface system diagram . . . . .	21	24.	Sample APPL statement for IP PrintWay basic mode . . . . .	246
9.	AOPSTART startup procedure — SYS1.IBM.PROCLIB(AOPSTART) . . . . .	69	25.	Sample JES2 initialization statements . . . . .	318
10.	AOPDEMON startup procedure — SYS1.IBM.PROCLIB(AOPDEMON). . . . .	71	26.	Sample JES3 initialization statements . . . . .	321
11.	AOPSTAR2 startup procedure — SYS1.IBM.PROCLIB(AOPSTAR2) . . . . .	72	27.	IDCAMS parameters to allocate a transmission-queue data set — SYS1.SAMPLIB(ANFDEAL). . . . .	331
12.	AOPSTOP shutdown procedure — SYS1.IBM.PROCLIB(AOPSTOP). . . . .	74	28.	IP PrintWay startup procedure — SYS1.IBM.PROCLIB(ANFWPROC) . . . . .	335
13.	AOPSTOP2 shutdown procedure — SYS1.IBM.PROCLIB(AOPSTOP2) . . . . .	75	29.	Language Environment runtime options for IP PrintWay . . . . .	337
14.	AOPSTAT procedure — SYS1.IBM.PROCLIB(AOPSTAT) . . . . .	76			
15.	Sample JCL to define a DASD-only log stream to the system logger. . . . .	106			
16.	Sample JCL to define a coupling-facility structure to the system logger . . . . .	108			



## Tables

1.	Syntax notation . . . . .	xiii
2.	Documents for Infoprint Server . . . . .	xv
3.	Documents for IBM transform products . . . . .	xvi
4.	Documents for Ricoh products . . . . .	xvi
5.	Transforms that work with Infoprint Server . . . . .	22
6.	Summary of Infoprint Server components . . . . .	25
7.	Customization tasks for Printer Inventory Manager . . . . .	27
8.	Configurations that affect how Infoprint Server operates. . . . .	29
9.	Number of documents in the Historical Inventory . . . . .	43
10.	Where to define environment variables for the Printer Inventory Manager . . . . .	48
11.	RACF access for Infoprint Server . . . . .	52
12.	Customization tasks for startup and shutdown procedures . . . . .	62
13.	Dynamic attributes and ISPF field names . . . . .	79
14.	Attributes in system configuration definition and the equivalent attributes in aopd.conf and aopmsg.conf and equivalent environment variables . . . . .	81
15.	Number of documents in the Historical Inventory . . . . .	90
16.	Customization tasks for system logger . . . . .	98
17.	DASD-only log stream parameters for Infoprint Server. . . . .	105
18.	Coupling-facility structure parameters for Infoprint Server. . . . .	107
19.	Coupling-facility log stream parameters for Infoprint Server. . . . .	107
20.	Customization tasks for Print Interface . . . . .	125
21.	Customization tasks for Infoprint Server components . . . . .	126
22.	Common customization tasks for Print Interface . . . . .	127
23.	Customization tasks for Print Interface LPD . . . . .	143
24.	Customization tasks for Print Interface IPP Server . . . . .	150
25.	IPP attributes and corresponding Infoprint Server attributes . . . . .	155
26.	Customization tasks for Print Interface when SAP R/3 application server runs on same z/OS system. . . . .	156
27.	Customization tasks for Print Interface when SAP R/3 application server does not run on same z/OS system. . . . .	156
28.	Customization tasks for Print Interface subsystem . . . . .	163
29.	Customization tasks for Transform Interface . . . . .	167
30.	Other components to customize for Transform Interface . . . . .	167
31.	Customization tasks for NetSpool. . . . .	179
32.	Other components to customize for NetSpool . . . . .	179
33.	Customization tasks for NetSpool startup procedure. . . . .	183
34.	Customization tasks for IP PrintWay extended mode . . . . .	207
35.	Customization tasks for IP PrintWay basic mode . . . . .	208
36.	Customization tasks for IP PrintWay protocols . . . . .	209
37.	Other components to customize for IP PrintWay . . . . .	209
38.	How IP PrintWay extended mode differs from basic mode . . . . .	212
39.	IP PrintWay job selection rules. . . . .	217
40.	Customization tasks for TCP/IP . . . . .	241
41.	Contents of ANFUEXTP fields when <b>resubmit for filtering</b> option is selected (IP PrintWay basic mode) . . . . .	250
42.	XTP_ADDR_TYPE values . . . . .	262
43.	XTP_ADDR_TYPE values . . . . .	272
44.	XTP_ADDR_TYPE values . . . . .	283
45.	IP PrintWay FSS definition . . . . .	325
46.	Customization tasks for IP PrintWay startup procedure (basic mode) . . . . .	332
47.	Customization tasks for Infoprint Central . . . . .	340
48.	Other components to customize for Infoprint Central. . . . .	340
49.	Customization tasks for using the HTTP Server - Powered by Apache with Infoprint Central. . . . .	344
50.	Environment variables in bin/envvars . . . . .	347
51.	Customization tasks for Infoprint Central security . . . . .	352
52.	Profiles in the OPERCMDS class and minimum access required for the AOOPER group— JES2 systems. . . . .	353
53.	Profiles in the OPERCMDS class and minimum access required for the AOOPER group— JES3 systems. . . . .	353
54.	Infoprint Central printer actions and the minimum access required to profiles in the PRINTSRV class . . . . .	357
55.	Infoprint Central printer actions and minimum access required to profiles in the OPERCMDS class — JES2 systems . . . . .	361
56.	Infoprint Central printer actions and the minimum access required to profiles in the OPERCMDS class — JES3 systems . . . . .	362
57.	Infoprint Central NetSpool LU actions and the minimum access required to the AOP.NETSPOOL profile in the PRINTSRV class . . . . .	364
58.	Infoprint Central job selection rule actions and minimum access required to the profile in the PRINTSRV class . . . . .	365
59.	Infoprint Central print job actions and minimum access required to profiles in the OPERCMDS class - JES2 . . . . .	368

60.	Infoprint Central print job actions and the minimum access required to profiles in the OPERCMDS class - JES3 . . . . .	369	64.	Customization tasks for PSF for z/OS	385
61.	Infoprint Central print job actions and the minimum access required to profiles in the JESSPOOL class. . . . .	371	65.	Number of documents in the Historical Inventory . . . . .	402
62.	Infoprint Central daemon action and the minimum access required to the AOP.DAEMON profile in the PRINTSRV class	373	66.	Where to set environment variables . . . .	414
63.	Comparison of Windows printing methods	381	67.	Number of documents in the Historical Inventory . . . . .	451
			68.	API files . . . . .	461
			69.	API data structures . . . . .	461
			70.	API functions . . . . .	465
			71.	ErrorNumber names and error descriptions	472

---

## About this document

This document helps you customize Infoprint Server in z/OS® Version 2 Release 2. 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 need to 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 that are 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 V2R2 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 type the entire command.

Table 1. Syntax notation

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

Table 1. Syntax notation (continued)

This notation:	Means:	You enter:	For example:	
			This document shows:	You enter:
Parentheses	List of items	The parentheses and one or more items from the list	(GT10,GT12)	(GT10,GT12)
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 that is related to z/OS and Infoprint Server.

### Preventive Service Planning information

Before you install Infoprint Server, review the current Preventive Service Planning (PSP) information, also called the *PSP bucket*. You also periodically review the current PSP information. The PSP upgrade ID depends on your z/OS operating system; for example, ZOSV2R2. 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 before installation.

### Infoprint Server migration information

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

Document	Form number	Description
<i>z/OS Migration</i>	GA32-0889	Describes the tasks that are required to upgrade to z/OS V2R2. Includes the migration tasks for Infoprint Server.
<i>z/OS Summary of Message and Interface Changes</i>	SA23-2300	Describes new and changed messages and interfaces in z/OS V2R2. Includes the messages and interfaces for Infoprint Server.

Document	Form number	Description
<i>z/OS Infoprint Server Customization</i>	SA38-0691	Describes the tasks that are required to migrate to IP PrintWay extended mode from IP PrintWay basic mode. Also, describes the tasks that are required to use the new functions that are introduced in z/OS V2R2.

## Documents

These tables list 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 2. Documents for Infoprint Server*

Document	Form number	Description
<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 by 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 that are 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 3. Documents for IBM transform products

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

Table 4. Documents for Ricoh products

Ricoh Document	Description
<i>InfoPrint Transform Manager for AIX®, Linux, and Windows: Planning and Installing</i>	Describes InfoPrint Transform Manager and provides installation and configuration information about the product
<i>InfoPrint Transform Manager for Linux: afp2pdf Transform Installation and User's Guide</i>	Describes the InfoPrint Transform Manager for Linux AFP to PDF transform

## 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 general help system and help for individual web pages, select the question mark (?) on the title bar.

The **man** command provides online help for z/OS UNIX commands. The syntax is:

**man** *command\_name*

TSO/E provides online help for the AOPCMND command. The syntax is:

**HELP** AOPCMND



---

## 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](mailto:mhvrcfs@us.ibm.com).
2. Send an email from the "Contact us" web page for z/OS (<http://www.ibm.com/systems/z/os/zos/webqs.html>).

Include the following information:

- Your name and address.
- Your email address.
- Your telephone or fax number.
- The publication title and order number:  
z/OS Infoprint Server Customization  
SA38-0691-01
- The topic and page number that is 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 the comments in any way appropriate without incurring any obligation to you.

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

---

## If you have a technical problem

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 z/OS Support Portal (<http://www-947.ibm.com/systems/support/z/zos/>).



---

## Summary of changes

This section describes the release enhancements that were made.

---

### z/OS Version 2 Release 2

The following changes are made for z/OS Version 2 Release 2.

#### New

- You can now configure Infoprint Server to start and stop individual daemons. When this configuration is set, the **aopstart** EXEC and the **start-daemons** attribute are ignored. See this new information:
    - “Determining the Infoprint Server operating mode” on page 29
    - “Setting environment variables for the Printer Inventory Manager” on page 46
    - “Customizing startup and shutdown commands and procedures” on page 61 for the new startup and shutdown commands and JCL procedures, including “AOPDEMON JCL procedure” on page 69, “AOPSTAR2 JCL procedure” on page 71, “AOPSTOP2 JCL procedure” on page 74.
- Note:** In some cases throughout the publication, specific references to **aopstart** and **aopstop** commands and AOPSTART and AOPSTOP JCL procedures are changed to more general “start and stop commands” and “JCL startup and shutdown procedures”.
- “Creating an environment variables file (*envvar*)” on page 65 for a new file that is used instead of the **aopstart** EXEC.
  - A new field, **Operating mode**, is added to the ISPF Infoprint Server System Configuration panel in “Steps for editing the system configuration definition” on page 91. Also, the PIDU attribute, **ipsmode**, is added to “Dynamic attributes” on page 79, “Editing the system configuration definition for IP PrintWay (extended mode)” on page 232, and Appendix D, “Configuration attributes,” on page 437.
  - A new TSO/E command processor, AOPCMND, is added for starting and stopping IP PrintWay extended mode printers. See Table 38 on page 212 and “Customizing AOPCMND for starting and stopping printers” on page 337.
  - “Editing the JCL status procedure” on page 76 is added to describe the AOPSTAT JCL procedure for displaying the status of Infoprint Server daemons and NetSpool tasks.
  - “Restoring the Printer Inventory with the loadprinterinventory command” on page 120 describes a new command for restoring the Printer Inventory.
  - Infoprint Central now supports Mozilla Firefox. See “Identifying software requirements for Infoprint Central” on page 340 and “Customizing web browsers” on page 376.
  - Because z/OS V2R2 supports the IBM HTTP Server - Powered by Apache only, you must now use the Apache 31-bit IBM HTTP Server with Infoprint Central. “Customizing the IBM HTTP Server - Powered by Apache” on page 343 is updated with tasks for customizing the IBM HTTP Server - Powered by Apache.
  - “GetAPIVersionNumber -- return the current API version number” on page 473 is new.

- Appendix H, “Benefiting from new functions in z/OS V2R2,” on page 487 includes new functions.

## Changed

- The description of Infoprint Central is updated in Chapter 1, “Introducing Infoprint Server,” on page 1.
- Web pages for downloading the AFP Printer Driver and the AFP Viewer plug-in are updated. For example, see “Windows client” on page 6.
- Tasks that you can do with Infoprint Central are updated. See “Infoprint Central” on page 23.
- “Making Language Environment and C++ runtime libraries available” on page 31 is updated.
- Information is added to “/var/Printsrv directory” on page 33 about not placing user directories or files in the /var/Printsrv directory and about using a z/OS File System (zFS) instead of a Hierarchical File System (HFS). The note has been clarified to indicate that the directory might contain Printer Inventory Version 1 and Version 2 files when you upgrade from Infoprint Server V1R11 or earlier.
- The **dynamic-configuration**, **log-retention**, and **start-daemons** attributes are updated in “Creating and editing the Infoprint Server configuration file (aopd.conf)” on page 37, “aopd.conf -- Infoprint Server configuration file” on page 395, and Appendix D, “Configuration attributes,” on page 437.
- The **log-retention** attribute in “Creating and editing the Infoprint Server configuration file (aopd.conf)” on page 37, “Editing the system configuration definition for Printer Inventory Manager” on page 87, “Editing the aopd.conf file for Infoprint Central” on page 341, “Editing the system configuration definition for Infoprint Central” on page 374, “aopd.conf -- Infoprint Server configuration file” on page 395, and Appendix D, “Configuration attributes,” on page 437 is updated with information about setting environment variables so the date and time are displayed in messages.
- The rule is updated for the **base-directory** and **inventory** attributes in “Creating and editing the Infoprint Server configuration file (aopd.conf)” on page 37, “Editing the aopd.conf file for the Print Interface subsystem” on page 164, and “aopd.conf -- Infoprint Server configuration file” on page 395.
- References to the IPP Server daemon named “**aopippd**” are changed to “**aopippd**”. See:
  - “start-daemons= {[ippd] [lpd] [netd] [outd] [ssid] [subd] [xfd]}” on page 38
  - “Starting Infoprint Server daemons automatically” on page 75
  - Table 13 on page 79
  - “Customizing z/OS Communications Server” on page 127
  - “Working with JES spool limits” on page 129
  - Table 63 on page 381
  - AOP\_MAXTHREADTASKS and AOPIPPD\_MAXTHREADTASKS in Appendix C, “Environment variables,” on page 413
  - **aopippd-max-thread-tasks** and **max-thread-tasks** in Appendix D, “Configuration attributes,” on page 437
- “Setting permissions for directories and executable files -- aopsetup” on page 59 is updated to indicate that all Infoprint Server daemons have the set-group-ID bit on.
- The example for displaying the UID of the owner ID is updated in “Steps for running aopsetup” on page 59.
- “Editing the aopstart EXEC” on page 62 is updated.
- “Creating a user ID to assign to the JCL startup and shutdown procedures” on page 65 is updated.

I

- The AOPSTART procedure is updated in “Editing JCL startup procedures” on page 67.
- The AOPSTOP procedure is updated in “Editing JCL shutdown procedures” on page 72.
- “Steps for enabling dynamic configuration” on page 82 is updated with a tip.
- “Editing the system configuration definition for Printer Inventory Manager” on page 87 is updated with information about message AOP192E.
- “Backing up and restoring the Printer Inventory” on page 114 is updated with a new command for restoring the Printer Inventory.
- Only Java™ Technology Edition, V7.1 is supported in Infoprint Server V2R2. For example, see “Identifying software requirements for Print Interface” on page 126 and “Setting environment variables for the IPP Server” on page 152.
- A UNIX filter that is defined as a REXX exec now requires that the program control extended attribute is turned on. See “Writing a UNIX filter” on page 134.
- The term “OS/390” is changed to “z/OS” in Figure 20 on page 141 and “Defining a ROMS and LOMS to the SAP R/3 application server” on page 159.
- Information about using the SHRLIBRGNSIZE OMVS parameter in a 31-bit environment is added to “Ensuring sufficient memory is available to start the IPP Server” on page 151, AOP\_MAXTHREADTASKS on page 420, and **max-thread-tasks** on page 452.
- The *envar* file example in “Setting environment variables for Transform Interface” on page 176 is updated.
- An example for receiving Japanese messages in NetSpool is added to “Using the Japanese version” on page 183.
- The word “standard” is removed from the type of message you can include in the beginning of emails. See “Comparing IP PrintWay extended mode to basic mode” on page 210.
- “Updating the BPXPRMxx member of SYS1.PARMLIB (basic mode)” on page 313 is updated with guidelines and tips for specifying parameters.
- “Authorizing the AOOPER group to RACF profiles” on page 353 is updated to include the profile in the JESSPOOL class for JES2. Also, the statement about giving the AOPADMIN group access to the profiles is deleted.
- Table 52 on page 353 and “Protecting actions on print jobs with profiles in the OPERCMDS class” on page 368 are updated with profiles and commands for TSO output jobs.
- “Logging on to Infoprint Central from the web” on page 377 is updated with a procedure for how to stop using Infoprint Server.
- “Re-creating the search database” on page 378 is updated and renamed.
- When you are changing the **Log messages** field in PSF V4R5, you do not need to restart the PSF functional subsystem (FSS) because PSF automatically picks up a change to this field. See “Customizing PSF for z/OS to write messages to the common message log” on page 386. Also, a new field, **PINST trace dsname**, is added to the PSF FSS panel.
- Appendix A, “Checklist of Infoprint Server customization tasks,” on page 389 is updated.
- Appendix C, “Environment variables,” on page 413 is updated.
- The description and usage notes in Appendix E, “AOPBATCH program,” on page 457 are updated.

## Deleted

- The “SNMP subagent” section is removed from Chapter 1, “Introducing Infoprint Server,” on page 1.
- References to “Simple Network Management Protocol (SNMP) subagent”, “SNMP”, “SNMP subagent”, “snmp-community”, and “snmpd” are removed from:
  - The Infoprint Server components that are listed after Figure 1 on page 2
  - Figure 2 on page 4 and the list of components in bullet 2
  - Table 6 on page 25
  - **start-daemons** in “Creating and editing the Infoprint Server configuration file (aopd.conf)” on page 37
  - “Starting Infoprint Server daemons automatically” on page 75
  - Table 13 on page 79
  - Table 14 on page 81
  - “Steps for editing the system configuration definition” on page 91
  - “Customizing z/OS Communications Server” on page 127
  - “Customizing PSF for the Printer Inventory” on page 386
  - Appendix A, “Checklist of Infoprint Server customization tasks,” on page 389
  - “aopd.conf -- Infoprint Server configuration file” on page 395
  - Appendix D, “Configuration attributes,” on page 437
  - “Glossary” on page 497
- References to “Windows Server 2003” and “Windows XP” as supported operating systems for Infoprint Port Monitor are removed throughout the document. For example, see “Windows client” on page 6.
- References to daemons **aoplogd**, **aopsdbd**, and **aophinvd** are removed from:
  - **start-daemons** in “Creating and editing the Infoprint Server configuration file (aopd.conf)” on page 37 and “aopd.conf -- Infoprint Server configuration file” on page 395
  - AOP\_MAXTHREADTASKS in Appendix C, “Environment variables,” on page 413
  - **max-thread-tasks** in Appendix D, “Configuration attributes,” on page 437
- References to using the **/etc/rc** shell script to start Infoprint Server are removed in “Starting Infoprint Server daemons automatically” on page 75 and throughout the document.
- **aophinvd-max-thread-tasks** and **aopsdbd-max-thread-tasks** are removed from Table 13 on page 79, Table 14 on page 81, and Appendix D, “Configuration attributes,” on page 437.
- LC\_TIME and TZ are removed from “Customizing the environment variables file for the IBM HTTP Server - Powered by Apache (bin/envvars)” on page 346.
- AOPHINVD\_MAXTHREADTASKS and AOPSDBD\_MAXTHREADTASKS are removed from Appendix C, “Environment variables,” on page 413.
- A chapter called *Customizing the SNMP subagent* is deleted.

---

## z/OS Version 2 Release 1

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

- *z/OS Summary of Message and Interface Changes*
- *z/OS Introduction and Release Guide*

- *z/OS Planning for Installation*
- *z/OS Migration*





---

## Chapter 1. Introducing Infoprint Server

Infoprint Server supports LAN and host printing on your z/OS system. It works together with data stream transforms that other transform products provide.

Figure 1 on page 2 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.

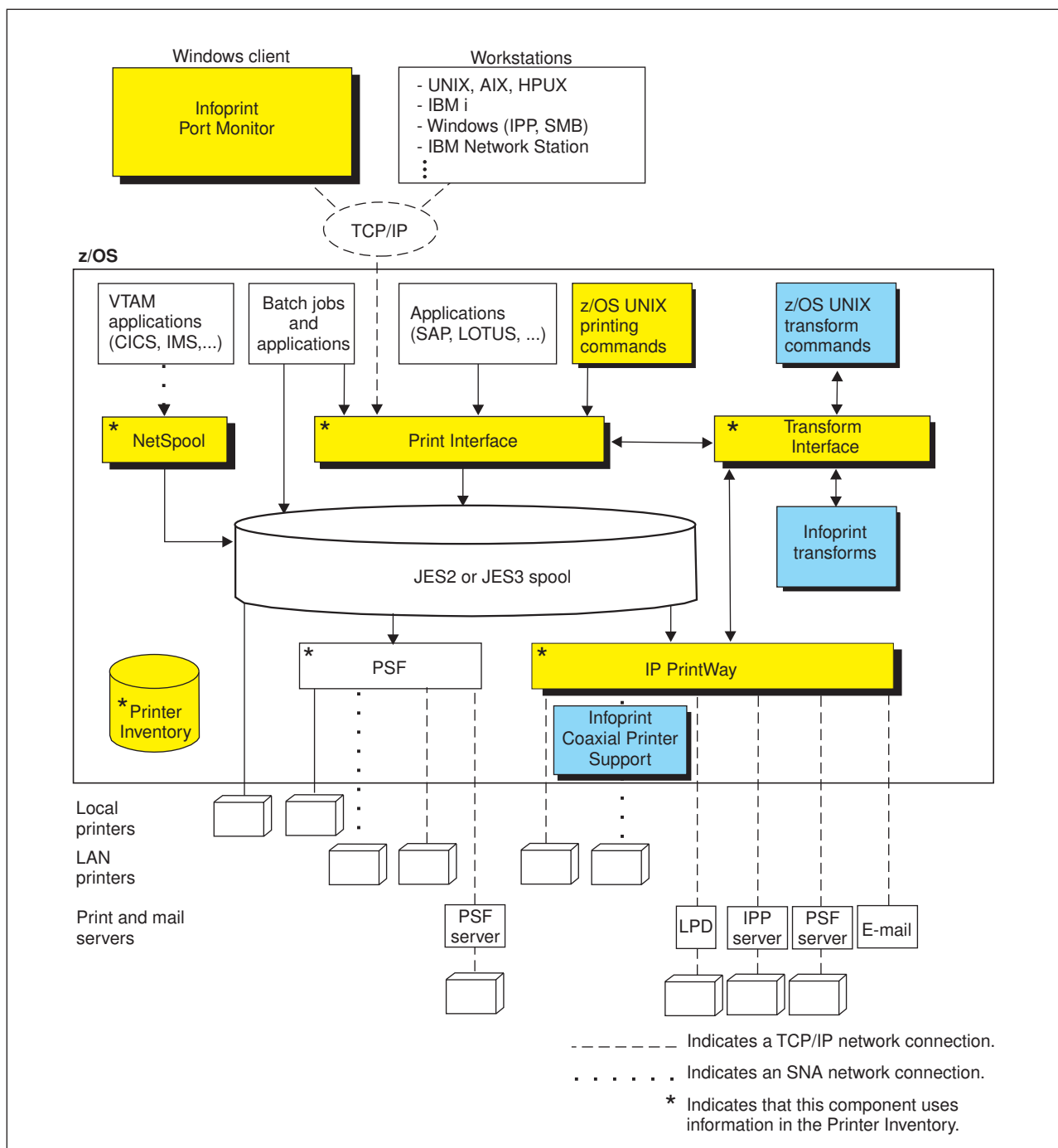


Figure 1. Infoprint Server system diagram

### 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 that are controlled by IP PrintWay extended mode or PSF, and NetSpool logical units. It also lets operators check system status, view printer definitions, and view, start, and stop job selection rules.

### **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 that is 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.

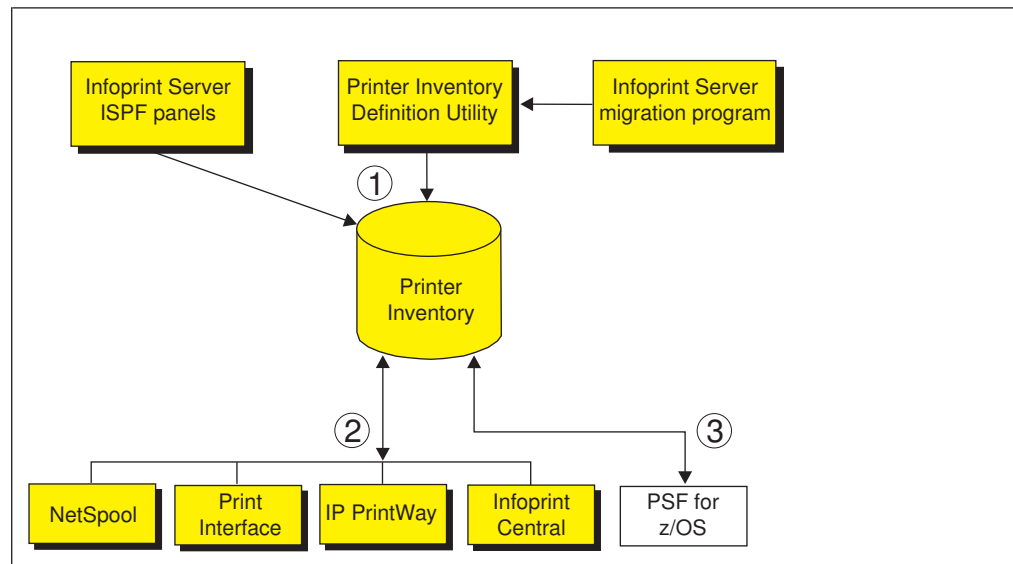


Figure 2. Printer Inventory Manager system diagram

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.
  - 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. For information, see *z/OS Infoprint Server Printer Inventory for PSF*. 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.

More functions that the Printer Inventory Manager provides are:

#### Common message log

The common message log displays 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 processed but that are no longer on the JES spool because

they finished processing or were deleted. Authorized users can use Infoprint Central to 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. You can use this program to 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 2008, Windows 7, and Windows Vista (Business Edition, Enterprise Edition).

**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 included with Infoprint Server at:

/usr/lpp/Printsrv/win/En\_US/aopwin32.exe

/usr/lpp/Printsrv/win/En\_US/aopwin64.exe

You can also download the Infoprint Port Monitor (<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:

#### **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.

#### **AFP Viewer plug-in for Windows**

You can use this product to view documents in AFP format. For example, documents that are downloaded from the z/OS system or documents on the web. You can also use the viewer to 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 these websites:

- Ricoh AFP Printer Driver website (<http://support.rpp.ricoh-usa.com/internet/dcfdata.nsf/vwWeb/P4000384>)
- Ricoh AFP Viewer plug-in website (<http://support.rpp.ricoh-usa.com/internet/dcfdata.nsf/vwwweb/p4000233>)

---

## **Print Interface**

The Print Interface component of Infoprint Server processes print requests received from both remote clients and local users. Figure 3 on page 8 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.

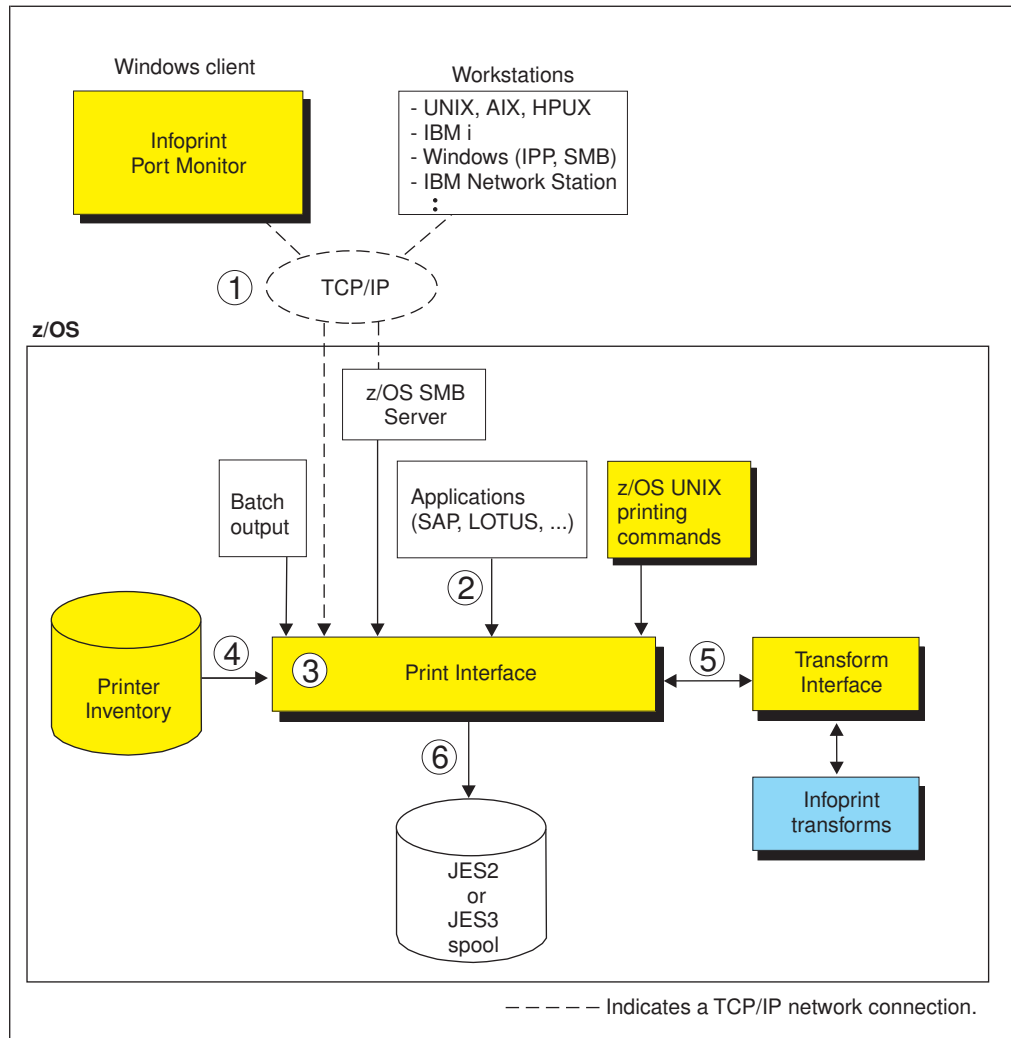


Figure 3. Print Interface system diagram

1. Users can submit print requests and query job status from remote clients in the Internet Protocol network with 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 6).
- The **lptrafp** command for Windows and UNIX systems. You can use this command to specify Infoprint Server job attributes. You can download the **lptrafp** command from the z/OS downloads website (<http://www.ibm.com/systems/z/os/zos/downloads/>). In the “As-is z/OS downloads” section, select **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 with one of these methods:

#### **Print Interface subsystem**

By using the Print Interface subsystem, you can transform and print output data that is created by a batch application with minimal changes to your JCL.

#### **AOPPRINT**

By 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**

By 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)**

By 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 that are 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 that are described in “Transforms” on page 22.

## **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 document processing is complete and the data set is 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*) are completed successfully, are retained because of failed transmission to LAN printers, or are 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 it writes 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. SAP users can use the OMS and Callback daemon to 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, see the SAP website. “Print Interface with an SAP R/3 application server on the z/OS system” on page 11 describes this support.

The Print Interface LPD can print documents that are received from an SAP R/3 application server that is 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 12 describes this support.

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

Figure 4 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 that run on other SAP R/3 systems, if the SAP spool work process runs on the z/OS system.

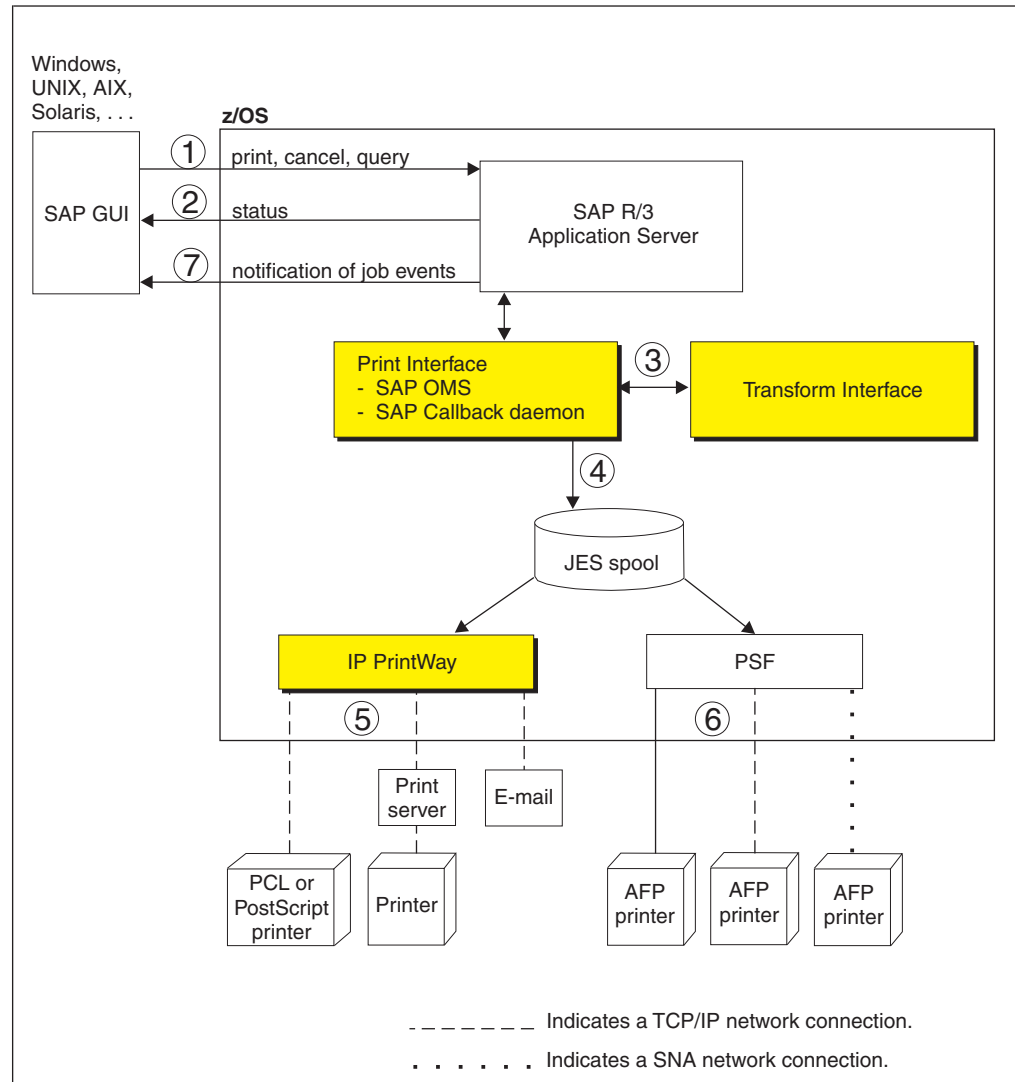


Figure 4. Printing SAP R/3 documents with an SAP R/3 application server that runs on a z/OS system

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 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 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 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 on page 13 shows how the Print Interface LPD and an SAP R/3 application server that runs 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.

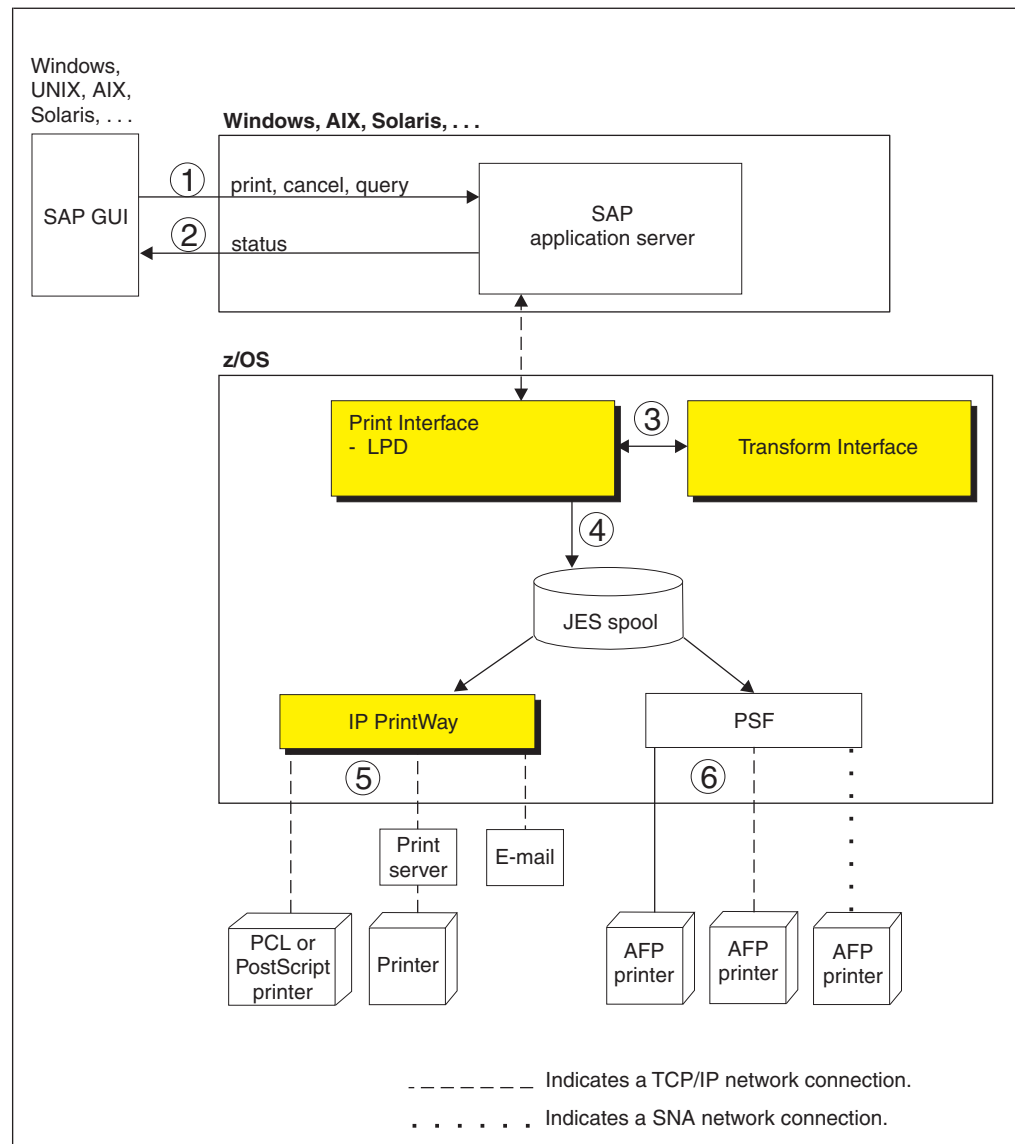


Figure 5. Printing SAP R/3 documents with an SAP R/3 application server that runs on a non-z/OS system

1. From an SAP R/3 GUI, users make print, cancel, job query, and device query requests to an SAP R/3 application server that runs 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 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 that 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 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.

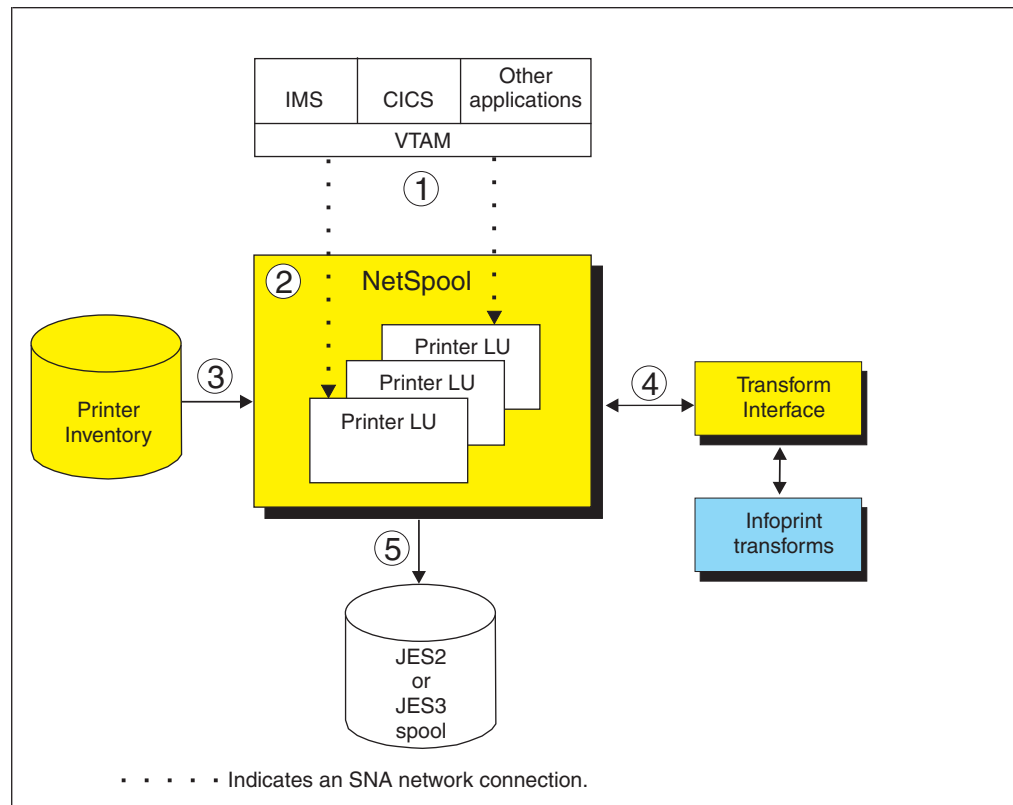


Figure 6. NetSpool system diagram

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 with JES allocation parameters that are 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.

More 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 by using one of the transform products that are described in "Transforms" on page 22. (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 because of 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 more 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 that is 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 with 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 on page 18 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.

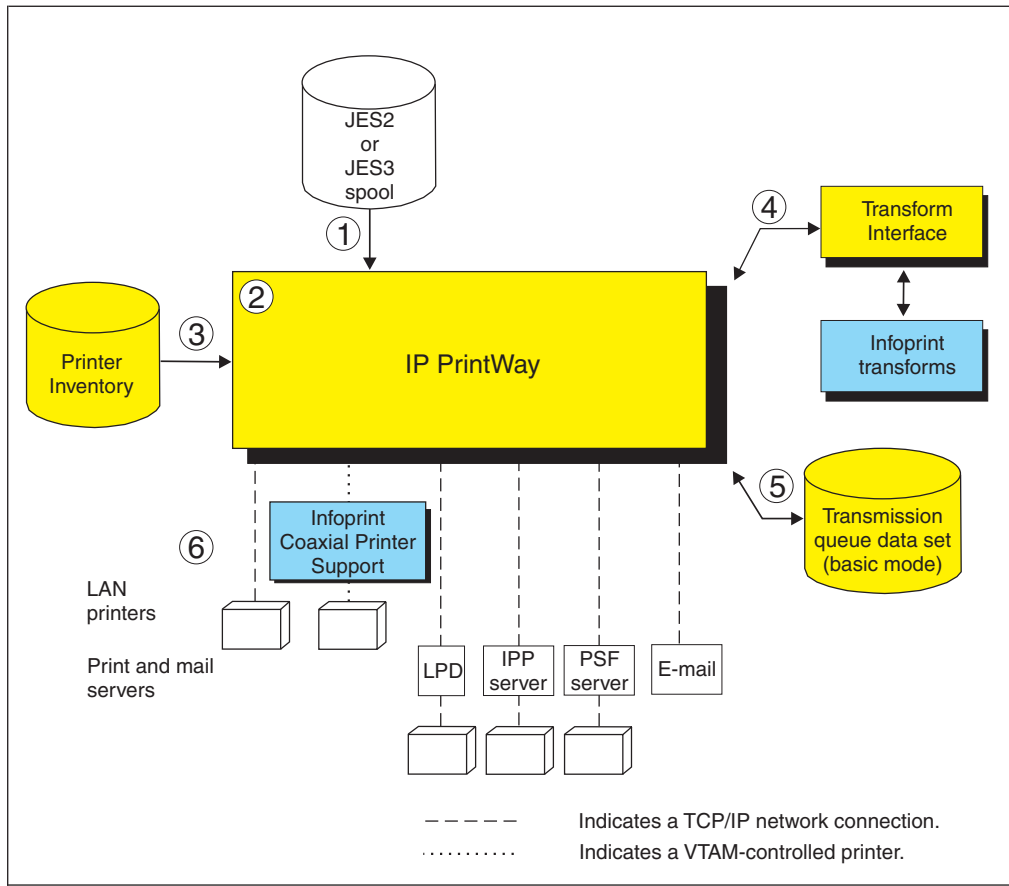


Figure 7. IP PrintWay system diagram

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 with 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 that is 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 is 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 with the carriage-control characters in line data, an FCB, or pagination attributes that are specified in the printer definition.
4. In most cases, IP PrintWay extended mode can automatically detect the input data format and transform data into the format that is 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 retained on the JES spool. In addition, you can use Infoprint Central to 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 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 with the protocol that is 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.

More functions that IP PrintWay provides include:

### **Retry**

IP PrintWay can try an unsuccessful transmission again 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 retransmissions, IP PrintWay tries an unsuccessful transmission automatically for a short 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 (PCL) and the direct sockets printing protocol. The administrator must request in the printer definition that IP PrintWay extended mode restart printing after the last successful page.

### **Retention of jobs on the JES spool**

After successfully transmitting each data set, or after it completes the requested number of transmission attempts, IP PrintWay can retain the data set on the JES spool forever or for a specific 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 by using one of the transform products that are described in "Transforms" on page 22.

### **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 retained. In addition,

you can use Infoprint Central to 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 are completed successfully, were retained because of 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 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 the name of the printer definition in the FSSDATA parameter or 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 Facilities (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 (PCL) 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**

You can use IP PrintWay to 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 it transmits data sets. IP PrintWay retains this order even if the transmission of the data sets must be tried again. 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 it transmits 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. You can use the Begin Data Set exit to 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 22.

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

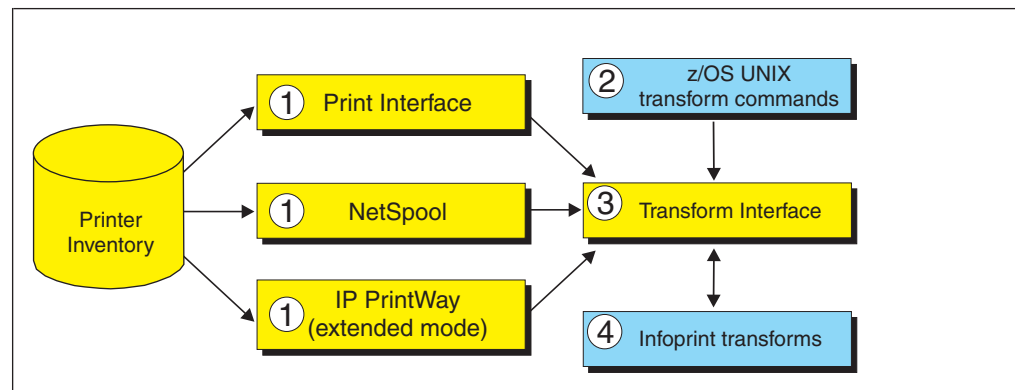


Figure 8. Transform Interface system diagram

1. When they process 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 with configuration information that is 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 5 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 5. 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 AIX, Linux, and Windows: Planning and Installing</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>

Table 5. Transforms that work with Infoprint Server (continued)

Transform product	Transform	For information, see:
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 IBM HTTP Server - Powered by Apache 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, including Time Sharing Option (TSO) output jobs. Plus, you can see more information about print jobs that Infoprint Server or PSF V4R4 and later 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 consists 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 various 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 that is 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 various 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.

### **Manage IP PrintWay job selection rules**

You can display the job selection rules that are defined in the Printer Inventory. IP PrintWay selects only print jobs whose properties match all of the criteria in a started job selection rule. In addition, you can start and stop IP PrintWay job selection rules to change which print jobs IP PrintWay processes.

### **Check system status**

You can see the status of all Infoprint Server daemons and tasks to help determine the cause of a printing problem.

More functions that Infoprint Central provides are:

### **Security**

You can authorize access to Infoprint Central by configuring IBM HTTP Server - Powered by Apache so that it uses an authorized user ID that is authenticated by a z/OS security manager, such as RACF. 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.



---

## 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 6 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 6. 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"><li>• Clients that use LPR to LPD protocol</li><li>• Clients that use Internet Printing Protocol (IPP)</li><li>• Windows clients that use Server Message Block (SMB) protocol</li><li>• z/OS UNIX <b>lp</b>, <b>lpstat</b>, and <b>cancel</b> commands</li><li>• The AOPPRINT JCL procedure</li><li>• Any Windows application that supports printing</li><li>• The Infoprint Server Application Programming Interface</li><li>• Batch z/OS job that specifies the Print Interface subsystem on a DD statement</li></ul>	Printer Inventory Manager Print Interface Infoprint Port Monitor for Windows (optional)	27 125 381
Receive print requests from VTAM applications (such as CICS and IMS), and create output data sets on the JES spool	Printer Inventory Manager NetSpool	27 179
Send output data sets from the JES spool to an ASCII printer, a print server, or an email destination	Printer Inventory Manager IP PrintWay, basic or extended mode Print Interface (if you use IP PrintWay basic mode and transforms)	27 207 125
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 PSF for z/OS	27 385
Transform data from one format to another	Printer Inventory Manager Transform Interface	27 167
Use Infoprint Central	Printer Inventory Manager Infoprint Central	27 339



---

## 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 that are running at the same or different levels on other systems. For sysplex considerations, see “/var/Printsrv directory” on page 33.

Table 7 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 7. Customization tasks for Printer Inventory Manager*

Task	Condition
“Identifying software requirements for Printer Inventory Manager” on page 28	Required
“Enabling the Infoprint Server feature” on page 28	Optional: If the Infoprint Server feature was not in your original z/OS order
“Updating the BPXPRMxx member of SYS1.PARMLIB” on page 28	Required
“Making Language Environment and C++ runtime libraries available” on page 31	Optional: If required libraries are not in LNKLIST
“Creating Infoprint Server directories” on page 32	Required
“Determining the Infoprint Server operating mode” on page 29	Optional: To activate z/OS V2R2 functions in Infoprint Server.
“Creating and editing the Infoprint Server configuration file (aopd.conf)” on page 37	Required
“Setting environment variables for the Printer Inventory Manager” on page 46	Required
“Setting up security for the Printer Inventory and operator commands” on page 52	Required
“Setting permissions for directories and executable files -- aopsetup” on page 59	Required
“Customizing startup and shutdown commands and procedures” on page 61	Optional: To specify environment variables or to start and stop Infoprint Server with JCL procedures

Table 7. Customization tasks for Printer Inventory Manager (continued)

Task	Condition
"Enabling Infoprint Server ISPF panels" on page 76	Required
"Enabling dynamic configuration" on page 78	Optional: To customize Infoprint Server without restarting Infoprint Server
"Editing the system configuration definition for Printer Inventory Manager" on page 87	Optional: To specify dynamic attributes
"Sending non-console messages to the hardcopy log" on page 94	Optional: To send messages to the hardcopy log
"Setting up the MVS system logger for the common message log" on page 98	Optional: To use the MVS system logger for the common message log
"Backing up and restoring the Printer Inventory" on page 114	Required
"Saving messages in the common message log in the z/OS UNIX file system" on page 120	Optional: To save the common message log
"Reducing the size of the common message log in the z/OS UNIX file system" on page 122	Optional: To free space in the z/OS UNIX file system
"Setting Workload Manager goals" on page 123	Optional: To prioritize Infoprint Server work

**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

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:

```
FILESTYPE 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*.

## Determining the Infoprint Server operating mode

The product function level that Infoprint Server operates with depends on whether dynamic configuration is enabled. *Dynamic configuration* is a function you use to change configuration attributes in the system configuration definition in the Printer Inventory while Infoprint Server is running, without stopping and restarting Infoprint Server. For more information about the advantages of using dynamic configuration, see “Enabling dynamic configuration” on page 78.

When dynamic configuration is not enabled, Infoprint Server operates with z/OS V1R13 functions. When dynamic configuration is enabled, Infoprint Server operates with z/OS V2R1 or z/OS V2R2 functions. You can use the operating mode function to specify whether Infoprint Server operates with z/OS V2R1 functions (the default) or with z/OS V2R2 functions. The operating mode is set with the **Operating mode** field on the ISPF System Configuration panel or the **ipsmode** attribute in the PIDU command.

When Infoprint Server operates with z/OS V2R1 functions, the Printer Inventory Manager daemon is always started when you start Infoprint Server; other daemons are started if they are specified in the Printer Inventory system configuration. When Infoprint Server operates with z/OS V2R2 functions, you can enter one command to start one daemon at a time or start multiple daemons with a JCL procedure. You can also display TSO output jobs in the list of JES print jobs from a search in Infoprint Central. See Appendix H, “Benefiting from new functions in z/OS V2R2,” on page 487 for all the functions that are available with z/OS V2R2.

The commands and JCL procedures you use to start and stop Infoprint Server daemons depend on whether dynamic configuration is enabled and if so, what the **Operating mode** field is set to. Table 8 shows the operating environment and commands that you use depending on how Infoprint Server is configured.

**Keep in mind:** Table 8 lists the commands that you enter from the UNIX command line to start and stop Infoprint Server; however, IBM strongly suggests that you use the JCL startup procedures to start Infoprint Server instead of running the start command directly from the z/OS UNIX command line. See “Customizing startup and shutdown commands and procedures” on page 61.

Table 8. Configurations that affect how Infoprint Server operates

Infoprint Server configuration	Operating environment	Commands	Display status
Dynamic configuration not enabled	<b>aopstart</b> EXEC, aopd.conf, and aopmsg.conf (see Note 1)	<b>aopstart</b> and <b>aopstop</b> (see Note 2)	
Dynamic configuration enabled with <b>Operating mode</b> set to z/OS 2.1	Printer Inventory system configuration, <b>aopstart</b> EXEC, and aopd.conf (see Note 3)	<b>aopstart</b> and <b>aopstop</b> (see Note 2)	<b>aopstat</b> (see Note 4)

Table 8. Configurations that affect how Infoprint Server operates (continued)

Infoprint Server configuration	Operating environment	Commands	Display status
Dynamic configuration enabled with <b>Operating mode</b> set to z/OS 2.2	Printer Inventory system configuration, <b>aopdemon</b> variables file, and aopd.conf (see Notes 3 and 5)	<b>aopdemon</b> and <b>aopsend</b> (see Note 6)	<b>aopstat</b>

#### Notes:

1. All environment variables are defined in **aopstart** EXEC, while configuration attributes are defined in aopd.conf and aopmsg.conf.
2. All daemons that are specified with the **start-daemons** attribute in aopd.conf are started. IBM suggests that you use the AOPSTART JCL procedure to start Infoprint Server. The **aopstop** command stops all Infoprint Server daemons or only specified daemons. You can also use the AOPSTOP procedure to stop Infoprint Server daemons.
3. When dynamic configuration is enabled, Infoprint Server automatically creates the system configuration definition in the Printer Inventory, which contains many environment variables and attributes from **aopstart** EXEC, aopd.conf, and aopmsg.conf. The attributes in the system configuration definition are used instead of the variables in **aopstart** EXEC and the attributes in aopd.conf. The aopmsg.conf is ignored.
4. The **aopstat** command can be used to display the status of Infoprint Server daemons and NetSpool tasks. You can also use the AOPSTAT JCL procedure.
5. The **aopdemon** command defaults to the same environmental variables that the sample **aopstart** EXEC specifies. To change the default values or define new environment variables, you create a user-defined variables file instead of editing **aopstart** EXEC. See “Creating an environment variables file (*envvar*)” on page 65.
6. The **aopdemon** command starts one Infoprint Server daemon, while the **aopsend** command stops either one or all daemons. IBM suggests that you use the AOPDEMON or AOPSTAR2 JCL procedures to start Infoprint Server. The AOPDEMON procedure starts one daemon and the AOPSTAR2 procedure starts multiple daemons. You can also use the AOPSTOP2 JCL procedure to stop one or all daemons.

## Setting the operating mode for z/OS V2R2 functions

To activate z/OS V2R2 functions in Infoprint Server:

1. Copy any customized environment variables in the **aopstart** EXEC to a user-defined variables file. See “Creating an environment variables file (*envvar*)” on page 65.
2. See “Editing JCL startup procedures” on page 67 for information about using the AOPDEMON JCL procedure. For the **aopdemon** command options and more examples of how to specify options on the AOPDEMON procedure, see z/OS *Infoprint Server Operation and Administration*.
3. Enable dynamic configuration if it is not already enabled. See “Steps for enabling dynamic configuration” on page 82. When dynamic configuration is enabled, Infoprint Server automatically creates the system configuration definition in the Printer Inventory.
4. If Infoprint Server is not already started, start it with the AOPSTART procedure. See “AOPSTART JCL procedure” on page 67.

5. Set the **Operating mode** field on the ISPF System Configuration panel to z/OS 2.2 or the **ipsmode** attribute in the PIDU command to z202. See *z/OS Infoprint Server Operation and Administration*.

**Tip:** When you change the operating mode, you do not need to restart Infoprint Server. However, because the **aopstart** command started all specified daemons, when the Printer Inventory Manager daemon, **aopd**, is shutdown the first time after the change, all daemons are shutdown. The next time you start Infoprint Server, you must start each daemon individually or use the AOPSTAR2 JCL procedure to start multiple daemons.

---

## 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 LNKST. 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 LNKST 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 LNKST with no adverse effects. If you do not add *hlq.SCEERUN2* to LNKST, 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 LNKST, do these steps:

- a. Add the *hlq.SCEERUN* and *hlq.SCLBDLL* libraries to the STEPLIBLIST facility of the BPXPRMxx *hlq.PARMLIB* member.

- b. 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
```

- c. 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
```

- d. Make this change in the **aopstart** EXEC if the **Operating mode** field on the ISPF System Configuration panel is set to z/OS 2.1 or in the *envvar* file for the **aopdemon** command if the **Operating mode** field is set to z/OS 2.2:

#### **aopstart** EXEC

Uncomment and edit this statement:

```
/*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'
```

#### *envvar* file

Add this statement:

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

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

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

For information, see “Editing the aopstart EXEC” on page 62 or “Creating an environment variables file (*envvar*)” on page 65.

- e. 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 77.
- f. Add the *hlq.SCEERUN* and *SCLBDLL* libraries to JCL that starts AOPBATCH. The AOPSTART, AOPDEMON, and AOPPRINT JCL procedures use AOPBATCH.
- g. 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 182 and “Customizing the IP PrintWay startup procedure (basic mode)” on page 332.

**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:

- 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

**Note:** Infoprint Server directories can be in a Hierarchical File System (HFS) or a z/OS File System (zFS). However, because a zFS has higher performance characteristics than an HFS and is the strategic file system for z/OS, you need to allocate a zFS rather than an HFS. For information about zFS, see *z/OS Distributed File Service zFS Administration*.



**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
- Writable only by users with an effective UID of 0

The permission bits, in octal, are 755.

If you want, 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 or the *envvar* file for the **aopdemon** command:

- 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 413.

## **/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 59

### **Guidelines:**

1. 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
  MOUNTPPOINT('/&SYSNAME./var/Printsrv')

```

The /var/Printsrv directory can be in a Hierarchical File System (HFS) or a z/OS File System (zFS) that is HFS-compatible. Because a zFS has higher performance characteristics than an HFS and is the strategic file system for z/OS, you need to allocate a zFS rather than an HFS. For information about zFS, see *z/OS Distributed File Service zFS Administration*.

2. 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.”
3. Do not change the owner or permissions of the /var/Printsrv directory after it is created. For a secure environment, this directory must be:
  - Owned by UID of 0
  - Readable and writable only by users with an effective UID of 0 or members of the AOPADMIN group
  - Executable by everyone
4. The /var/Printsrv directory is a working directory for Infoprint Server. Do not place any user directories or files in the /var/Printsrv directory.

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 are in the /var/Printsrv directory:

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

Other Printer Inventory files with the .v2db extension are 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 upgraded to Infoprint Server V2R2 from V1R11 or an earlier release, the /var/Printsrv directory might contain Printer Inventory Version 1 and Version 2 files. Infoprint Server V2R2 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 is 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 and data files. By default, most clients send the control file after they send 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:

1. 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.
2. 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.

#### **Tips:**

1. 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.
2. 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.
3. 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 122.

#### **/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:

1. Specify an alternate trace directory in the AOPTRACEDIR environment variable before you activate tracing.
2. 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 and editing the Infoprint Server configuration file (aopd.conf)

The Infoprint Server configuration file (aopd.conf) is a required file. You must create it before you start 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.

**Rule:** Do not change this attribute while Infoprint Server (including NetSpool, IP PrintWay basic mode, and Infoprint Central) or the Printer Inventory for PSF is running. If you change this attribute, you must stop and restart:

- All Infoprint Server daemons
- NetSpool
- IP PrintWay basic mode
- IBM HTTP Server - Powered by Apache (when running Infoprint Central)
- PSF for z/OS (when running Printer Inventory for PSF)

**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 78.

**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 you restart Infoprint Server.

**Default:** dynamic-configuration = no

**Example:** dynamic-configuration = yes

**inventory = inventory**

The name of the Printer Inventory and the Print Interface subsystem. Specify exactly four 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.

**Rule:** Do not change this attribute while Infoprint Server (including NetSpool, IP PrintWay basic mode, and Infoprint Central) or the Printer Inventory for PSF is running. If you change this attribute, you must stop and restart:

- All Infoprint Server daemons
- NetSpool
- IP PrintWay basic mode
- IBM HTTP Server - Powered by Apache (when running Infoprint Central)
- PSF for z/OS (when running Printer Inventory for PSF)

**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:**

1. 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 you restart Infoprint Server.
2. If you use Infoprint Central and change this attribute, restart the IBM HTTP Server - Powered by Apache.

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

**Example:** jes-name = JESX

**start-daemons= {[ipdd] [lpd] [netd] [outd] [ssid] [subd] [xfd]}**

The Infoprint Server daemons that are started when you use the **aopstart** command. This attribute is ignored when the **Operating mode** field on the ISPF Configuration panel is set to **z/OS 2.2**.

You can specify one or more of these daemon names that are 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 an Internet Protocol network or to email destinations.
- 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 daemon **aopd** always starts, regardless of which daemons you specify in this attribute. To start only the Printer Inventory Manager daemon, 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
```

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

1. You added value **ssid**.
2. Dynamic configuration is not enabled.
3. 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:**

1. In this example, all daemons start:  
start-daemons = {ippd lpd netd outd ssid subd xfd}
2. In this example, only the Printer Inventory Manager daemon starts:  
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 you restart 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 that is 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 that is 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 that is 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:

1. 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.
2. 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 you restart Infoprint Server.

**Default:** `ascii-codepage = IS08859-1`

**`ebcdic-codepage = codepage`**

The name of an IBM-supplied or custom EBCDIC code page that is 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:

1. 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.
2. 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 you restart 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 are removed from the JES spool.

You can specify a value 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 are removed from the JES spool.

#### Guidelines:

1. 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.
2. 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 34.
3. 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.
4. 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 122.
5. If you use **aoplogu** to view messages, the date and time format that is displayed in messages is controlled by the `TZ` and either the `LC_TIME` or `LC_ALL` environment variables. See “Setting environment variables for the Printer Inventory Manager” on page 46.  
  
If you use Infoprint Central to view messages, the date and time format for the messages is controlled by the operating system that runs your web browser.

#### Rules:

1. 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.

2. 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 you restart 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 that are specified in the **log-retention** attribute limits the amount of information that Infoprint Server keeps in the Historical Inventory.

Table 9 shows the number of documents 1 MB in the Historical Inventory can contain.

*Table 9. Number of documents in the Historical Inventory*

If print jobs on the JES spool contain:	1 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 1 MB

For example:

- Documents processed in one day = 100,000
- Days to retain information = 3 (**log-retention** = 3)
- Documents in 1 MB = 1350 documents

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

**Rules:**

1. 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.
2. 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 you restart 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 395.
- Enabling dynamic configuration, see “Enabling dynamic configuration” on page 78.

## 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 you start 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 45.

**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 have authority 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, in the **aopdemon** environment variables file, 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 AOPOPER groups” on page 54).

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

5. Change the permissions of the file so that it is readable by everyone and writable 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 by 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:

1. 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 wanted.
2. If a value contains blank characters or special characters (such as {}-> =, # &), enclose the value in single or double quotation marks.
3. Start comments with the # symbol.
4. Include blank lines, if wanted.

## 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 44.

**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 by 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 startup 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 you start 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
```

**Tip:** The **start-daemons** attribute is ignored when the **Operating mode** field on the ISPF Configuration panel is set to z/OS 2.2.

- b. Do not change the *base-directory*, *inventory*, or *xcf-group-qualifier* attributes while Infoprint Server is running.
3. If you add or change any of these attributes and you run Infoprint Central, restart the IBM HTTP Server - Powered by Apache:

- **base-directory**
- **inventory**
- **jes-name**

**Tip:** You can restart the IBM HTTP Server - Powered by Apache while Infoprint Server is stopped.

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

1. 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 wanted.
2. If a value contains blank characters or special characters (such as {}-> =, # &), enclose the value in single or double quotation marks.
3. Start comments with the # symbol.
4. Include blank lines, if wanted.

---

## 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 V2R2.0 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 413 for a summary of all the environment variables that Infoprint Server uses.

The Printer Inventory Manager and other Infoprint Server daemons use environment variables that are specified in these places:

#### **aopstart EXEC**

For a secure environment, Printer Inventory Manager uses environment variables set in this 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.

**Tip:** The **aopstart EXEC** is only used when the **Operating mode** field is not set to z/OS 2.2 on the ISPF System Configuration panel.

#### **aopdemon *envvar* file**

This is an optional, user-defined z/OS UNIX file or MVS data set that is used instead of **aopstart EXEC** when dynamic configuration is enabled and the **Operating mode** field is set to z/OS 2.2. When you use the **aopdemon** command in a UNIX System Services shell to start an Infoprint Server daemon, the daemon accepts certain environment variables from the shell. You can use the *envvar* file to define other environment variables for a daemon. The *envvar* file is specified with the **aopdemon** command or, if you use the AOPDEMON JCL procedure to start an Infoprint Server daemon, with the VARFILE DD statement.

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 62. For information about the *envvar* file, see “Creating an environment variables file (*envvar*)” on page 65.

To edit the `/etc/profile` file, you can use your preferred editor. For example, enter:

```
oedit /etc/profile
```

To set and export an environment variable, use the z/OS UNIX **export** command.

#### **Examples:**

1. 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
```
2. To display the value of an environment variable, use the z/OS UNIX **echo** command:

```
echo $PATH
```



Table 10 summarizes the variables that affect the Printer Inventory Manager, including the common message log and the **pidu** command. The 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 10. Where to define environment variables for the Printer Inventory Manager

Environment variable	Location where set		
	/etc/profile file	aopstart EXEC or <i>envvar</i> file	STDENV data set (AOPSTART JCL procedure)
_BPX_UNLIMITED_OUTPUT	No	Optional <sup>2</sup>	No
AOPCONF	Optional	Optional <sup>2</sup>	No
AOPMSG_CONF	Optional	Optional or No <sup>1</sup>	No
AOPRXF_CONF	Optional	Optional <sup>2</sup>	No
AOPTRACEDIR	No	Optional <sup>2</sup>	No
AOPVALIDATEDB	Optional	Optional	Optional
AOPXFD_CONF	Optional	Optional <sup>2</sup>	No
CLASSPATH	No	Optional <sup>2</sup>	No
JAVA_HOME	No	Optional <sup>2</sup>	No
LANG	Optional	Optional	Optional
LC_ALL	Optional	Optional	Optional
LC_CTYPE	Optional	Optional	Optional
LC_TIME	Optional	Optional	Optional
LIBPATH	Required	Optional <sup>2</sup>	No
MANPATH	Required	No	No
NLSPATH	Required	Optional <sup>2</sup>	No
PATH	Required	Optional <sup>2</sup>	No
TZ	Optional	Optional	Optional
1. This environment variable is Optional for the <b>aopstart</b> EXEC and No for the <i>envvar</i> file. 2. The default <b>aopstart</b> EXEC and <b>aopdemon</b> command sets these environment variables to the default values. You do not need to set them in the <b>aopstart</b> EXEC or <i>envvar</i> file if the default values are suitable.			

**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:

#### **\_BPX\_UNLIMITED\_OUTPUT**

Specifies that unlimited spooled output is processed for non-local spawn requests. The **aopstart** EXEC and **aopdemon** command set this variable to **\_BPX\_UNLIMITED\_OUTPUT=YES**. Do not remove this variable or change its value.

#### **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**.



**Default:** /etc/Printsrv/aopd.conf

#### AOPMSG\_CONF

The full path name of the Infoprint Server message configuration file, aopmsg.conf. If dynamic configuration is enabled, you did not create this configuration file or you created the configuration file in the default location, you do not need to set this environment variable.

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

#### AOPRXF\_CONF

The full path name of the Infoprint Server 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.

**Default:** /etc/Printsrv/aoprxf.conf

#### AOPTRACEDIR

The name of the trace directory. Set this environment variable if you did not create the trace directory in the default location. When you create the trace directory, the AOPADMIN group must have permission to write to the specified directory.

**Default:** *base-directory*/trace

If you use the default base directory, the default is /var/Printsrv/trace.

**Example:** AOPTRACEDIR=/mydirectory/trace

#### AOPVALIDATEDB

If this environment variable is set to any value, Infoprint Server checks internal databases for validity when it starts processing. 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

#### AOPXFD\_CONF

The full path name of the Infoprint Server transform configuration file (aopxfd.conf). Set this environment variable if you did not create the transform configuration file in the default location.

**Default:** /etc/Printsrv/aopxfd.conf

#### CLASSPATH

The full path names of IPP Server Java Archive (JAR) files.

If you installed Infoprint Server files in the default directory, /usr/lpp/Printsrv, you do not need to set this environment variable.

If you installed Infoprint Server files in a non-default directory, add

CLASSPATH=*/directory/classes/ipp.jar:*  
*/directory/classes/ippreal.jar:*  
*/directory/classes/ippserver.jar*

in the *envvar* file, where *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 that is used to locate Java files. JAVA\_HOME must specify the path of the 31-bit version of Java because the IPP Server does not support the 64-bit version. Set this environment variable if you did not create the Java file path in the default location.

**Default:** /usr/lpp/java/J7.1

**LANG**

The language that is 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 that is used to format time and date information in messages. This locale overrides the locale in the LC\_TIME variable.

**Rule:** In the IBM HTTP Server - Powered by Apache environment variables file (bin/envvars), do not specify a value that is associated with a code page other than IBM-1047.

**Examples:**

1. **aopstart** EXEC: LC\_ALL=Fr\_FR.IBM-297
2. bin/envvars file: LC\_ALL=Fr\_FR.IBM-1047

**Default:** C (also called POSIX)

**LC\_CTYPE**

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

**Rules:** In the IBM HTTP Server - Powered by Apache environment variables file (bin/envvars):

1. Do not specify the LC\_CTYPE environment variable.
2. 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 that is 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 that is 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`

The default is set in the **aopstart** EXEC and with the **aopdemon** command. 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 and with the **aopdemon** command. You must specify this environment variable in the `/etc/profile` file.

**PATH** The path that is used to locate executable files. This environment variable is required. If you installed Infoprint Server executable files 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 started.

**Default:** `/usr/lpp/Printsrv/bin:/bin`

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

**TZ** The time zone that is used to format date and time information in messages that are displayed by **aoplogu**.

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

#### 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 61. For information about the *envvar* file, see “Creating an environment variables file (*envvar*)” on page 65.
- 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 11 summarizes the RACF access that is required to do different functions. For more security requirements if you use Infoprint Central or the MVS system logger, see:

- “Setting up security for Infoprint Central” on page 352
- “Setting up security for log streams” on page 101

Table 11. 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"><li>• View: READ access</li><li>• View and update: UPDATE access</li></ul>	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"><li>• View: READ access</li><li>• View and update: UPDATE access only if AOP.CONFIGURATION is not defined</li></ul>	<ul style="list-style-type: none"><li>• View: READ access</li><li>• View and update: UPDATE access</li></ul>	Not required	Not required
Use the PIDU program to view and update the Printer Inventory except for the system configuration definition	<ul style="list-style-type: none"><li>• View: READ access</li><li>• View and update: UPDATE access</li></ul>	Not required	Required	Not required
Use the PIDU program to view and update the system configuration definition	<ul style="list-style-type: none"><li>• View: READ access</li><li>• View and update: UPDATE access only if AOP.CONFIGURATION is not defined</li></ul>	<ul style="list-style-type: none"><li>• View: READ access</li><li>• View and update: UPDATE access</li></ul>	Required	Not required

Table 11. 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 Infoprint Central to view PSF printers, printer definitions, and IP PrintWay job selection rules	READ access	Not required	Not required	Not required
Use the <b>aoplogu</b> 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
<b>Note:</b> 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 by 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 by 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:

1. If the AOP.ADMINISTRATOR profile does not exist in the PRINTSRV class, Infoprint Server checks the AOPADMIN profile in the FACILITY class. READ or higher access to the AOPADMIN profile is equivalent to UPDATE access to the AOP.ADMINISTRATOR profile.
2. 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 that you can use to restrict access to the system configuration definition separately from other definitions 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 and update other definitions in the Printer Inventory.

### Tips:

1. 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.
2. If the AOP.ADMINISTRATOR and AOP.CONFIGURATION profiles both do not exist in the PRINTSRV class, Infoprint Server checks the AOPADMIN profile in the FACILITY class. READ or higher access to the AOPADMIN profile is equivalent to UPDATE access to the AOP.ADMINISTRATOR and AOP.CONFIGURATION profiles.
3. 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 101.

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

Infoprint Server operators can start and stop Infoprint Server daemons with the start and stop commands from the z/OS UNIX command line or the JCL startup and shutdown 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 authorize 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 and 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 with 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 authorize group AOPADMIN to the AOP.ADMINISTRATOR resource profile or you can authorize 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 authorizes the AOPADMIN group to the AOP.ADMINISTRATOR profile in the PRINTSRV class:

```
PERMIT AOP.ADMINISTRATOR CLASS(PRINTSRV) ACCESS(UPDATE) ID(AOPADMIN)
SETOPTS 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 authorize group AOPCONF to the AOP.CONFIGURATION resource profile or you can authorize each user separately.

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

```
PERMIT AOP.CONFIGURATION CLASS(PRINTSRV) ACCESS(UPDATE) ID(AOPCONF)
SETOPTS 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)
SETOPTS 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. This step shows 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*. 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 AOOPER, AOPADMIN, or any another group that has an OMVS GID.

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

#### Guidelines:

- a. The user's default group must have a GID defined so that the user can use z/OS UNIX.

- b. For instructions about how to define a user ID for use with the JCL startup and shutdown procedures, see “Creating a user ID to assign to the JCL startup and shutdown procedures” on page 65.

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

If you use the JCL startup and shutdown 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 JCL startup 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 222.
  - 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 223.
  - If you use Infoprint Central, access to profiles in the OPERCMDS and JESSPOOL classes. For information about the type of access that is required, see “Setting up security for Infoprint Central” on page 352.
  - If you use IBM transforms from AFP to PCL, PDF, or PostScript, READ access to transform resources, such as AFP resource libraries, that are 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 101.

**Tip:** If you always use the JCL startup and shutdown procedures (and not the start and stop commands from the z/OS UNIX command line) 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 JCL startup and shutdown 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 that is 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 writable 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 (AOOPER).
- Administrative commands, such as **pidu**, are executable by members of the RACF group for Infoprint Server administrators (AOPADMIN).
- All Infoprint Server daemons have 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 54). 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 54). 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 52.
- 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.

These steps show how to run **aopsetup**:

1. Switch to an effective UID of 0.

**Example:** su

**Tip:** To use the z/OS UNIX **su** command, you must be authorized 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 must run **aopsetup** before you start Infoprint Server daemons. However, if you attempted to start Infoprint Server before you ran **aopsetup**, stop all Infoprint Server daemons before you do this step.

**Examples:**

- a. If you defined group AOPOPER for operators and group AOPADMIN for administrators, enter:  

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

```
/usr/lpp/Printsrv/bin/aopsetup AOPADMIN AOPADMIN
```
  - c. 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 AOPOPER 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 looks similar to the following sample output. Also, the owner ID of the /var/Printsrv directory and the executable files, which are shown as ROOT in the sample output, must 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 -n /var/Printsrv

drwxrwxr-x  12 0 256 ... .. /var/Printsrv
```
- ```
ls -n /usr/lpp/Printsrv/bin

...
-rwx--S---  1 0 4000 ... .. aopd
```

```

-rwsr-x--- 1 0 4001 ... .. aopdemon
-rwx--S--- 1 0 4000 ... .. aopipppd
-rwxr-x--- 1 0 4000 ... .. aoplogu
-rwx--S--- 1 0 4000 ... .. aoplpd

```

## 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:

- JCL startup and shutdown procedures
- Start and stop commands from the z/OS UNIX command line

In addition, you can use z/OS automation tools to start Infoprint Server daemons automatically during system initialization.

The user ID assigned to the JCL startup procedure or the user ID that runs the start command from the z/OS UNIX command line 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 352.
- 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 222 and “Authorizing IP PrintWay to write SMF type 6 records (extended mode)” on page 223.

**Guideline:** IBM strongly suggests that you use the JCL startup procedures to start Infoprint Server instead of running the start command directly from the z/OS UNIX command line for these reasons:

- With a JCL startup procedure, you can associate one user ID with the 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 a start 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*.

### Tips:

1. The JCL startup and shutdown procedures and the start and stop commands from the z/OS UNIX command line, 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
```
2. When you are starting Infoprint Server with z/OS V2R2 for the first time, and dynamic configuration is enabled in aopd.conf, you see message AOP192E if Infoprint Server cannot find a valid system configuration definition in the Printer Inventory. You must supply a system configuration definition before Infoprint Server can complete the startup. See “Restoring the Printer Inventory with the loadprinterinventory command” on page 120 or, if you do not have a

PIDU export file, use the ISPF System Configuration panel or the Printer Inventory Definition Utility (PIDU) to edit and save the system configuration definition. See *z/OS Infoprint Server Operation and Administration*.

Table 12 summarizes the customization tasks that are 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.

*Table 12. Customization tasks for startup and shutdown procedures*

Task	Condition
"Editing the aopstart EXEC"	Optional: To specify environment variables
"Creating an environment variables file ( <i>envvar</i> )" on page 65	Optional: To specify environment variables
"Creating a user ID to assign to the JCL startup and shutdown procedures" on page 65	Optional: To use the JCL startup and shutdown procedures
"Defining the JCL startup and shutdown procedures to RACF" on page 66	Optional: To use the JCL startup and shutdown procedures
"Editing JCL startup procedures" on page 67	Optional: To use the JCL startup procedures and change default settings
"Editing JCL shutdown procedures" on page 72	Optional: To use the JCL shutdown procedures and change default settings
"Starting Infoprint Server daemons automatically" on page 75	Optional: To start Infoprint Server daemons automatically

## 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 that are specified in the **start-daemons** attribute in the Infoprint Server configuration file (*aopd.conf*).

**Note:** When dynamic configuration is enabled and the **Operating mode** field is set to z/OS 2.2 on the System Configuration panel, **aopstart** EXEC is not used. Instead, you can specify environment variables in an *envvar* file for the **aopdemon** command. See "Creating an environment variables file (*envvar*)" on page 65.

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 413 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 that uses 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 can be authorized 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 already copied the **aopstart** EXEC to a new directory, follow the steps in “Steps for editing the **aopstart** EXEC” on page 64.

1. Switch to an effective UID of 0:

```
su
```

**Tip:** To use the z/OS UNIX **su** command, you must be authorized 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 must 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 54).

```
chgrp AOPOPER aopstart
```

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

```
oedit aopstart
```

Edit this file by using the IBM-1047 code page. For more 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 writable 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 on the PARM parameter in the AOPSTART procedure. For details, see "Editing JCL startup procedures" on page 67.

## 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 63.

**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 authorized 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 with your preferred editor and save your changes. For example:

```
oedit aopstart
```

See comments in the EXEC. Edit this file by using the IBM-1047 code page. For more 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 an environment variables file (*envvar*)

When dynamic configuration is enabled and the **Operating mode** field is set to z/OS 2.2 on the ISPF System Configuration panel, **aopstart** EXEC is not used. Instead, these default environment variables are used with the **aopdemon** command:

```
_BPX_UNLIMITED_OUTPUT=YES,  
PATH=/usr/lpp/Printsrv/bin:/bin,  
LIBPATH=/usr/lpp/Printsrv/lib,  
CLASSPATH=/usr/lpp/Printsrv/classes/ipp.jar:  
           /usr/lpp/Printsrv/classes/ippserver.jar:  
           /usr/lpp/Printsrv/classes/ippreal.jar,  
NLSPATH=/usr/lpp/Printsrv/%L/%N:/usr/lpp/Printsrv/En_US/%N:/usr/lib/nls/msg/%L/%N,  
JAVA_HOME=/usr/lpp/java/J7.1,  
AOPCONF=/etc/Printsrv/aopd.conf,  
AOPRXF_CONF=/etc/Printsrv/aoprxf.conf,  
AOPTRACEDIR=/var/Printsrv/trace,  
AOPXFD_CONF=/etc/Printsrv/aopxfd.conf
```

If you need to change the value of the environment variables from the default or specify more variables, you can create an *envvar* file, which is a user-defined z/OS UNIX file or MVS data set. For example, you might want to create the *envvar* file in /etc/Printsrv.

**Keep in mind:** When dynamic configuration is enabled, many values that used to be specified as environment variables are now attributes in the system configuration definition; you do not need to specify environment variables for them in the *envvar* file.

You specify an *envvar* file with the **aopdemon** command or the VARFILE DD statement in the AOPDEMON JCL procedure.

For information about authorizations required to use the *envvar* file, see z/OS *Infoprint Server Operation and Administration*.

1. In the *envvar* file, specify each environment variable in the format *variable=value*. For example:  
AOPTRACEON=1

### Rules:

- a. Specify one environment variable per line or record, starting in column 1.
  - b. Do not enclose *variable=value* or *value* in quotation marks.
  - c. Use a # symbol to start a comment. All data between the first “#” and the end of the line is ignored.
2. If an environment variable specifies a long variable string, you might need to define an MVS data set with a record length greater than 80. For example, you might need to specify these DCB attributes:
    - RECFM=VB
    - LRECL=255

## Creating a user ID to assign to the JCL startup and shutdown procedures

Do these steps to create a user ID that is assigned to the JCL startup and shutdown 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 52. 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:

```
ADDUSER AOPSTC OMVS(UID(user-identifier) PROGRAM('/bin/sh')) HOME(/tmp)
DFLTGRP(AOPOPER) NOPASSWORD
```

**Tips:**

- a. For *user-identifier*, specify an integer that is different from other UIDs in your installation. If the **Operating mode** field is set to z/OS 2.2 on the ISPF System Configuration panel and you use the VARFILE DD statement to specify environment variables, *user-identifier* must be 0.
  - b. 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 because of incorrect password attempts.
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 RACF user ID that is assigned to the JCL startup and shutdown procedures 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 JCL startup and shutdown procedures to RACF

If you use the JCL startup and shutdown 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 AOPDEMON.* STDATA(USER(AOPSTC) GROUP(AOPOPER))
RDEFINE STARTED AOPSTOP.* STDATA(USER(AOPSTC) GROUP(AOPOPER))
SETROPTS RACLIST(STARTED) REFRESH
```

**Tips:**

1. 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.
2. 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 JCL startup procedures

Three JCL procedures, AOPSTART, AOPDEMON, and AOPSTAR2, are included in SYS1.IBM.PROCLIB for starting Infoprint Server. AOPDEMON and AOPSTAR2 are used only when dynamic configuration is enabled and the **Operating mode** field is set to z/OS 2.2 on the ISPF System Configuration panel; otherwise, AOPSTART is used. AOPDEMON starts one daemon; the AOPSTAR2 procedure starts one or multiple daemons. You can use the AOPSTAR2 procedure to start all daemons in the same way you use the AOPSTART procedure.

**Note:** During installation, the procedures might be copied to another data set in the PROCLIB concatenation.

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

- Read the data set or file specified in the STDENV or VARFILE DD statement.
- 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:

### MVS data set

Give the AOPOPER group RACF authorization to read to the data set.

### UNIX file

Make AOPOPER the group that is associated with this file; give group members permission to read the file.

## AOPSTART JCL procedure

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

**Tip:** When dynamic configuration is enabled, you can use this startup procedure only when the **Operating mode** field is set to z/OS 2.1 on the ISPF System Configuration panel.

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 413. For security purposes, you can specify only these 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 62. You can also choose to specify these environment variables in the **aopstart** EXEC. However, if you specify them in the **aopstart** EXEC, do not specify them in the AOPSTART procedure.

**Rules for environment variables:**

1. 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.
2. If you specify environment variables in an MVS data set that have long variable strings, you might need to define a record length greater than 80. For example, you might need to specify these DCB attributes:
  - RECFM=VB
  - LRECL=255
3. 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 for using the AOPSTART procedure:**

1. If you are using JES2 dynamic PROCLIBs, refresh the PROCLIBs before you restart Infoprint Server.

**Example:** To refresh PROCLIBs, enter this JES2 command:

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

2. 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 you restart Infoprint Server.

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

```
START AOPSTOP  
START AOPSTART
```

Figure 9 on page 69 shows the AOPSTART procedure. If you need more information about the AOPBATCH program, see Appendix E, “AOPBATCH program,” on page 457.

```

/*-----
/*
/* 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 9. AOPSTART startup procedure — SYS1.IBM.PROCLIB(AOPSTART)

## AOPDEMON JCL procedure

The AOPDEMON JCL procedure invokes the **aopdemon** command to start Infoprint Server. AOPDEMON starts one daemon.

**Tip:** This startup procedure is used only when dynamic configuration is enabled and the **Operating mode** field is set to z/OS 2.2 on the ISPF System Configuration panel.

You must edit the AOPDEMON JCL procedure in these situations:

- You need to specify the directory path for the **aopdemon** command you created for your installation in the PARM parameter of the EXEC statement.
- You need to specify the *envar* file if you change the following default environment variables or you specify more variables:

```

_BPX_UNLIMITED_OUTPUT=YES,
PATH=/usr/lpp/Printsrv/bin:/bin,
LIBPATH=/usr/lpp/Printsrv/lib,
CLASSPATH=/usr/lpp/Printsrv/classes/ipp.jar:
           /usr/lpp/Printsrv/classes/ippserver.jar:
           /usr/lpp/Printsrv/classes/ippreal.jar,
NLSPATH=/usr/lpp/Printsrv/%L/%N:/usr/lpp/Printsrv/En_US/%N:/usr/lib/nls/msg/%L/%N,
JAVA_HOME=/usr/lpp/java/J7.1,
AOPCONF=/etc/Printsrv/aopd.conf,
AOPRXF_CONF=/etc/Printsrv/aoprxf.conf,
AOPTRACEDIR=/var/Printsrv/trace,
AOPXFD_CONF=/etc/Printsrv/aopxfd.conf

```

You can specify the *envvar* file with a STDENV or VARFILE DD statement. Use STDENV to specify only these environment variables: AOPDUMPON, AOPVALIDATEDB, AOPTRACEBYTES, AOPTRACEMAXFILES, AOPTRACEON, AOPTRACEON\_DATA, AOPTRACEON\_FILTER, AOPTRACEON\_MSGLOG, LANG, LC\_APP, LC\_CTYPE, LC\_TIME, and TZ. Otherwise, if you specify any other variables, use VARFILE and assign UID 0 to the started task.

- You need to specify the type of daemon if you are using the procedure to start a specific daemon. Otherwise, you can add the type when you enter the MVS START command. For example,  
START AOPDEMON,TYPE=SSI

**Rule:** The Printer Inventory daemon (**aopd**) must be running before you start any other daemons.

Figure 10 on page 71 shows the AOPDEMON procedure that Infoprint Server provides in SYS1.IBM.PROCLIB.

```

/*-----
/*
/* AOPDEMON - This procedure starts an Infoprint Server daemon.
/*           The USERID assigned to this proc must be a member of
/*           the AOPOPER group.
/*
/*           TYPE the 3 character "type" of the daemon to start.
/*           Valid choices are: AOP IPP LPD NET OUT SSI SUB WSM XFD
/*
/*-----
//AOPDEMON PROC TYPE=
/*-----
/* The AOPBATCH PARM includes the path of the aopdemon command.
/* If Infoprint Server was not installed in /usr/lpp/Printsrv then
/* this path must be changed appropriately.
/*-----
//AOPDEMON EXEC PGM=AOPBATCH,REGION=512M,TIME=NOLIMIT,
//          PARM='/usr/lpp/Printsrv/bin/aopdemon -t&TYPE'
//STDIN DD DUMMY
//STDOUT DD PATH='/dev/console' or '/dev/operlog'
//STDERR DD PATH='/dev/console' or '/dev/operlog'
/*-----
/* To change values of default environment variables or to specify
/* more variables.
/*
/* 1. Define one of these:
/*    - An MVS PDS member (or sequential data set), such as:
/*      //dd_name DD DSN=hlq.stdev(member),DISP=SHR
/*    - A UNIX System Services file, such as:
/*      //dd_name DD PATH='/etc/Printsrv/env.conf',PATHOPTS=ORDONLY
/*
/* 2. In this file, specify each environment variable in the format
/*    VARIABLE=VALUE by following these rules:
/*    - Specify one VARIABLE=VALUE per line, starting in column 1.
/*    - Use a # symbol to start a comment.
/*    For example:
/*    # Environment variables to turn on tracing and set time zone.
/*    AOPTRACEON=1
/*    TZ=MST7MDT           # Mountain Time Zone
/*
/* 3. Use dd_name=STDENV for AOPTRACEBYTES, AOPTRACEMAXFILES, any
/*    variable whose name begins with AOPTRACEON, AOPVALIDATEDB,
/*    AOPDUMPON, LANG, LC_APP, LC_CTYPE, LC_TIME, and TZ. Otherwise,
/*    use dd_name=VARFILE and assign UID 0 to the started task.
/*-----
/* dd_name DD DSN=USER.PARMLIB(ENVVARS),DISP=SHR

```

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

## AOPSTAR2 JCL procedure

The AOPSTAR2 JCL procedure invokes the AOPDEMON JCL procedure one or more times to individually start one or more Infoprint Server daemons. You can use this procedure in the same way you use the AOPSTART procedure to start all daemons.

**Tip:** This startup procedure is used only when dynamic configuration is enabled and the **Operating mode** field is set to z/OS 2.2 on the ISPF System Configuration panel.

To start all Infoprint Server daemons that are specified in the procedure, enter this MVS command:

```
START AOPSTAR2
```

You can edit the AOPSTAR2 procedure and comment out those daemons you do not want to run. Figure 11 shows the AOPSTAR2 procedure that Infoprint Server provides in SYS1.IBM.PROCLIB.

```

/*-----
/*
/*  AOPSTAR2 - This procedure starts the Infoprint Server daemons.
/*             The USERID executing this proc must be a member of
/*             the AOPOPER group.
/*             The first step MUST be coded. The rest are optional
/*             and depend on which Infoprint Server services you
/*             wish to use.
/*-----
//AOPSTAR2 PROC
//*
//INVENTORY EXEC AOPDEMON,TYPE=AOP Required
//*
//INETPRNT EXEC AOPDEMON,TYPE=IPP
//*
//LPDEMON  EXEC AOPDEMON,TYPE=LPD
//*
//NETSPOOL EXEC AOPDEMON,TYPE=NET
//*
//PRINTWAY EXEC AOPDEMON,TYPE=OUT
//*
//GUIDEMON EXEC AOPDEMON,TYPE=SSI
//*
//SUBSYSTEM EXEC AOPDEMON,TYPE=SUB
//*
//WORKSELC EXEC AOPDEMON,TYPE=WSM
//*
//TRANSFRM EXEC AOPDEMON,TYPE=XFD
//

```

Figure 11. AOPSTAR2 startup procedure — SYS1.IBM.PROCLIB(AOPSTAR2)

## Editing JCL shutdown procedures

These JCL procedures are included in SYS1.IBM.PROCLIB for stopping Infoprint Server:

### AOPSTOP

Stops all daemons or specified daemons.

### AOPSTOP2

Stops one or all daemons. AOPSTOP2 is used only when dynamic configuration is enabled. It must be used when the **Operating mode** field is set to z/OS 2.2 on the ISPF System Configuration panel; otherwise, AOPSTOP is used. You can use the AOPSTOP2 procedure to stop all daemons in the same way that you use the AOPSTOP procedure.

### Guidelines:

1. During installation, the procedures might be copied to another data set in the PROCLIB concatenation.
2. If you specify an MVS data set or z/OS UNIX file in a DD statement in the AOPSTOP or AOPSTOP2 procedure, the user ID associated with the shutdown procedure must be authorized to:
  - Read the data set or file specified in the STDENV DD statement (AOPSTOP only).



- 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:

#### **MVS data set**

Give the AOOPER group RACF authorization to read or write to the data set.

#### **z/OS UNIX file**

Make AOOPER the group that is associated with this file; give group members permission to read the file.

3. You can also enter this MVS STOP command to stop an Infoprint Server daemon instead of using AOPSTOP or AOPSTOP2:

`STOP jobname_id`

*jobname\_id*

The descriptive name of an Infoprint Server daemon. When dynamic configuration is enabled, you can use the AOPSTAT JCL procedure or **aopstat** command from the UNIX command line to list the job IDs for the daemons.

For more information about the MVS STOP command, see *z/OS MVS System Commands*.

## **AOPSTOP JCL procedure**

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

**Tip:** When dynamic configuration is enabled, you can use AOPSTOP only when the **Operating mode** field is set to z/OS 2.1 on the ISPF System Configuration panel. Otherwise, you must use the AOPSTOP2 procedure.

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 where it exists 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*.

Figure 12 shows the AOPSTOP procedure. If you need more information about the AOPBATCH program, see Appendix E, “AOPBATCH program,” on page 457.

```

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

```

Figure 12. AOPSTOP shutdown procedure — SYS1.IBM.PROCLIB(AOPSTOP)

## AOPSTOP2 JCL procedure

The AOPSTOP2 procedure invokes the **aopsend** command to stop one or all Infoprint Server daemons. You can use this procedure in the same way you use the AOPSTOP procedure to stop all daemons.

**Tip:** This shutdown procedure is used only when dynamic configuration is enabled. When the **Operating mode** field is set to z/OS 2.2 on the ISPF System Configuration panel, you must use AOPSTOP2 instead of the AOPSTOP procedure.

You must edit the AOPSTOP2 JCL procedure in these situations:

- If the **aopsend** command is not in the default directory /usr/lpp/Printsrv/bin, specify the directory in the PARM parameter on the EXEC statement.
- You want to specify one daemon to stop rather than stopping all daemons (the default). Otherwise, you can add the daemon member name when you enter the MVS START command. For example:

```
START AOPSTOP2,MEMB=XFDIBM00
```

**Rule:** Do not stop the Printer Inventory Manager daemon while other Infoprint Server daemons are still running.

Figure 13 shows the AOPSTOP2 procedure provided in SYS1.IBM.PROCLIB.

```

/*-----
/*
/* AOPSTOP2 - This procedure stops Infoprint Server daemons.
/*           The USERID assigned to this proc must be a member of
/*           the AOPOPER group.
/*
/*           MEMB the 8 character XCF-name of the daemon to stop
/*           the value "ALL" stops all READY daemons
/*
/*-----
//AOPSTOP  PROC MEMB=ALL
/*-----
/* The AOPBATCH PARM includes the path of the aopsend command.
/* If Infoprint Server was not installed in /usr/lpp/Printsrv then
/* this path must be changed appropriately.
/*-----
//AOPSTOP  EXEC PGM=AOPBATCH,
//          PARM='/usr/lpp/Printsrv/bin/aopsend -c STOP -m&MEMB'
//STDIN    DD    DUMMY
//STDOUT   DD    SYSOUT=*
//STDERR   DD    PATH='/dev/console' or '/dev/operlog'

```

Figure 13. AOPSTOP2 shutdown procedure — SYS1.IBM.PROCLIB(AOPSTOP2)

## Starting Infoprint Server daemons automatically

To start Infoprint Server daemons automatically during system initialization, you can use this method:

- z/OS automation tools to run the AOPSTART, AOPDEMON, or AOPSTAR2 procedure.

### Rules:

1. Start the Printer Inventory Manager daemons before you start IP PrintWay basic mode FSAs or the NetSpool task.
2. 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*.
3. Start TCP/IP, and make sure that TCP/IP finished initialization, before you start these Infoprint Server daemons: LPD (**aoplpd**), the IPP Server daemon (**aopippd**), and the IP PrintWay extended mode daemons (**aopoutd** and **aopwsmd**).

If TCP/IP does not finish 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. errno2=0x74b30296

```

**Tip:** If you use the startup 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.

---

## Editing the JCL status procedure

The AOPSTAT JCL procedure invokes the **aopstat** command to display the status of Infoprint Server daemons and NetSpool tasks. AOPSTAT is included in SYS1.IBM.PROCLIB; it is used only when dynamic configuration is enabled.

**Note:** During installation, the procedure might be copied to another data set in the PROCLIB concatenation.

Figure 14 shows the AOPSTAT procedure. If you need more information about the AOPBATCH program, see Appendix E, “AOPBATCH program,” on page 457.

```
/*-----  
/*  
/* AOPSTAT - This procedure allows the console operator to display  
/*           the status of the Infoprint Server daemons.  
/*  
/*           You may need to specify OPTIONS if Infoprint Server is  
/*           running on a different processor within the sysplex.  
/*  
/*-----  
//AOPSTAT  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 must be changed to reflect the installation directory.  
/*-----  
//STEP0001 EXEC PGM=AOPBATCH,  
//          PARM='/usr/lpp/Printsrv/bin/aopstat -c &OPTIONS'  
//STDIN    DD  DUMMY  
//STDOUT   DD  SYSOUT=(,)  
//STDERR   DD  PATH='/dev/console' or '/dev/operlog'  
/*-----  
/* STDENV is used to specify environment variables.  
/*  
/* A UNIX System Services file is usually specified, for example:  
/* //STDENV DD  PATH='/etc/Printsrv/env.conf',PATHOPTS=ORDONLY  
/* but an MVS data set can be used, for example:  
/* //STDENV DD  DSN=h1q.stdenv,DISP=SHR  
/*  
/* This file or data set should contain VARIABLE=VALUE pairs  
/* starting in column 1. Comment lines start with a #.  
/* For example:  
/* # An unnecessary variable since this is the default location.  
/* AOPCONF=/etc/Printsrv/aopd.conf  
/*-----  
/* STDENV DD  DSN=USER.PARMLIB(ENVVARS),DISP=SHR
```

Figure 14. AOPSTAT procedure — SYS1.IBM.PROCLIB(AOPSTAT)

---

## 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 is translated to code page IBM-939.

## Defining libraries in the TSO logon procedure

If you followed the instructions in the ISPF Setup sections in the *z/OS V2R2 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 ISPLLIB, depending on the language in which you want to receive messages:

**English**

AOP.SAOPMENU

**Japanese**

AOP.SAOPMJPN

- 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 LNKLIB, 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 V2R2 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 you use 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**

`/usr/lpp/Printsrv/En_US/%N`

**Japanese**

`/usr/lpp/Printsrv/Ja_JP/%N`

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

You use the dynamic configuration function to 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 configure Infoprint Server to start and stop individual daemons.
- You can benefit from new functions in Infoprint Server that require dynamic configuration. For example, you can use 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 the AOP.CONFIGURATION or AOP.ADMINISTRATOR profile does 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”
- How to set up security for the system configuration definition, see “Setting up security for the Printer Inventory and operator commands” on page 52.
- How to enable dynamic configuration, see “Steps for enabling dynamic configuration” on page 82
- How to view and edit the system configuration definition, see “Editing the system configuration definition for Printer Inventory Manager” on page 87.

## 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 13 lists the dynamic attributes, the name of the field on the Infoprint Server System Configuration ISPF panel that you use to specify the attribute, and whether you need to restart any Infoprint Server daemons before a change you make can take effect.

Table 13. 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
aopipd-max-thread-tasks	Other	IPP Server daemon (aopipd)



Table 13. Dynamic attributes and ISPF field names (continued)

Dynamic attribute	ISPF field	Restart these daemons
<b>aoplpd-max-thread-tasks</b>	Other	Infoprint Server LPD ( <b>aoplpd</b> )
<b>aopnetd-max-thread-tasks</b>	Other	NetSpool daemon ( <b>aopnetd</b> )
<b>aopoutd-max-thread-tasks</b>	Other	IP PrintWay extended mode daemons ( <b>aopoutd</b> , <b>aopwsmd</b> )
<b>aopssid-max-thread-tasks</b>	Other	Infoprint Central daemon ( <b>aopssid</b> )
<b>aopsubd-max-thread-tasks</b>	Other	Subsystem daemon ( <b>aopsubd</b> )
<b>aopwsmd-max-thread-tasks</b>	Other	IP PrintWay extended mode daemons ( <b>aopoutd</b> , <b>aopwsmd</b> )
<b>applid</b>	APPL ID	None
<b>ascii-codepage</b>	ASCII	None
<b>blank-truncation-classes</b>	Blank truncation classes	None
<b>console-name</b>	Extended MCS console name	Infoprint Central daemon ( <b>aopssid</b> )
<b>ebcdic-codepage</b>	EBCDIC	None
<b>hardcopy-messages</b>	Send messages to hardcopy log	None
<b>hardcopy-message-list</b>	Message list for hardcopy log	None
<b>ignore-dcf-routing-errors</b>	Ignore DEST, CLASS, FORMS routing errors	None
<b>ipp-port-number</b>	IPP port number	None
<b>ipsmode</b>	Operating mode	None
<b>job-prefix</b>	Job ID prefix	None
<b>log-retention</b>	Log retention period (days)	None
<b>lpd-port-number</b>	LPD port number	None
<b>logstream-name</b>	Log stream name	None
<b>mail-do-not-add-suffixes</b>	Do not add suffixes	None
<b>mail-preserve-suffixes</b>	Preserve suffixes	None
<b>mail-use-first-address</b>	Send email to address of first document	None
<b>mailer-options</b>	Application options	None
<b>mailer-path-name</b>	Application path	None
<b>max-historical-inventory-size</b>	Maximum Historical Inventory size (MBs)	None
<b>max-thread-tasks</b>	Other	None
<b>netspool-use-fixed-jobid</b>	Other	NetSpool daemon ( <b>aopnetd</b> )
<b>netspool-use-unaltered-jobid</b>	Other	NetSpool daemon ( <b>aopnetd</b> )
<b>select-work-from-hold-queue</b>	Select work from hold queue	IP PrintWay extended mode daemons ( <b>aopoutd</b> , <b>aopwsmd</b> )



Table 13. Dynamic attributes and ISPF field names (continued)

Dynamic attribute	ISPF field	Restart these daemons
<b>smf-recording</b>	Write SMF records	None
<b>suppress-post-unix-filter-formatting</b>	Suppress formatting after UNIX filter	None

Table 14 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 14. 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 <code>aopd.conf</code> and <code>aopmsg.conf</code> files	Equivalent environment variable in <b>aopstart</b> EXEC
<b>allow-all-characters-in-line-data</b>	None	ALLOW_ALL_CHARACTERS_IN_LINE_DATA
<b>aopipdd-max-thread-tasks</b>	None	AOPIPPD_MAXTHREADTASKS
<b>aoplpd-max-thread-tasks</b>	None	AOPLPD_MAXTHREADTASKS
<b>aopnetd-max-thread-tasks</b>	None	AOPNETD_MAXTHREADTASKS
<b>aopoutd-max-thread-tasks</b>	None	AOPOUTD_MAXTHREADTASKS
<b>aopssid-max-thread-tasks</b>	None	AOPSSID_MAXTHREADTASKS
<b>aopsubd-max-thread-tasks</b>	None	AOPSUBD_MAXTHREADTASKS
<b>aopwsmd-max-thread-tasks</b>	None	AOPWSMD_MAXTHREADTASKS
<b>applid</b>	None	AOP_APPLID
<b>ascii-codepage</b>	<b>ascii-codepage</b> ( <code>aopd.conf</code> )	None
<b>blank-truncation-classes</b>	None	AOP_BLANK_TRUNCATION_CLASSES
<b>console-name</b>	<b>console-name</b> ( <code>aopd.conf</code> )	None
<b>ebcdic-codepage</b>	<b>ebcdic-codepage</b> ( <code>aopd.conf</code> )	None
<b>hardcopy-messages</b>	<b>hardcopy-messages</b> ( <code>aopmsg.conf</code> )	None
<b>hardcopy-message-list</b>	<b>hardcopy-message-list</b> ( <code>aopmsg.conf</code> )	None
<b>ignore-dcf-routing-errors</b>	None	AOP_IGNORE_DCF_ROUTING_ERRORS
<b>ipp-port-number</b>	<b>ipp-port-number</b> ( <code>aopd.conf</code> )	None
<b>ipsmode</b>	None	None
<b>job-prefix</b>	<b>job-prefix</b> ( <code>aopd.conf</code> )	None
<b>log-retention</b>	<b>log-retention</b> ( <code>aopd.conf</code> )	None
<b>lpd-port-number</b>	<b>lpd-port-number</b> ( <code>aopd.conf</code> )	None
<b>logstream-name</b>	None	None
<b>mail-do-not-add-suffixes</b>	None	AOP_MAIL_DO_NOT_ADD_SUFFIX
<b>mail-preserve-suffixes</b>	None	AOP_MAIL_PRESERVE_SUFFIXES
<b>mail-use-first-address</b>	None	AOP_MAIL_USE_FIRST_ADDRESS
<b>mailer-options</b>	None	AOPMAILER_OPTIONS
<b>mailer-path-name</b>	None	AOPMAILER

Table 14. 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 <i>aopd.conf</i> and <i>aopmsg.conf</i> files	Equivalent environment variable in <i>aopstart EXEC</i>
<b>max-historical-inventory-size</b>	<b>max-historical-inventory-size</b> ( <i>aopd.conf</i> )	None
<b>max-thread-tasks</b>	None	AOP_MAXTHREADTASKS
<b>netspool-use-fixed-jobid</b>	None	AOPNETD_USE_FIXED_JOBID
<b>netspool-use-unaltered-jobid</b>	None	AOPNETD_USE_UNALTERED_JOBID
<b>select-work-from-hold-queue</b>	None	AOP_SELECT_WORK_FROM_HOLD_QUEUE
<b>smf-recording</b>	<b>smf-recording</b> ( <i>aopd.conf</i> )	None
<b>suppress-post-unix-filter-formatting</b>	None	AOP_SUPPRESS_POST_UNIX_FILTER_FORMATTING
<b>Note:</b> These attributes do not apply to IP PrintWay basic mode: <b>aopoutd-max-thread-tasks</b> , <b>applid</b> , <b>ignore-dcf-routing-errors</b> , <b>ipsmode</b> , <b>mail-do-not-add-suffixes</b> , <b>mail-preserve-suffixes</b> , <b>mail-use-first-address</b> , <b>mailer-path-name</b> , <b>mailer-options</b> , and <b>select-work-from-hold-queue</b> . 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.		

**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 87.
- All attributes that you can specify in the system configuration definition, see Appendix D, “Configuration attributes,” on page 437.

## 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 and editing the Infoprint Server configuration file (*aopd.conf*)” on page 37.

## Steps for enabling dynamic configuration

These steps show how to enable dynamic configuration.

**Before you begin:**

- Make sure that you do not want to 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 that is 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.

These steps show how to enable dynamic configuration:

1. Edit the Infoprint Server configuration file (aopd.conf) and save the file:
  - a. Specify this attribute:  
**dynamic-configuration = yes**
  - b. 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**
2. 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
```

3. Start Infoprint Server.

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

```
START AOPSTART
```

**Tip:** When you are starting Infoprint Server with z/OS V2R2 for the first time, and dynamic configuration is enabled in aopd.conf, you see message AOP192E if Infoprint Server cannot find a valid system configuration definition in the Printer Inventory. You must supply a system configuration definition before Infoprint Server can complete the startup. See “Restoring the Printer Inventory with the loadprinterinventory command” on page 120 or, if you do not have a

PIDU export file, edit and save the system configuration definition (see “Steps for editing the system configuration definition” on page 91).

4. (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.
5. If you use Infoprint Central, restart the IBM HTTP Server - Powered by Apache.

**Tip:** You can restart the HTTP server while Infoprint Server is stopped.

6. 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.
7. (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 you specify **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:**

- a. For a list of the dynamic attributes, see Table 14 on page 81.
  - b. To comment out a dynamic attribute, type the # symbol in the first column of the line with the attribute.
  - c. 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.
8. (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 you specify **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.

**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 52.
- How to edit the `aopd.conf` file, see “Steps for editing the Infoprint Server configuration file (`aopd.conf`)” on page 45.
- How to edit the `aopmsg.conf` file, see “Editing the message configuration file (`aopmsg.conf`)” on page 97.

- Which attributes in the `aopd.conf` and `aopmsg.conf` files and which environment variables have equivalent attributes in the system configuration definition, see Table 14 on page 81.
- How to determine whether dynamic configuration is enabled, see “Steps for determining whether dynamic configuration is enabled” on page 86.
- How to back up the Printer Inventory, see “Backing up and restoring the Printer Inventory” on page 114.

## Steps for disabling dynamic configuration

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

- If you changed 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.

These steps show how to disable dynamic configuration:

1. Run the **pidu** command to display all attributes that are stored 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 45.
- How to edit the `aopmsg.conf` file, see “Editing the message configuration file (`aopmsg.conf`)” on page 97.
- Dynamic attributes, see “Dynamic attributes” on page 79.
- How to back up the Printer Inventory, see “Backing up and restoring the Printer Inventory” on page 114.

## 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:

Option not enabled

2. If Infoprint Server is not running, view the `aopd.conf` file. If the **dynamic-configuration=yes** attribute is set, dynamic configuration is enabled or will be enabled the next time that 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 45.

## 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.

Infoprint Server automatically creates the system configuration definition when dynamic configuration is enabled. In some cases, you might see message AOP192E when you start Infoprint Server if Infoprint Server cannot find a valid system configuration definition in the Printer Inventory. You must supply a system configuration definition before Infoprint Server can complete the startup. See “Restoring the Printer Inventory with the loadprinterinventory command” on page 120 or, if you do not have a PIDU export file, edit and save the default system configuration definition (“Steps for editing the system configuration definition” on page 91).

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 that is 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 that is 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 that is 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:

1. 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.

2. 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 you restart Infoprint Server.

**Default:** `ascii-codepage = IS08859-1`

**`ebcdic-codepage = codepage`**

The name of an IBM-supplied or custom EBCDIC code page that is 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:**

1. 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.
2. 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 you restart 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 Central displays for print jobs and printers. Also, the number of days for which Infoprint Central displays print jobs that are 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 that is specified in the system logger log stream definition is used instead.

You can specify a value 0 - 59. A value of 0 means that Infoprint Central cannot display any messages and cannot display information about print jobs are 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:

1. 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.
2. If you use the system logger for messages and also use Infoprint Central, the value needs to 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.
3. 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.
4. 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 34.
5. 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.
6. 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 122.
7. If you use **aoplogu** to view messages, the date and time format that is displayed in messages is controlled by the TZ and either the LC\_TIME or LC\_ALL environment variables. See “Setting environment variables for the Printer Inventory Manager” on page 46.  
  
If you use Infoprint Central to view messages, the date and time format for the messages is controlled by the operating system that runs your web browser.

#### Rules:

1. 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.
2. 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 you restart 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 that are specified in the **log-retention** attribute limits the amount of information that Infoprint Server keeps in the Historical Inventory.

Table 15 shows the number of documents 1 MB in the Historical Inventory can contain.

*Table 15. Number of documents in the Historical Inventory*

If print jobs on the JES spool contain:	1 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 1 MB

For example:

- Documents processed in one day = 100,000
- Days to retain information = 3 (**log-retention** = 3)
- Documents in 1 MB = 1350 documents

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

**Rules:**

1. 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.
2. 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 you restart 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 78.
  - 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.

- 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.

These steps show how to edit the system configuration definition:

1. On the main Infoprint Server ISPF panel, select **8 System** and press Enter.

```

                                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 not enabled, this error message is displayed on the  
| panel:  
| Option not enabled

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.
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
      ssid      subd      xfd

```

### General:

```

Operating mode . . . . . 1 1. z/OS 2.1 2. z/OS 2.2
Job ID prefix . . PS
_ Allow all characters in line data

```

### 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

```

### 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 52.
- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 78.
- How to back up the Printer Inventory, see “Backing up and restoring the Printer Inventory” on page 114.
- 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 437.

---

## 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:

1. 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.
2. 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 you restart 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:**

1. 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.
2. If your installation uses the MPFLSTxx Parmlib member for message automation, you might want to specify the same message IDs in MPFLSTxx.
3. If your installation uses an automated operations tool, the tool can process these messages.

**Rules:**

1. If you specify this attribute, also specify **hardcopy-messages = list**.
2. 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.
3. 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 you restart Infoprint Server.

**Default:** None.

**Example:** hardcopy-message-list={A0P3614I A0P3803E}

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

Infoprint Server: System Configuration			
Messages:			
Send messages to hardcopy log	..... 3	1. All	2. List 3. None
Message list for hardcopy log	.....		(more)

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).”
- How to edit an existing message configuration file (aopmsg.conf), see “Editing the message configuration file (aopmsg.conf)” on page 97.
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 91.
- 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 97.

**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.

To create the message configuration file:

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

To use the **su** command, you must be authorized 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:

- a. 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 /etc/profile file and the **aopstart** EXEC. For information, see “Setting environment variables for the Printer Inventory Manager” on page 46.
  - b. 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 must 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 AOPOPER groups” on page 54).

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

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

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

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

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

Edit this file by 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 you restart. 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:

1. Use lowercase characters for the attribute names. Use uppercase or lowercase characters for the values.
2. Use blank characters before or after the equal sign, if wanted.
3. Start comments with the # symbol.
4. Include blank lines, if wanted.

**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 406.
- 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 91.

## 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 with your preferred editor and save your changes. For example, enter:

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

Edit this file by 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:

1. Use lowercase characters for the attribute names. Use uppercase or lowercase characters for the values.
2. Use blank characters before or after the equal sign, if wanted.
3. Start comments with the # symbol.
4. Include blank lines, if wanted.

**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 406.
- How to create the aopmsg.conf file, see “Creating the message configuration file (aopmsg.conf)” on page 96.
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 91.

---

## 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 16 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 16. Customization tasks for system logger*

Task	Condition
“Deciding whether to use the MVS system logger” on page 99	Required
“Deciding the type of log stream” on page 99	Required
“Deciding on a naming convention for log streams” on page 100	Required
“Determining how long to retain messages” on page 100	Required
“Determining the size of log-stream data sets and interim storage” on page 101	Required
“Setting up security for log streams” on page 101	Required
“Defining log streams to the MVS system logger” on page 104	Required

Table 16. Customization tasks for system logger (continued)

Task	Condition
"Activating a log stream in Infoprint Server" on page 109	Required
"Switching from one log stream to another" on page 111	Optional: To switch to a new log stream
"Deactivating a log stream in Infoprint Server" on page 112	Optional: To deactivate a log stream
"Handling system logger errors" on page 113	Optional: To handle any logger errors
"Viewing messages in the log stream" on page 113	Optional: To view messages

## Deciding whether to use the MVS system logger

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

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

If you do not set up the MVS system logger for the common message log, the log, by default, is in the z/OS UNIX file system that is specified in the **base-directory** attribute in the `aopd.conf` configuration file if 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 numerous 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 78

## 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 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 log stream, Infoprint Central can display messages only 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 is 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:

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

AOP.MSG

2. 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 can 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 (<http://www.ibm.com/redbooks>) *System Programmer's Guide to: System Logger*, SG24-6898

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 Coupling Facility sizer (<http://www.ibm.com/systems/support/z/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 that are 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 that is specified in the NAME parameter of the DEFINE LOGSTREAM statement in the LOGR policy. If you plan to define more than one log stream that starts 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 JCL startup procedure and not a z/OS UNIX start 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 JCL startup 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 it displays 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 authorize 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 352.

## 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 that is specified in the NAME parameter of the DEFINE LOGSTREAM statement in the LOGR policy. If you plan to define more than one log stream that starts 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 JCL startup procedure and not a z/OS UNIX start 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 JCL startup 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 it displays 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 authorize 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 352.

## 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 that 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.

To define a DASD-only log stream for the common message log:

1. Define the log stream in the LOGR policy. Table 17 on page 105 shows some of the parameters that you specify to define a log stream. Figure 15 on page 106 shows sample JCL for running the logger data administrative utility, IXCMIAPU, to define a DASD-only log stream.
2. (Optional) Verify that the log stream is 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 17 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 17, you can specify any values that are suitable for your installation.

*Table 17. 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 100.
DASDONLY	Type of log stream	DASDONLY(YES)
RETPD	Number of days that the system logger retains messages before it deletes them	See “Determining how long to retain messages” on page 100.
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 4 KB blocks, of the offload data sets for the log stream	See “Determining the size of log-stream data sets and interim storage” on page 101.
STG_SIZE	Size, in 4 KB blocks, of the staging data sets for the log stream	See “Determining the size of log-stream data sets and interim storage” on page 101.
HIGHOFFLOAD	High offload threshold	HIGHOFFLOAD(80)
LOWOFFLOAD	Low offload threshold	LOWOFFLOAD(0)
DIAG	Allows collection of diagnostic information	YES

Figure 15 on page 106 shows sample JCL to define a DASD-only log stream that is 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 15. 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 101.
- 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.

To define a coupling-facility log stream:

1. Define the structure in the LOGR policy. Table 18 on page 107 shows some of the parameters that you specify to define a structure in the LOGR policy. Figure 16 on page 108 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 17 on page 108 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 19 on page 107 shows some of the parameters that you specify to define a log stream in the LOGR policy. Figure 18 on page 109 shows sample JCL to run the IXCMIAPU utility to define the log stream.

4. (Optional) Verify that the log stream is 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 18 lists the coupling-facility structure parameters and the suggested values for Infoprint Server.

*Table 18. 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 that are written to all the log streams that use this coupling facility structure	AVGBUFSIZE(512)
MAXBUFSIZE	Size, in bytes, of the largest log block that can be written to log streams that are allocated to the coupling facility	MAXBUFSIZE(8192)

Table 19 lists the coupling-facility log stream parameters that have special considerations for the Infoprint Server. For parameters that are not shown in Table 19, you can specify any values that are suitable for your installation.

*Table 19. 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 100.
STRUCTURE	Name of the structure	Name of a structure that is defined to the system logger
DASDONLY	Type of log stream	DASDONLY(NO)
RETPD	Number of days that the system logger retains messages before it deletes them	See “Determining how long to retain messages” on page 100.
AUTODELETE	Automatic delete option	AUTODELETE(YES)
LS_SIZE	Size, in 4 K blocks, of the offload data sets for the log stream	See “Determining the size of log-stream data sets and interim storage” on page 101.
STG_SIZE	Size, in 4 K blocks, of the staging data sets for the log stream	See “Determining the size of log-stream data sets and interim storage” on page 101.
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.

Table 19. Coupling-facility log stream parameters for Infoprint Server (continued)

Parameter	Meaning	Suggested value
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 16 shows sample JCL to define a coupling-facility structure that is 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 16. Sample JCL to define a coupling-facility structure to the system logger

Figure 17 shows sample JCL to define a coupling-facility structure that is 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 17. Sample JCL to define a coupling-facility structure in the CFRM policy

Figure 18 on page 109 shows sample JCL to define a coupling-facility log stream that is 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 18. 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 103.
- The DISPLAY LOGGER command, see *z/OS MVS System Commands*.

## Activating a log stream in Infoprint Server

After you define 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.

To activate the log stream:

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
```

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, to connect to the log stream and write at least one message to it when you use IP PrintWay basic mode, you must use one of these methods to submit a print job:

- z/OS UNIX **lp** command
- TCP/IP **lpr** command
- Windows operating system
- NetSpool

**Related information:** For information about:

- How to set up security for a log stream, see “Setting up security for log streams” on page 101.
- How to define a log stream to the system logger, see “Defining log streams to the MVS system logger” on page 104.
- How to edit the system configuration definition, see “Editing the system configuration definition for Printer Inventory Manager” on page 87.
- How to handle system logger errors, see “Handling system logger errors” on page 113.
- 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.

To switch from one log stream to another:

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** field.

**Example:** This example shows a portion of the panel:

```

Edit                               Infoprint Server: System Configuration
Command ==>

Messages:

Log stream name . . . . . AOP.MSG.NEW
```

- b. 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 101.
- How to define a log stream to the system logger, see “Defining log streams to the MVS system logger” on page 104.
- How to edit the Infoprint Server system configuration definition, see “Editing the system configuration definition for Printer Inventory Manager” on page 87.
- How to handle system logger errors, see “Handling system logger errors” on page 113.
- 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.

To deactivate a log stream:

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 91.
- The **Log retention period (days)** field, see “Editing the system configuration definition for Printer Inventory Manager” on page 87.

## 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 Infoprint Server writes 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 112.
- How to activate a log stream, see “Activating a log stream in Infoprint Server” on page 109.
- 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 that is 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 that is 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 that are 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 system:name inventory:name [job:id]
[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 important that you back up the Printer Inventory regularly. You also need to back up the Printer Inventory after you create or modify 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)

When you try to start Infoprint Server with z/OS V2R2, and dynamic configuration is enabled, you see message AOP192E if Infoprint Server cannot find a valid system configuration definition in the Printer Inventory. You can use the **loadprinterinventory** command to restore the Printer Inventory with a valid system configuration definition.

**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 that is 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 restore the Printer Inventory, restart Infoprint Server, and run with the restored Printer Inventory for a few days.
4. 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:
  - Run the **pidu** command.

**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
```

- If you see message AOP192E, run the **loadprinterinventory** command. See “Restoring the Printer Inventory with the loadprinterinventory command” on page 120.
6. (Optional) 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 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 /\hlq.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 /\hlq.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 restore the Printer Inventory, restart Infoprint Server, and run with the restored Printer Inventory for a few days.
4. 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 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\(\)\' ' >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 that is 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 before the newly created data set:

```
pidu -c "export /\ 'hlq.INV.BACKUPS\(+1)\' ";
cat /\ 'hlq.INV.BACKUPS\(\)\' "
```

To create a new generation data set and then display that data set, you would need to run this command:

```
pidu -c "export /\ 'hlq.INV.BACKUPS\(+1)\' ";
cat /\ 'hlq.INV.BACKUPS\(+1)\' "
```

- If you see message AOP192E, run the **loadprinterinventory** command as a batch job. For example:

```
//AOPRINV JOB MSGLEVEL=(1,1)
//STEP1 EXEC PGM=AOPBATCH,PARM='/loadprinterinventory //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
/*
```

See “Restoring the Printer Inventory with the **loadprinterinventory** command” on page 120.



6. (Optional) 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.

## Restoring the Printer Inventory with the **loadprinterinventory** command

When you try to start Infoprint Server with z/OS V2R2 and dynamic configuration is enabled, you see message AOP192E if Infoprint Server cannot find a valid system configuration definition in the Printer Inventory:

AOP192E The Configuration object is missing in the inventory.

When you see message AOP192E, use the **loadprinterinventory** command to restore the Printer Inventory.

**Before you begin:** Make sure that you are a member of the AOPADMIN group and have UPDATE access to the AOP.ADMINISTRATOR resource profile in the PRINTSRV class.

To supply a valid system configuration definition and restore the Printer Inventory, run this command:

```
loadprinterinventory pidu_export
```

where *pidu\_export* is the name of the PIDU export file. If you do not have a PIDU export file, use the ISPF System Configuration panel or the Printer Inventory Definition Utility (PIDU) to edit and save the system configuration definition. See *z/OS Infoprint Server Operation and Administration*.

Otherwise, contact your service representative in the IBM Support Center, or use your electronic link with IBM service for assistance.

You can also run the **loadprinterinventory** command as a batch job. For example:

```
//LOADPINV JOB 000,CLASS=2,NOTIFY=&SYSUID
//*
//DEFAULT OUTPUT CLASS=T,DEFAULT=YES,JESDS=ALL
//*
//STEP0001 EXEC PGM=AOPBATCH,
//PARM='/loadprinterinventory /u/dewolfe/base/pidu.export.july15'
//STEPLIB DD DSN=DEWOLFE.DEVB.Z22.LINKLIB,DISP=SHR
//STDERR DD SYSOUT=(,)
//STDOUT DD SYSOUT=(,)
//STDIN DD DUMMY
//
```

---

## 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 52.

**Before you begin:** Create a separate file system for the saved message log files; for example, mounted at /var/Printsrv.backups. Do not save messages in the /var/Printsrv directory because the saved message log files can be large. For information about creating and mounting file systems, see *z/OS UNIX System Services Planning*

## 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 authorized 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 and editing the Infoprint Server configuration file (aopd.conf)” on page 37.)

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:

1. 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.
2. 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.
3. 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 to a Transaction Name Group (TNG) by using Transaction Class Groups. If you want to list each Infoprint Server daemon individually, see “Customizing startup and shutdown commands and procedures” on page 61 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 169.)

**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 that is running on the system. In most cases, 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, with 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 7.

Table 20 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 20. Customization tasks for Print Interface*

Task	Condition
“Identifying software requirements for Print Interface” on page 126	Required
“Common customization for Print Interface” on page 126	Required
“Customizing the z/OS UNIX printing commands” on page 140	Optional: To use <b>lp</b> command
“Customizing the AOPPRINT JCL procedure” on page 141	Optional: To use AOPPRINT procedure
“Customizing the Print Interface LPD” on page 142	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
“Customizing the z/OS SMB server to use Print Interface” on page 149	Optional: To print from a Windows client with SMB protocol
“Customizing the Print Interface IPP Server” on page 150	Optional: To print from an IPP client
“Customizing SAP R/3” on page 155	Optional: To print from SAP R/3
“Customizing the Print Interface subsystem” on page 163	Optional: To use the Print Interface subsystem

Table 21 describes other components of Infoprint Server that you might need to customize. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies.

*Table 21. Customization tasks for Infoprint Server components*

Task	Condition
Chapter 3, "Customizing the Printer Inventory Manager," on page 27	Required
Chapter 5, "Customizing Transform Interface," on page 167	Optional: To transform data
Chapter 7, "Customizing IP PrintWay," on page 207	Optional: To print on printers in a TCP/IP or SNA network
Chapter 9, "Installing and customizing the Infoprint Server Windows client," on page 381	Optional: To submit print requests from the Infoprint Port Monitor for Windows

**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 a JCL startup procedure or a start command from the z/OS UNIX command line.

## Identifying software requirements for Print Interface

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

- To submit print jobs with 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, IBM 31-bit SDK for z/OS, Java Technology Edition, V7.1 (5655-W43) is required. For information, see Java products for z/OS.

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 22. Common customization tasks for Print Interface

Task	Condition
“Customizing z/OS Communications Server”	Required
“Updating the BPXPRMxx member of SYS1.PARMLIB” on page 129	Required
“Working with JES spool limits” on page 129	Required
“Editing the aopd.conf file for Print Interface” on page 130	Optional: To change default configuration
“Writing a filter program” on page 131	Optional: To inspect or modify input data streams
“Customizing JES to specify no blank truncation” on page 137	Optional: To print XML data with PSF
“Setting environment variables for Print Interface” on page 137	Optional: To print line data that has unprintable characters
“Editing the system configuration definition for Print Interface” on page 138	Optional: To specify dynamic attributes

## Customizing z/OS Communications Server

You must customize and start the IP component (TCP/IP) of z/OS Communications Server if you plan to do one of these:

- Submit print jobs to Print Interface from remote systems.
- Use IP PrintWay to transmit print jobs to remote printers or email destinations.

This section describes considerations that are related to customizing TCP/IP.

Print Interface and IP PrintWay extended mode support IPv6. If you enable or disable IPv6 while Infoprint Server daemons are running, you must stop and restart the LPD (**aop1pd**), the IPP Server daemon (**aopipdd**), and the IP PrintWay extended mode daemons (**aopoutd** and **aopwsmd**).

For complete information about how to customize TCP/IP, see:

- *z/OS V2R2.0 Communications Server: IP Configuration Guide*
- *z/OS V2R2.0 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:

1. If you change any of the values in the *hlq*.PROFILE.TCPIP data set, restart the Communications Server so that your changes take effect.
2. For considerations that apply if you also run IP PrintWay, see “Customizing z/OS Communications Server” on page 241.

**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 TCPRCVBFRSIZE 32768 TCPSENDBFRSIZE 32768

**INTERVAL** *minutes* **SENDGARBAGE TRUE**

Number of minutes TCP waits after it receives 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:** 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 contains data.

**TCPRCVBFRSIZE** *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 that is 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

**Example:** This example comments out a line that reserves port 515 for the TCP/IP LPD. A semicolon in the first column indicates that 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 - 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 that 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 143.

## Updating the BPXPRMxx member of SYS1.PARMLIB

When you use Print Interface, you 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 200 for processes that are used by both NetSpool and Print Interface. If you run IP PrintWay, add an extra 50 (extended mode) or an extra 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 (**aoppipd**)
- 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 and **aopdemon** command define `_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` 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:**

1. 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.
2. The job ID that Print Interface returns to the job submitter does not contain this prefix.

### **Rules:**

1. 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.
2. 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 you restart 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 395.
- How to edit the `aopd.conf` file, see “Steps for editing the Infoprint Server configuration file (`aopd.conf`)” on page 45.
- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 78.

## Writing a filter program

### Programming interface information

This information documents programming-interfaces that you can use to write programs to obtain the services of Infoprint Server.

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
- Stop 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 is 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 133 and “Writing a UNIX filter” on page 134.

#### Options:

1. 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 267 and “End Data Set exit” on page 287.
2. 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*.
3. 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*

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:

1. 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).

2. 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 in the example 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, such as:

```
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 135 and *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 **aopfilter.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 **aopfilter.so** filter.

Both files are in directory **/usr/lpp/Printsrv/samples**. For more information about **aopfilter.so**, see “Filters provided with Infoprint Server” on page 137.

To customize the **aopfilter.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 are already converted to the printer code page.
- Name of the printer code page that is 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 that is 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 creates 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 19 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 19. Sample UNIX filter

## Rules:

1. z/OS UNIX Services restrict the UNIX filter files that can be used by z/OS UNIX set-user-ID and set-group-ID privileged programs. A UNIX filter that is defined as a REXX exec now requires that the program control extended attribute is turned on; otherwise, you see an error message. To set the program control extended attribute, enter:

```
extattr +p pathname
```

where *pathname* is the REXX exec.

2. 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 that is 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 you are coding a filter program:

- You need to 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 an Infoprint Serverabend. 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 obtains 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 is already 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 you modify a DLL filter, you must restart the Infoprint Server daemons to load the new version of the filter if Infoprint Server already uses 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 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.

End programming interface information

## 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 need to 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

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:**

1. 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.
2. 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.
3. 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:**

1. If dynamic configuration is enabled, specify the equivalent **allow-all-characters-in-line-data** attribute in the system configuration definition.
2. 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 62
- The LANG, LIBPATH, NLSPATH, and PATH variables, see "Setting environment variables for the Printer Inventory Manager" on page 46

## 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 zeros, 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:**

1. 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.
2. 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.
3. 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:**

1. 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.
2. If dynamic configuration is not enabled, specify the equivalent AOP\_ALLOW\_ALL\_CHARACTERS\_IN\_LINE\_DATA environment variable in the **aopstart** EXEC.
3. 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:**

1. 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.
2. The job ID that Print Interface returns to the job submitter does not contain this prefix.

**Rules:**

1. 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.
2. 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 you restart 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:

```
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 78.
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 91.

---

## 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.

### AOPTIONS

Specifies a string of job attributes and values that are to be in effect for each **lp** command. The **lp** command includes the attributes that are specified in this environment variable before the attributes specified, if any, on the **-o** option of the **lp** command. Because the **lp** command reads the value of the AOPTIONS environment variable before the options you specify on the command line, a user can override the values of this variable. This variable is optional. Use this environment variable to specify job attributes that are constant for all print jobs. For the format of the job attributes and values, see *z/OS Infoprint Server User's Guide*.



## AOPPATH

The directory that contains attribute files that are 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 that is named in either LPDEST or PRINTER overrides the printer that is 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 included in SYS1.IBM.PROCLIB. However, during installation, AOPPRINT might be 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 20 shows the AOPPRINT procedure.

```
/*-----  
/* AOPPRINT - z/OS Print Server batch print procedure  
/*-----  
/*AOPPRINT PROC PRINTER='p1',OPTIONS=,OUTCLASS='*',ERRCLASS='*'  
//LP      EXEC PGM=AOPBATCH,  
// PARM='/lp -d &PRINTER -o "&OPTIONS" //DD:SYSIN'  
//STDOUT  DD  SYSOUT=&OUTCLASS  
//STDERR  DD  SYSOUT=&ERRCLASS  
/* STDENV may point to a dataset containing environment variables.  
/*      Builtin values will work for the default installation.  
/*STDENV  DD  DSN=USERID.JCL(ENVVARS),DISP=SHR
```

Figure 20. 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 do not change the name, the default is **p1**. The administrator must 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 LNKLIB, 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
  - **lptrafp** 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 operating system 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 23. Customization tasks for Print Interface LPD

Task	Condition
"Customizing z/OS Communications Server (TCP/IP)"	Required
"Editing the aopd.conf file for the LPD" on page 145	Required
"Editing the system configuration definition for the LPD" on page 146	Optional: To specify dynamic attributes
"Using the LPD compatibility filter" on page 147	Optional: To obtain formatting similar to z/OS Communications Server LPD
"Customizing LPR clients" on page 147	Optional: To print from a remote client

**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 a JCL startup procedure or a start command from the z/OS UNIX command line 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 only the Print Interface LPD or 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" on page 144.

### 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 128.
- 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 128.
- 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 147.

## 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 each LPD on a separate TCP/IP stack or 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 that is 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 or *envvar* file as described in “Editing the aopstart EXEC” on page 62 or “Creating an environment variables file (*envvar*)” on page 65. 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 V2R2.0 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 by 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 you restart.

**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.

**Tip:** 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:

1. Make sure that the port is not used by any other service on the same TCP/IP stack.
2. 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 127.

3. 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.
4. 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 you restart 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 45.
- All attributes in the `aopd.conf` file, see Appendix B, “Infoprint Server files,” on page 395.
- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 78.

## 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.

**Tip:** 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:**

1. Make sure that the port is not used by any other service on the same TCP/IP stack.
2. 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 127.



3. 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.
4. 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 you restart Infoprint Server.

**Default:** `lpd-port-number = 515`

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 78.
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 91.

## Using the LPD compatibility filter

The LPD compatibility filter, `lpd_compat.so`, supports 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 that is 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 that you specified in **lpd-port-number** attribute. The default port is 515.
- The name of the printer definition for the target printer that is 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 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 is in this range, which is defined by RFC 1179. However, the Print Interface LPD does not require that the LPR is 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 sends 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 timeout value on an AIX system" shows an example of how to configure an AIX print queue to send the control file first.

## **Increasing the timeout value on an AIX system**

When you are 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 timeout value that is used by the AIX system. The timeout 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 acknowledgment only after data is transformed to a format accepted by the printer and written successfully to the JES spool, the LPD might send an acknowledgment only after several minutes. If the document contains several gigabytes of data that needs to be transformed, the LPD might send an acknowledgment 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 timeout value of 90 seconds. IBM suggests that you change this default to a long timeout value when you are 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 by using the standard Windows Server Message Block (SMB) protocol. Windows users can print on any printer that is 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 381.

Before Windows users can print with 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 be 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 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 that is named `aopapi.dll` in one of the directories that is 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 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.

**Related tasks:**

- Run a JCL startup procedure or a start command from the z/OS UNIX command line 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 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 25 on page 155. 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.

Table 24 summarizes the tasks to customize the IPP Server. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies to your installation.

*Table 24. Customization tasks for Print Interface IPP Server*

Task	Condition
“Ensuring sufficient memory is available to start the IPP Server” on page 151	Required
“Editing the aopd.conf file for the IPP Server” on page 151	Required
“Setting environment variables for the IPP Server” on page 152	Optional: To change the default configuration
“Editing the system configuration definition for the IPP Server” on page 153	Optional: To specify dynamic attributes
“Customizing the IPP workstation client” on page 153	Required

**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 a JCL startup procedure or a start command from the z/OS UNIX command line 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 that you increase the region size depends on how you start Infoprint Server:

### JCL startup procedure

Specify a region size of 200 MB in the REGION parameter on the EXEC statement in the JCL startup procedure. If the REGION parameter is not specified, the default region size that is defined for your installation is used.

### Start command

If you enter a start 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:

1. This statement sets the maximum region size to 200 MB:

```
MAXASSIZE(209715200)    /* 200*1024*1024 = 200MB      */
```

2. This command sets the maximum region size to 200 MB:

```
SETOMVS MAXASSIZE=209715200
```

### Tips:

1. 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.
2. Be careful when you are using the SHRLIBRGNSIZE OMVS parameter in a 31-bit environment because it limits the size of usable regions across the system.
3. 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 a start 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.

When you are using **aopstart** 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 you restart Infoprint Server.

**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:**

1. Make sure that the port is not used by any other service on the same TCP/IP stack.
2. 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.
3. 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 you restart 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 45.
- All attributes in the aopd.conf file, see Appendix B, “Infoprint Server files,” on page 395.
- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 78.

## 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, see “Editing the aopstart EXEC” on page 62 or “Creating an environment variables file (*envvar*)” on page 65.

### 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 that is used to locate Java files. The IPP Server can use Java 7.1 .

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 this directory:  
/usr/lpp/java/J7.1.

**Example:** JAVA\_HOME=/usr/lpp/java/J7.1

## 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:

1. Make sure that the port is not used by any other service on the same TCP/IP stack.
2. 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.
3. 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 you restart Infoprint Server.

**Default:** ipp-port-number = 631

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 78.
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 91.

## Customizing the IPP workstation client

To submit a print request over the Internet with 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 that is 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 that is 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 that is 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 is encrypted. Therefore, the user must 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 25 on page 155 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 25. IPP attributes and corresponding Infoprint Server attributes

IPP job attribute	Infoprint Server job attribute
<b>copies</b>	<b>copies</b>
<b>job-name</b>	<b>title-text</b>
<b>job-priority</b>	<b>jes-priority</b>
<b>requesting-user-name</b>	<b>owner<sup>1</sup></b>
<p>1. JES does not authorize 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 that is displayed by SDSF, the IPP Server uses the owner name as the job name when it allocates data sets on the JES spool.</p>	

## Customizing SAP R/3

Print Interface lets SAP R/3 users print on any printers that are 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 that 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 complete successfully or unsuccessfully. The Callback daemon also can provide notification to SAP R/3 applications that are 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 11 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 13 shows this configuration.

Table 26 on page 156 summarizes the customization tasks when an SAP R/3 application server runs on the same z/OS system as Infoprint Server. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies to your installation.

Table 26. Customization tasks for Print Interface when SAP R/3 application server runs on same z/OS system

Task	Condition
"Defining an SAP user for the callback daemon"	Required
"Creating the SAP callback daemon configuration file (aopsapd.conf)" on page 157	Required
"Setting environment variables for the SAP OMS" on page 159	Required
"Defining a ROMS and LOMS to the SAP R/3 application server" on page 159	Required
"Modifying the LOMS to specify job attributes" on page 160	Optional: To specify Infoprint Server job attributes
"Creating an SAP R/3 output device" on page 161	Required

Table 27 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 27. Customization tasks for Print Interface when SAP R/3 application server does not run on same z/OS system

Task	Condition
"Creating an SAP R/3 output device" on page 161	Required
"Customizing the Print Interface LPD" on page 142	Required

**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 that 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 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 by using your preferred editor and save your changes. For example:

```
oedit /etc/Printsrv/aopsapd.conf
```

Edit this file by 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 or *envvar* file, 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 that is defined in the **system** attribute. The Print Interface Callback daemon initially contacts this server instance to obtain configuration information about the LOMSs defined on that SAP R/3 system.

**client** = *clientname*

The name of the client that is 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 that is defined to SAP for use by the user ID.

**language** = *language*

The language that is used for messages from the SAP R/3 application server.

**Note:** The client name, user ID, and password must match the values that are specified when you defined an SAP user for the OMS Callback daemon. See “Defining an SAP user for the callback daemon” on page 156.

## 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 = xxxxxxx
  language = EN
;

system EX0
  initial-target = boefcs5_EX0_00
  client = 000
  userid = mike
  password = xxxxxxx
  language = EN
;

```

## Syntax rules

When you are editing the SAP Callback daemon configuration file, follow these syntax rules:

1. End the last attribute in each entry with a semicolon.
2. Use lowercase characters for the attribute names and uppercase or lowercase characters for the attribute values.
3. Code blank characters before or after the equal sign, if wanted.
4. If a value contains blank characters or special characters (such as {}-> =, #), enclose the value in single or double quotation marks.
5. Start comments with the # symbol.

6. Include blank lines, if wanted.

## Setting environment variables for the SAP OMS

The following environment variables affect the behavior of the SAP OMS. Set these environment variables in the **aopstart** EXEC or *envvar* file, and in the */etc/profile* file.

### 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 LOMs 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 z/OS

It creates two LOMs 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 LOMs, 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 LOMs. 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 LOM. 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. The values are 1 - 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** commands 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 that you select.
- The name of the LOMS defined to SAP R/3. Default names for Infoprint Server LOMs 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 with 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, complete 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, complete 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 with 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, complete 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, complete 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 that the print job completed, the print job might remain on the JES spool.
  - If the status indicates that the print job is waiting in the host spooler, the position in the queue does not reflect the order in which the job is printed. For example, jobs that are not submitted through Infoprint Server might print before the job.
- Display device information:
  - Print jobs that are submitted to the same printer with 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 is printed.
- Cancel a print job:
 

Users can cancel a print job only if Infoprint Server is not yet selected it for processing.

---

## Customizing the Print Interface subsystem

The Print Interface subsystem can process output data that is 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 by using job attributes and JCL parameters that are specified on the DD JCL statement, the associated OUTPUT JCL statement, and values that are 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.

Table 28 summarizes the customization tasks for the Print Interface subsystem. Required tasks are required by all installations that want to use the Print Interface subsystem. Optional tasks are required only if the listed condition applies to your installation.

*Table 28. Customization tasks for Print Interface subsystem*

Task	Condition
"Editing the aopd.conf file for the Print Interface subsystem" on page 164	Required

Table 28. Customization tasks for Print Interface subsystem (continued)

Task	Condition
"Increasing the maximum number of active transform daemons" on page 165	Optional

**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 a start command or a JCL startup 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 you restart Infoprint Server.

**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 = {xpd 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 startup procedures (if PSF uses the Printer Inventory).

**Rule:** Do not change this attribute while Infoprint Server (including NetSpool, IP PrintWay basic mode, and Infoprint Central) or the Printer Inventory for PSF is running. If you change this attribute, you must stop and restart:

- All Infoprint Server daemons
- NetSpool

- IP PrintWay basic mode
- IBM HTTP Server - Powered by Apache (when running Infoprint Central)
- PSF for z/OS (when running Printer Inventory for PSF)

**Default:** inventory = A0P1

**Related information:** For information about:

- All attributes in the aopd.conf file, see Appendix B, “Infoprint Server files,” on page 395.
- How to edit the aopd.conf file, see “Steps for editing the Infoprint Server configuration file (aopd.conf)” on page 45.

## 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 does 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 are not 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 409.



---

## 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 21 and “Transforms” on page 22.

Table 29 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 29. Customization tasks for Transform Interface*

Task	Condition
“Identifying software requirements for Transform Interface” on page 168	Required
“Editing the aopd.conf file for the Transform Manager” on page 168	Optional: To use IBM transforms that run on the z/OS system
“Creating the transform configuration file (aopxfd.conf)” on page 169	Optional: To use IBM transforms that run on the z/OS system
“Creating the remote transform configuration file (aoprxf.conf)” on page 175	Optional: To use Ricoh transforms that InfoPrint Transform Manager for Linux manages
“Setting environment variables for Transform Interface” on page 176	Optional: To change the default location of configuration files
“Customizing the TCP/IP firewall” on page 178	Optional: To use transforms that run on a remote system
“Setting up security for transform resources” on page 178	Optional: To use transform resources
“Creating a code page conversion table” on page 178	Optional: To use Ricoh transforms that InfoPrint Transform Manager for Linux manages with a custom EBCDIC code page for the z/OS locale

Table 30 lists other components of Infoprint Server that you might need to customize:

*Table 30. Other components to customize for Transform Interface*

Task	Condition
Chapter 3, “Customizing the Printer Inventory Manager,” on page 27	Required
Chapter 4, “Customizing Print Interface,” on page 125	Optional: To transform data that is submitted to Print Interface or to IP PrintWay basic mode
Chapter 7, “Customizing IP PrintWay,” on page 207	Optional: To transform data that is submitted to IP PrintWay basic mode or extended mode

**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 you restart Infoprint Server.

**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 395.
- How to edit the aopd.conf file, see “Steps for editing the Infoprint Server configuration file (aopd.conf)” on page 45.



---

## 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 transform data only 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 starts a transform automatically or the user starts 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:  
`pcl2afp.dll -c letter_300`
- The job submitter specifies the transform class on the z/OS UNIX transform command in the **-c** option. For example:  
`pcl2afp -c letter_300 -o myfile.afp myfile.pcl`
- The job submitter specifies the transform class in the **filter-options** job attribute on the **lp** command or 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 1 - 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 start 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 that is identified in the PATH environment variable in the **aopstart** EXEC or with the **aopdaemon**, 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 or *envvar* file. 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 123). 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:**

1. 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. 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.
2. 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.
3. 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 high number because active transform daemons use some system resources. However, starting transform daemons uses more CPU resources, so you need to 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 started.

```
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 174.

**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.

To create the transform configuration file:

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 authorized 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:

- a. 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 or *envvar* file, and the `/etc/profile` file. For information, see “Setting environment variables for Transform Interface” on page 176.
  - b. You can use the sample job that is 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 must 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 use 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 writable only by UID 0 and members of the AOPADMIN group:

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

6. Edit the file by using your preferred editor and save the changes:

```
oedit /etc/Printsrv/aopxfd.conf
```

Edit this file by 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 you run 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 168.

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 that is 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:

1. Start comments with the # symbol
2. End the last attribute in a transform entry with a semicolon.
3. Code blank characters before and after equal signs, if wanted.
4. Use lowercase characters for the attribute names. Use uppercase or lowercase characters for the attribute values.
5. Environment variable names are case-sensitive. Typically, environment variable names are all uppercase.
6. If a value contains blank characters or special characters (such as {}-> =, # &), enclose the value in single or double quotation marks.
7. To continue a line that contains a character string that is enclosed in quotation marks, end the line with a backslash and do not type any characters, even blank characters, after the backslash. For example:

```
environment = {AOP_FONTLIB -> "sys1.font300  \
               sys1.sfntlib inst.font300"}
```

## Steps for editing the transform configuration file

These steps show how to edit an existing `aopxfd.conf` configuration file. If you did not create the `aopxfd.conf` file yet, follow the steps in “Steps for creating the transform configuration file” on page 172.

**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 transform configuration file:

1. Edit the file by using your preferred editor and save the changes:

```
oedit /etc/Printsrv/aopxfd.conf
```

Edit this file by 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 that is 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:

1. Start comments with the `#` symbol.
2. End the last attribute in a transform entry with a semicolon.
3. Code blank characters before and after equal signs, if wanted.
4. Use lowercase characters for the attribute names. Use uppercase or lowercase characters for the attribute values.
5. Environment variable names are case-sensitive. Typically, environment variable names are all uppercase.
6. If a value contains blank characters or special characters (such as `{}`, `->`, `=`, `#` &), enclose the value in single or double quotation marks.
7. To continue a line that contains a character string that is enclosed in quotation marks, end the line with a backslash and do not type any characters, even blank characters, after the backslash. For example:

```
environment = {AOP_FONTLIB -> "sys1.font300  \
               sys1.sfntlib 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:**

1. `remote-transform-manager-ip-address = linux1.xyz.com`
2. `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.

To create the remote transform configuration file:

1. Switch to an effective UID of 0:

```
su
```

To use the z/OS UNIX **su** command, you must be authorized 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:**

- a. 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 `/etc/profile` and the **aopstart** EXEC or *envvar* file. For information, see “Setting environment variables for Transform Interface.”
  - b. You can use the sample job that is 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 must 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 use 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 writable only by UID 0 and members of the AOPADMIN group:  

```
chmod 664 /etc/Printsrv/aoprxf.conf
```
  6. Edit the file by using your preferred editor and save the changes.  

```
oedit /etc/Printsrv/aoprxf.conf
```

Edit this file by 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:

1. Use lowercase characters for the attribute names. Use uppercase or lowercase characters for the values.
2. Use blank characters before or after the equal sign, if wanted.
3. Start comments with the `#` symbol.
4. Include blank lines, if wanted.

---

## 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 or the **aopdemon** environment variables file: Transform Interface uses environment variables that are specified in this file. For information about how to edit this file, see “Editing the aopstart EXEC” on page 62 or “Creating an environment variables file (*envvar*)” on page 65.
- */etc/profile* file: z/OS UNIX transform commands, such as **ps2afp** and **remotexf**, use environment variables that are specified in this file.

#### 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:

1. 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
```

2. Change these lines in the *envvar* file:

```
AOPRXF_CONF=/etc/Printsrv/aoprxf.conf,
AOPXFD_CONF=/etc/Printsrv/aopxfd.conf
```

#### Tips:

- a. If you use the AOPDEMON JCL procedure, define the *envvar* file on the VARFILE statement and assign UID 0 to the task.
- b. After you edit the *envvar* file, stop and restart all Infoprint Server daemons. For example, enter these MVS commands to run the AOPSTOP2 and the AOPDEMON or AOPSTAR2 JCL procedures:

```
START AOPSTOP2
START AOPDEMON or START AOPSTAR2
```

3. 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 firewall

If you configured a TCP/IP firewall, make sure that the z/OS system is allowed to communicate by 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 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.d11** 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 are 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 are on the z/OS system. And, the SAP to AFP transform uses configuration files that are 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 are on the z/OS system. For example, InfoPrint Transform Manager for Linux transforms use a PostScript halftone dictionary, which can be on the z/OS system. If a resource is 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 14.

Table 31 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 31. Customization tasks for NetSpool*

Task	Condition
“Identifying software requirements for NetSpool” on page 180	Required
“Updating the BPXPRMxx member of SYS1.PARMLIB” on page 180	Required
“Editing the aopd.conf file for NetSpool” on page 181	Required
“Customizing the NetSpool message log” on page 182	Optional: To record messages in the NetSpool log
“Customizing the NetSpool startup procedure” on page 182	Required
“Working with JES spool limits” on page 188	Required
“Changing Language Environment runtime options” on page 189	Optional: To change default options
“Writing NetSpool exit and filter programs” on page 189	Optional: To customize how NetSpool processes SCS and 3270 data streams
“Setting environment variables for NetSpool” on page 202	Optional: To print line data that contains unprintable characters
“Editing the system configuration definition for NetSpool” on page 203	Optional: To specify dynamic attributes

Table 32 lists the other components of Infoprint Server that you might need to customize when you use NetSpool.

*Table 32. Other components to customize for NetSpool*

Task	Condition
Chapter 3, “Customizing the Printer Inventory Manager,” on page 27	Required
Chapter 5, “Customizing Transform Interface,” on page 167 <sup>1</sup>	Optional: To transform data from one format to another

Table 32. Other components to customize for NetSpool (continued)

Task	Condition
Chapter 7, “Customizing IP PrintWay,” on page 207	Optional: To print data to remote printers in your Internet Protocol network or send data to email destinations
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 that are 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 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:

### 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 that are used by both NetSpool and Print Interface. If you also run IP PrintWay, add an extra 50 (extended mode) or an extra 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 you restart Infoprint Server.

**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:

1. 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.
2. The job ID that Print Interface returns to the job submitter does not contain this prefix.

### Rules:

1. 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.

2. 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 you restart 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 45.
- All attributes in the `aopd.conf` file, see Appendix B, “Infoprint Server files,” on page 395.
- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 78.

---

## 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 you start NetSpool, you must create a startup procedure. The startup procedure identifies:

- Program name, region size, timeout 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

Table 33 summarizes the customization tasks that are 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.

*Table 33. Customization tasks for NetSpool startup procedure*

Task	Condition
"Determining the number of NetSpool tasks to start"	Required
"Using the Japanese version"	Optional: To receive messages in Japanese
"Using an extended MCS console" on page 184	Optional: To receive unsolicited messages at an extended console
"Creating a user ID to assign to the NetSpool startup procedure" on page 184	Required
"Defining the NetSpool startup procedure to RACF" on page 185	Required
"Editing the NetSpool startup procedure" on page 185	Optional: To change default settings

## 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 can run only 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 V2R2 Program Directory*.

To receive messages in Japanese, specify the LANGUAGE parameter in the EXEC statement by using the last 3 characters in the name of the message data set. For example, specify LANGUAGE=JPN. For more information about the LANGUAGE parameter, see “Editing the NetSpool startup procedure” on page 185.

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 46.

Be certain your console supports the language that 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 that are sent by NetSpool are already translated into Japanese.

## Using an extended MCS console

When you use 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 because of 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 included in SYS1.IBM.PROCLIB. However, during installation APIJPJCL might be 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 LNKLIST, 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 alternative 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 21 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.S.LOADLIB,DISP=SHR
//APIMSG DD DSN=&MSGFILE,DISP=SHR
//SYSPRINT DD SYSOUT=*
```

Figure 21. 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 21 specifies default values for symbolic parameters that are 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 starts. The class of each printer LU is specified in the printer definition in the Printer Inventory.

Valid class values are 1 - 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, 1 - 64, that does not match any classes that are

defined for printer LUs in the Printer Inventory. To start individual printers, use Infoprint Central or the NetSpool LUNAME ADD command that is 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 uses 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 reads the data set in the `TRACEOPT DD` statement to obtain the trace options. The tracing starts when NetSpool is started.

**STEPLIB DD DSN=dsname,DISP=SHR**

The name of a data set that contains NetSpool exit programs that are 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 `LNKLST`.

**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 and 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 182.

**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 (**aopippd**)
- 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 and **aopdemon** command define `_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` 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 22 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 22. 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(APIOPN). Place the CEEUOPT text deck first when you link-edit it with module APIOPN.
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

---

### Programming interface information

---

This section documents programming-interfaces intended to allow you to write programs to obtain the services of Infoprint Server.

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 is 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 131.
- 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 you are coding exit programs:

- You need to 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 that is passed to the exit. Code the references to fields in the control block by using the names that are 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 are 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 end.

**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, which is mapped by DSECT S2STG, shown in "Sample APIPP1X2 control block."

### Sample APIPP1X2 control block

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:
				Value	Class
		1... ....		1	Class 1
		.1.. ....		1	Class 2
		.... ...1		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
=====					
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'

## 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 it transmits 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 help calculate the line and column numbers you want, 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 that are 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 that are 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 “Sample APIPP1X2 control block” on page 191. 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 converts 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 that is 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 “Sample APIPP1X2 control block” on page 191.

**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 not add data from S2WRKARA to the output data set.
- 4      Close the VTAM session for this printer LU.

- 8 Do not call exit APIPPTD2 for any transparent data that is found in the data stream for this output data set. Do not add data from S2WRKARA to the output data set.
- 9 Do not call exit APIPPTD2 for any transparent data that is 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 allocates 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 *z/OS Infoprint Server User's Guide*.

### **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 an X'35' character followed by a 1-byte length field. IP PrintWay removes the X'35' and length field before it transmits 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 that are 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 that are 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 “Sample APIPP1X2 control block” on page 191. 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 converts the input data stream to a PCL data stream.

**S2PLUNAM**

LU name of the application that is 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 “Sample APIPP1X2 control block” on page 191.

**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 tracks the next output print position, that is, the column number and line number at which the next character prints. 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 help calculate the line and column numbers you want, 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 "Sample APIPP1X2 control block" on page 191. 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 converts the input data stream to a PCL data stream.

<b>S2COL</b>	The column number of the next print position.
<b>S2LM</b>	Left margin column number.
<b>S2RM</b>	Right margin column number.
<b>S2MPP</b>	Maximum presentation position (line length).
<b>S2LINE</b>	The line number of the next print position.
<b>S2TM</b>	Top margin line number (channel 1).
<b>S2C2T12</b>	Channel 2-12 line numbers.
<b>S2BM</b>	Bottom margin line number.
<b>S2MPL</b>	Maximum page length.
<b>S2PLUNAM</b>	LU name of the application that is generating the print request (PLU).
<b>S2LPTNAM</b>	Name of the printer LU that NetSpool is processing (SLU).
<b>S2PRTCLS</b>	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 "Sample APIPP1X2 control block" on page 191.
<b>S2TPDATA</b>	Transparent data that is 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 - 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 not modify the transparent data.
4	Close the VTAM session for this printer LU.
8	Do not call exit APIPPTD2 again for any additional transparent data that is found in this data set. Do not modify the transparent data.
9	Do not call exit APIPPTD2 again for any additional transparent data that is found in this data set. Modify the transparent data. S2WRKARA contains the modified data.

Also, set these fields in the APIPP1X2 control block:

<b>S2NCNUM</b>	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 *z/OS Infoprint Server User's Guide*.

**Note:** Do not change the TRN control or data in the S2TPDATA field.

## Examples

1. To include the transparent data that is passed to this exit, unmodified, in the output data stream, set code 2 or 8 in register 15.
2. 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.
3. To delete the transparent data that is passed to this exit from the output data stream:
  - Set the S2WRKLEN field to zero to indicate that no data is in S2WRKARA.
  - Set code 1 or 9 in register 15.

## Graphic Escape exit (APIUGEX)

The Graphic Escape exit (APIUGEX) for both SCS and 3270 data streams receives control when NetSpool finds a Graphic Escape (GE) character sequence in the input data stream. A GE character sequence begins with a 1-byte code, X'08', followed by the GE printable character. You can use this exit to replace the Graphic Escape character sequence with another character or sequence of characters.

If you do not modify the default Graphic Escape exit, NetSpool removes the GE code (X'08') and prints the next character in the font that is used to print the rest of the data set. For example, for GE character sequence X'08C1', letter A prints. A GE code typically means, however, that the printer prints the next character in an alternative font. Therefore, if any of your VTAM applications generate GE characters, you might want to write an exit that replaces GE characters with a different character in the same font or the same character in the printer's alternative font. The sample Graphic Escape exit illustrates how to replace GE characters with the same character in the alternative font.

NetSpool calls the same Graphic Escape 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 replace a GE character sequence, follow these guidelines:

- When NetSpool creates a PCL data stream, code the replacement characters in ASCII representation.
- When NetSpool creates a line data stream, code the replacement characters in EBCDIC representation.
- Do not include the X'08' GE code in the replacement character sequence.

## Input to APIUGEX

At input to the Graphic Escape exit (APIUGEX), 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 “Sample APIPP1X2 control block” on page 191. 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 converts the input data stream to a PCL data stream.

### S2PLUNAM

LU name of the application that is 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 “Sample APIPP1X2 control block” on page 191.

### S2TPDATA

First byte: The Graphic Escape (GE) character that is found in the input data stream. That is, the character after GE code X'08'. This character is usually in EBCDIC representation.

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'.

## 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 not 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

1. To keep the Graphic Escape character, set return code 2 in register 15.
2. 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, which is 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:

1. Make sure that the name of the file is correct. For the required names, see "Sample exits and macro."
2. 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.

3. Put the exit in an APF-authorized library such as SYS1.LINKLIB.

Identify the library that contains 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.

---

End programming interface information

---

---

## 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

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:

1. 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.

2. 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.
3. 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:**

1. If dynamic configuration is enabled, specify the equivalent **allow-all-characters-in-line-data** attribute in the system configuration definition.
2. 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 62
- Environment variables, see “Setting environment variables for the Printer Inventory Manager” on page 46

---

## 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 zeros, and printable characters. However, some applications might create line data that contains unprintable characters.

- |            |                                                                                                                                                                                                                                                              |
|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>yes</b> | Infoprint Server detects the data format as “line data” even if the input data contains unprintable characters.                                                                                                                                              |
| <b>no</b>  | 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:**

1. 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.



2. 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.
3. 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:**

1. 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.
2. If dynamic configuration is not enabled, specify the equivalent AOP\_ALLOW\_ALL\_CHARACTERS\_IN\_LINE\_DATA environment variable in the **aopstart** EXEC.
3. 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:**

1. 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.
2. The job ID that Print Interface returns to the job submitter does not contain this prefix.

**Rules:**

1. 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.
2. 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 you restart 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 78.
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 91.



---

## 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 TCP/IP-attached ASCII printers or print servers, VTAM controlled printers in an SNA network, or email destinations. For an overview of IP PrintWay and how this component fits into your system, see “IP PrintWay” on page 16.

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 more 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 210. 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 483.

Table 34 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 34. Customization tasks for IP PrintWay extended mode*

Task (IP PrintWay extended mode)	Condition
“Comparing IP PrintWay extended mode to basic mode” on page 210	Required
“Identifying software requirements for IP PrintWay” on page 213	Required
“Editing the aopd.conf file for IP PrintWay (extended mode)” on page 213	Required
“Increasing the space available in /var/Printsrv directory (extended mode)” on page 215	Required
“Updating the BPXPRMxx member of SYS1.PARMLIB (extended mode)” on page 215	Required
“Creating IP PrintWay job selection rules (extended mode)” on page 216	Required
“Authorizing IP PrintWay to select data sets from the JES spool (extended mode)” on page 222	Required
“Authorizing IP PrintWay to write SMF type 6 records (extended mode)” on page 223	Required
“Setting environment variables for IP PrintWay (extended mode)” on page 224	Optional: To customize IP PrintWay processing
“Editing the system configuration definition for IP PrintWay (extended mode)” on page 232	Optional: To specify dynamic attributes

Table 34. Customization tasks for IP PrintWay extended mode (continued)

Task (IP PrintWay extended mode)	Condition
"APF-authorizing SYS1.IMAGELIB (extended mode)" on page 240	Optional: To use FCBs
"Customizing from AFP transforms for IP PrintWay (extended mode)" on page 240	Optional: If you use the AFP to PCL or AFP to PostScript transform
"Defining printer host names in the DNS" on page 240	Required
"Customizing z/OS Communications Server" on page 241	Required
"Creating a VTAM APPL resource definition for IP PrintWay" on page 245	Optional: To print on VTAM controlled printers
"Writing IP PrintWay exit and filter programs" on page 246	Optional: To customize IP PrintWay processing
"Customizing PSF for sharing network printers" on page 337	Optional: To share printers with PSF
"Customizing AOPCMND for starting and stopping printers" on page 337	Optional: To start and stop printers from TSO instead of Infoprint Central.

Table 35 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 35. Customization tasks for IP PrintWay basic mode

Task (IP PrintWay basic mode)	Condition
"Comparing IP PrintWay extended mode to basic mode" on page 210	Required
"Identifying software requirements for IP PrintWay" on page 213	Required
"Defining printer host names in the DNS" on page 240	Required
"Customizing z/OS Communications Server" on page 241	Required
"Creating a VTAM APPL resource definition for IP PrintWay" on page 245	Optional: To print on VTAM controlled printers
"Writing IP PrintWay exit and filter programs" on page 246	Optional: To customize IP PrintWay processing
"Updating the BPXPRMxx member of SYS1.PARMLIB (basic mode)" on page 313	Optional: To start multiple IP PrintWay FSAs
"APF-authorizing SYS1.IMAGELIB (basic mode)" on page 314	Optional: To use FCBs
"Defining the IP PrintWay functional subsystem (basic mode)" on page 314	Required
"Creating FSS and FSA definitions in the Printer Inventory (basic mode)" on page 325	Optional: To print on VTAM controlled printers or to change default configuration options
"Creating the transmission-queue data set (basic mode)" on page 328	Required
"Creating the IP PrintWay message-log data set (basic mode)" on page 331	Required
"Customizing the IP PrintWay startup procedure (basic mode)" on page 332	Required
"Changing Language Environment runtime options (basic mode)" on page 337	Optional: To change default options

Table 35. Customization tasks for IP PrintWay basic mode (continued)

Task (IP PrintWay basic mode)	Condition
"Customizing PSF for sharing network printers" on page 337	Optional: To share printers with PSF

If you 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 36.

Table 36. Customization tasks for IP PrintWay protocols

Task	LPR	Direct sockets	IPP	VTAM	Email	See
Install Coax Printer Support	No	No	No	Yes	No	"Identifying software requirements for IP PrintWay" on page 213
Create a VTAM APPL definition	No	No	No	Yes	No	"Creating a VTAM APPL resource definition for IP PrintWay" on page 245
Specify the IP PrintWay APPL ID in an FSS definition (basic mode)	No	No	No	Yes	No	"Creating FSS definitions (basic mode)" on page 325
Specify the IP PrintWay APPL ID in the AOP_APPLID environment variable (extended mode)	No	No	No	Yes	No	"Setting environment variables for IP PrintWay (extended mode)" on page 224
Customize z/OS UNIX sendmail	No	No	No	No	Yes	"Customizing z/OS UNIX sendmail" on page 243
Specify environment variables in the IP PrintWay startup procedure (basic mode)	No	No	Yes	No	Yes	"Customizing the IP PrintWay startup procedure (basic mode)" on page 332
Specify environment variables (extended mode)	No	No	No	Yes	Yes	"Setting environment variables for IP PrintWay (extended mode)" on page 224

Table 37 lists other components of Infoprint Server that you might need to customize.

Table 37. Other components to customize for IP PrintWay

Task	Condition
Chapter 3, "Customizing the Printer Inventory Manager," on page 27	Required
Chapter 8, "Customizing Infoprint Central," on page 339	Optional but suggested (extended mode only)
Chapter 5, "Customizing Transform Interface," on page 167	Optional: To transform data from one format to another
Customizing the Print Interface LPD, see "Common customization for Print Interface" on page 126	Optional: To transform data from one format to another (basic mode only)

**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 that are required by IP PrintWay. To transform data, also

specify attributes that are 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 a start command or a JCL startup 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 selected to process, is waiting to retransmit, or 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 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 PjL 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 PjL 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 **tracert** command to pinpoint a network problem.
  - Redirect all print jobs on a printer's queue (except for the print job that is currently processing) and all future print jobs to an alternative printer. The alternative 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 selected them for processing.
  - Move print jobs to alternative printers, including print jobs that are currently being processed. The alternative 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 message in the beginning of emails.
- IP PrintWay extended mode writes messages in the Infoprint Server common message log.

### Limitations:

1. 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 can select print jobs in class P, while IP PrintWay on system B can 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.

2. 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.
3. IP PrintWay extended mode prints all data sets in an output group on the same printer, which is the printer that is 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 38 summarizes how IP PrintWay extended mode differs from IP PrintWay basic mode.

*Table 38. 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"> <li>• IP PrintWay extended mode does not support the Message exit.</li> <li>• IP PrintWay extended mode does not pass some values in the ANFUEXTP control block to the exits.</li> <li>• The exits cannot hold or delete output data sets.</li> </ul>
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"> <li>• To start IP PrintWay extended mode daemons, you must edit the Infoprint Server configuration file (aopd.conf).</li> <li>• 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.</li> <li>• You specify environment variables for IP PrintWay extended mode in the <b>aopstart</b> EXEC or <i>envvar</i> file instead of in the basic mode startup procedure.</li> <li>• You might need more space in the /var/Printsrv directory because IP PrintWay extended mode writes data to this directory instead of to hiperspace.</li> <li>• You must customize Infoprint Central so that operators can work with IP PrintWay extended mode print jobs and printers.</li> <li>• You do not need to define JES functional subsystems (FSSs and FSAs) for IP PrintWay extended mode.</li> </ul>

Table 38. How IP PrintWay extended mode differs from basic mode (continued)

Area	How IP PrintWay basic and extended modes differ
Diagnosis	<p>The major differences between IP PrintWay basic and extended modes are:</p> <ul style="list-style-type: none"> <li>• To trace IP PrintWay extended mode, you use the Infoprint Server trace facility. IP PrintWay basic mode uses GTF tracing.</li> <li>• IP PrintWay extended mode writes messages to the common message log. IP PrintWay basic mode writes messages to a separate data set.</li> </ul>
Job submission	<p>IP PrintWay extended mode prints all print jobs in the same JES output group to the same printer by using the same retry and retention values.</p>
Operations	<p>The major differences between IP PrintWay basic and extended modes are:</p> <ul style="list-style-type: none"> <li>• To start and stop IP PrintWay extended mode, you use the JCL startup procedure instead of the IP PrintWay basic mode startup procedure.</li> <li>• To start and stop individual printers, you can use: <ul style="list-style-type: none"> <li>– Infoprint Central instead of JES commands.</li> <li>– AOPCMND from TSO/E.</li> </ul> </li> <li>• 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.</li> <li>• To view messages from IP PrintWay extended mode, you can use either Infoprint Central or the <b>aoplogu</b> 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 <b>aoplogu</b> command to copy messages in a specified time range to a file.</li> </ul>

## Identifying software requirements for IP PrintWay

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 this attribute in the aopd.conf file:

**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 you restart Infoprint Server.

**Default:** start-daemons = {lpd}

**Example:** In this example, the Printer Inventory Manager daemon, the LPD, the IP PrintWay extended mode daemon, and the Infoprint Central daemon start: `start-daemons = {lpd outd ssid}`

You can also specify this attribute in the `aopd.conf` configuration 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 by 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 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:**

1. 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 by using the IP address instead of the printer's host name.
2. 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 you restart Infoprint Server.

**Default:** `resolve-printway-printers = yes`

**Example:** `resolve-printway-printers = no`

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:**

1. 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.
2. 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 you restart Infoprint Server.
3. 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 45.
- All attributes in the `aopd.conf` file, see Appendix B, “Infoprint Server files,” on page 395.
- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 78.

---

## Increasing the space available in `/var/Printsrv` directory (extended mode)

IP PrintWay extended mode writes data to the `/var/Printsrv` directory before it sends 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 34.

---

## Updating the BPXPRMxx member of SYS1.PARMLIB (extended mode)

When you use IP PrintWay extended mode, you 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:

**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 extra 50 to the current MAXPROCSYS value. If you also run Print Interface or NetSpool, add an extra 200 for processes that are 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. See “Setting environment variables for IP PrintWay (extended mode)” on page 224.

**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 more planning information and for information about how to create the file that is 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 that are shown in Table 39 on page 217.

Table 39. 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 <b>DEST</b> field of the printer definition. Wildcards are allowed <sup>1</sup> . <b>Note:</b> 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 <b>CLASS</b> field of the printer definition. You can specify 1 - 36 classes.	D, N
FORMS	The forms name of the print jobs. This value must match the FORMS parameter of the OUTPUT JCL statement, the <b>forms</b> job attribute, and the <b>FORMS</b> field of the printer definition. You can specify 1 - 8 forms names. Wildcards are allowed <sup>1</sup> .	STD, ACCT*
Creator	The z/OS user ID associated with the print jobs. For print jobs submitted with JCL, the <b>lp</b> command, or the AOPPRINT JCL procedure, specify the user ID of the user who submits the print requests. For other print jobs that Print Interface or NetSpool allocates on the JES spool, specify the user ID of the user who started the Infoprint Server daemons. Wildcards are allowed <sup>1</sup> .	USER01
Writer	The writer name for the print jobs. This value must match the WRITER parameter of the OUTPUT JCL statement and the <b>Writer</b> field of the printer definition. Wildcards are allowed <sup>1</sup> .	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"> <li>• <b>Include:</b> Select only print jobs that specify the DEST=IP: parameter.</li> <li>• <b>Exclude:</b> Select only print jobs that do not specify the DEST=IP: parameter.</li> <li>• <b>Ignore:</b> Select print jobs that specify the DEST=IP: parameter and jobs that do not specify the DEST=IP: parameter.</li> </ul>	Include



Table 39. IP PrintWay job selection rules (continued)

Job selection criteria	Description	Examples
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. <b>Note:</b> 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 32 K 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
1. You can specify an asterisk (*) and a question mark (?) in any position in the DEST, CLASS, FORMS, Creator, and WRITER fields as wildcards: <ul style="list-style-type: none"> <li>• An * represents zero or more characters.</li> <li>• A ? represents one character.</li> </ul>		

#### Tips:

1. If you specify multiple job selection criteria in a job selection rule, IP PrintWay selects print jobs that meet all the criteria that are specified in the rule. For example, if you select Include in the **DEST=IP** field and 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.
2. If you specify no job selection criteria in a job selection rule, IP PrintWay selects all print jobs from the JES spool.
3. 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 is 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 by 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.
4. 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 can select print jobs in class P, while IP PrintWay on system B can 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 can allocate print jobs in class P, while Print Interface and NetSpool running on system B can allocate print jobs in class Q.

5. 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 must be IP PrintWay print jobs.
6. 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 processes 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 processes 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 is not specified, IP PrintWay 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 if dynamic configuration is not enabled. 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 224.

7. 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 can 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.
8. To migrate from IP PrintWay basic mode:
  - a. 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:

- 1) 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.
- 2) 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.

- b. 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 must 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 is 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 must create separate job selection rules. For example, you can 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 can 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 are already selected from the JES spool.

### Examples:

1. This Infoprint Server ISPF panel shows a job selection rule that is 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 \_\_\_\_\_

2. This Infoprint Server ISPF panel shows a job selection rule that is 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 365.

## 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 that are already selected by 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 with 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:**

- a. 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;"
```
- b. 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:**

- a. 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 the following **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
```
  - b. 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:**

- a. `chmod +x /u/userid/do.dayrules`
  - b. `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 AOPOPER group access to profiles in the JESSPOOL class because users who start Infoprint Server must be members of the AOPOPER group or have a UID of 0. (AOPOPER is the default group name for Infoprint Server operators. However, your installation can use a different name for this group.) If someone with a user ID of 0 who is not a member of the AOPOPER group can start Infoprint Server (for example, by using the **aopstart** command), you must also give this user ID access to the profiles in the JESSPOOL class.

**Examples:**

1. To define a broad profile to protect all print jobs on node N1, enter this RACF command:  
`RDEFINE JESSPOOL (N1.*.*.D*.) UACC(NONE)`
2. 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:**

1. If you also define more specific profiles in the JESSPOOL class, give the AOOPER group access to those profiles.
2. If you authorize 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, if **Operating mode=z/OS 2.1**, enter these MVS commands to run the AOPSTOP and AOPSTART JCL procedures:  
`START AOPSTOP,OPTIONS='-d outd'`  
`START AOPSTART`
3. 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, by 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 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, if **Operating mode=z/OS 2.1**, 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 these environment variables in the **aopstart** EXEC or *envvar* file:

### **`_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:** The name that 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 that is specified by the STEPLIBLIST statement in the BPXPRMxx PARMLIB member. See “Updating the BPXPRMxx member of SYS1.PARMLIB (extended mode)” on page 215.

**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 that is created by batch applications print with extra blank characters at the end of lines (see `AOP_BLANK_TRUNCATION_CLASSES`).
- You want to suppress errors because of 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**

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:**

1. 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.
2. 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.
3. 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:**

1. If dynamic configuration is enabled, specify the equivalent **allow-all-characters-in-line-data** attribute in the system configuration definition.
2. 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 that is 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:**

1. If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **applid** attribute in the system configuration definition.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
3. 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.

**Value:** 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:**

1. Specify this environment variable only if line or text data prints with extra blank characters at the end of lines.
2. This environment variable can slow performance and cause IP PrintWay to use more system resources.
3. Before JES gives 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 can 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:**

1. If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **blank-truncation-classes** attribute in the system configuration definition.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
3. 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 that are 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:

1. Set this environment variable to YES if your IP PrintWay job-selection rules cause IP PrintWay to select many 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.
2. 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.
3. 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:

1. 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.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
3. 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:**

1. 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.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
3. 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 that are 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:**

1. Specify the suffixes that users might specify in file names. For example, you might specify the .doc suffix for Microsoft Word documents.
2. 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:**

1. Begin each suffix with a period.
2. 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.
3. If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **mail-preserve-suffixes** attribute in the system configuration definition.
4. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
5. 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).

**Values:**

- YES** IP PrintWay sends each document to the email address or addresses that are specified for the first document in the print job.
- NO** IP PrintWay sends each document to the email address or addresses that are specified for the document.

**Tips:**

1. 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 that are specified for the first document in the print job.
2. IP PrintWay basic mode ignores this environment variable. IP PrintWay basic mode always sends each document to the email address or addresses that are specified for the document unless the **Concatenate job** option is selected in the **Dataset grouping** field in the printer definition.

**Rules:**

1. 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.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart EXEC**.
3. 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. You can use Infoprint Central to move the held output groups to different IP PrintWay printers before you release them.
- NO** IP PrintWay extended mode does not select new held output groups from the JES spool. You can use JES or SDSF commands to release individual data sets in an output group for printing.

**Rules:**

1. 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.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart EXEC**.
3. 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 that is 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:**

1. 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.
2. 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 that is returned by DLL filters.

#### **Rules:**

1. 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.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
3. 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**

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:**

1. Start the path name with a slash (/).
2. If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **mailer-path-name** attribute in the system configuration definition.

3. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
4. 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:

1. If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **mailer-options** attribute in the system configuration definition.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
3. 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:

1. 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 or *envvar* file. Also, add the library to the file specified by the STEPLIBLIST statement in the BPXPRMxx PARMLIB member.
2. 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.
3. 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 62
- The LANG, LIBPATH, NLSPATH, and PATH variables, see “Setting environment variables for the Printer Inventory Manager” on page 46
- 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 zeros, 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:

1. 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.
2. 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.
3. 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:

1. 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.
2. If dynamic configuration is not enabled, specify the equivalent AOP\_ALLOW\_ALL\_CHARACTERS\_IN\_LINE\_DATA environment variable in the **aopstart** EXEC.
3. 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 that is 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:**

1. 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.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOP\_APPLID environment variable in the **aopstart** EXEC.
3. 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.

**Value:** 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:**

1. Specify this attribute only if line or text data prints with extra blank characters at the end of lines.
2. This attribute can slow performance and cause IP PrintWay to use more system resources.
3. Before JES gives 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 can 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:**

1. 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.
2. 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.
3. 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 that are 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:**

1. Set **ignore-dcf-routing-errors = yes** if your IP PrintWay job-selection rules cause IP PrintWay to select many 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.
2. 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.
3. 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:**

1. 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.
2. 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.
3. This attribute does not apply to IP PrintWay basic mode.

**Default:** ignore-dcf-routing-errors = no

**Example:** ignore-dcf-routing-errors = yes

**ipsmode = z201 | z202**

Indicates the product function level that you want Infoprint Server to operate with.

**z201** Infoprint Server operates with z/OS 2.1 functions (default).

**z202** Infoprint Server operates with z/OS 2.2 functions.

**Rule:** You can specify this attribute only in the system configuration definition.

**Default:** ipsmode=z201

**Example:** ipsmode=z202

**mail-do-not-add-suffixes = yes | no**

Indicates whether IP PrintWay extended mode appends a suffix to file names that are 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:**

1. 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.
2. 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.
3. 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 that are 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:**

1. Specify the suffixes that users might specify in file names. For example, you might specify the .doc suffix for Microsoft Word documents.
2. If the **mail-do-not-add-suffixes = yes**, IP PrintWay ignores the **mail-preserve-suffixes** attribute.

**Rules:**

1. Begin each suffix with a period.
2. To specify more than one suffix, separate the suffixes with a blank and enclose the suffixes in single or double quotation marks.
3. 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.
4. 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.
5. 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 that are specified for the first document in the print job.

- yes** IP PrintWay sends each document to the email address or addresses that are specified for the first document in the print job.
- no** IP PrintWay sends each document to the email address or addresses that are specified for that document.

**Rules:**

1. 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.
2. 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.
3. 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 that are 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:**

1. To specify more than one option, separate the options with a blank and enclose the options in single or double quotation marks.
2. 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.
3. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOPMAILER\_OPTIONS environment variable in the **aopstart** EXEC.
4. 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:**

1. Start the path name with a slash (`/`).
2. 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.
3. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOPMAILER environment variable in the **aopstart** EXEC.
4. 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.

- |            |                                                                                                                                                                                                      |
|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>yes</b> | IP PrintWay extended mode selects new held output groups from the JES spool. You can use Infoprint Central to move the held output groups to different IP PrintWay printers before you release them. |
| <b>no</b>  | IP PrintWay extended mode does not select new held output groups from the JES spool. You can use JES or SDSF commands to release individual data sets in an output group for printing.               |

**Rules:**

1. 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.
2. 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.
3. 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:**

1. 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.
2. 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 you restart Infoprint Server.
3. 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 that is 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:**

1. 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.
2. 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 that is returned by DLL filters.

**Rules:**

1. 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.
2. 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.
3. This attribute does not apply to IP PrintWay basic mode.

**Default:** suppress-post-unix-filter-formatting = no

**Example:** suppress-post-unix-filter-formatting = yes

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 78
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 91

---

## 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 and in the STEPLIBLIST sanction list in the BPXPRMxx PARMLIB member. For information, see:

- “Setting environment variables for IP PrintWay (extended mode)” on page 224
- “Updating the BPXPRMxx member of SYS1.PARMLIB (extended mode)” on page 215

---

## Customizing from AFP transforms for IP PrintWay (extended mode)

If the administrator selects one of the Printer Job Control (PJL) 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 PJL 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 PJL 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 PJL 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 PJL JOB commands, see *IBM Print Transforms from AFP for Infoprint Server for z/OS*.
- PJL 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 213.

## 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 V2R2.0 Communications Server: IP Configuration Guide* and *z/OS V2R2.0 Communications Server: IP Configuration Reference*.

Table 40 describes tasks that are related to TCP/IP customization. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies.

Table 40. Customization tasks for TCP/IP

Task	Condition
“Customizing the TCPIP.DATA data set”	Optional: To specify the TCP/IP name or high-level qualifier
“Customizing the PROFILE.TCPIP data set” on page 243	Optional: To change TCP/IP configuration
“Customizing z/OS UNIX sendmail” on page 243	Optional: To send output to email destinations

### 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 V2R2.0 Communications Server: IP Configuration Reference*.

This TCPIP.DATA statement primarily affects IP PrintWay extended mode:

### 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:

### 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 it searches 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 identify that data set in the IP PrintWay basic mode startup procedure with a SYSTCPD DD statement or use the job name 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 that 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 127.

## 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 that is provided with z/OS Communications Server that provides enhanced SMTP support.

If you did not already customize 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 V2R2.0 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 (<http://www.sendmail.org>).

**Guidelines:** Follow these guidelines when you customize sendmail:

1. Sendmail highly depends on the domain name server (DNS). Therefore, be sure to set up this component correctly.
2. Customize the **Timeout** option in the sendmail configuration file to suit your installation. The Timeout option specifies how long sendmail waits before it sends a warning that email cannot be delivered to a remote system and how long sendmail waits before it bounces email. Bounced email is undeliverable email.
3. 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 looks in its queue to process pending mail.

**Example:**

```
su
/usr/sbin/sendmail -bd -qlm &
exit
```

The **su** command switches to an effective UID of 0. To use **su**, you must be authorized 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.

4. In the sendmail aliases file, you can create alias names to represent a list of real email addresses. When you create an alias, you must 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).
5. 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.
6. 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.
7. 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 configured sendmail correctly, run the **sendmail** command directly from the z/OS UNIX command line. The **sendmail** command is described in *z/OS V2R2.0 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 is delivered to you and you 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 is delivered to the remote address and you 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 by using your favorite editor and save your changes:

```
su
oedit /etc/mail/aliases
```

To use the **su** command, you must be authorized 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 is delivered to you and you 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 232
- How to specify environment variables for IP PrintWay extended mode, see “Setting environment variables for IP PrintWay (extended mode)” on page 224
- How to specify environment variables for IP PrintWay basic mode, see “Customizing the IP PrintWay startup procedure (basic mode)” on page 332

---

## 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 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 23 and Figure 24 show sample APPL statements that are created under a major node named PRINTWAY.

```
PRINTWAY VBUILD TYPE=APPL Application major node
*
PWAPPL APPL AUTH=ACQ APPL for IP PrintWay extended
```

Figure 23. 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 24. 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 V2R2.0 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 325.

#### **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**

---

### **Programming interface information**

---

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 256.

**Note:** The email addresses specified in an email header override email addresses that are 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 can add a separator page. For more information, see “Begin Data Set exit” on page 267.

**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 279.

**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 can add a separator page. For more information, see “End Data Set exit” on page 287.

**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 is transmitted. For more information, see “SMF exit (ANFUXSMF)” on page 296.

**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 302.

**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 304.

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 it writes 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 131.

## 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 LNKLIST:
  - 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 333.
  - IP PrintWay extended mode: Specify the step library in the STEPLIB environment variable. For information, see “Setting environment variables for IP PrintWay (extended mode)” on page 224.

Also, specify the step library in the file that is 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 are run in supervisor state with a protection key of 1.
  - IP PrintWay extended mode: Exits are 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 252.
- 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 251.
- 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 zeros.
- 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 zeros 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 94.

## 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 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 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 that are originally specified by the job submitter in JCL parameters. Instead, these fields contain values that are specified in the printer definition or, if no value is specified, a default value that is assigned by JES. Table 41 summarizes the values in some of the ANFUEXTP fields when the **resubmit for filtering** option is selected.

*Table 41. 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.
XTIPIADR, XTPPRTQU, XTPIPORT	IP address, print queue name, and port number that is specified in the printer definition.
XTPRETNS, XTPRETNF	Retention times that are specified in the printer definition.
XTPRTRYL, XTPRTRYT	Retry values that are specified in the printer definition.
XTPUSRID	User ID of the person who submitted the original batch job.

To obtain the original JCL values that are 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 that are 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 that is 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 ANFUEXTTP control block to determine whether the **resubmit for filtering** option is selected in the printer definition.

## Programming considerations

Consider these points when you are coding an exit:

- You need to be an experienced programmer who is aware of overall system implications.
- Test the exit before you install it on a production system. A programming exception in an exit causes an IP PrintWay extended mode or IP PrintWay basic mode FSAabend. An ESTAI is in effect while the exit is running.
- The exit must 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 that is 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 are run in problem state with a protection key of 8.
- As it processes an output group, IP PrintWay passes the same Exit Parameter Area (ANFUEXTTP) 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 must contain zeros. 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 do 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 must also contain extra 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 that is 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
  - JSPJGRP
  - JSPJGRPD
  - JSPJRMNO
  - JSPJPNAM
  - JSPJDSNM
  - JSPJSOCL
  - JSPJPRI
  - 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:

1. 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).
2. 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.
3. 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 that are specified for the data set from the SWBTU. For example, you can obtain distribution parameters that are specified on the OUTPUT statement and print them on a separator page that is 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 250 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 takes 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. "Sample format of ANFUEXTP" shows the format of ANFUEXTP. Macro ANFUEXTP provides a DSECT in assembler language for this control block.

### Sample format of ANFUEXTP:

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	PRTOPTS 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	XTPPRLLEN Length of original record
168	(A8)	Character	16	XTPOPTNM Name of components
184	(B8)	Unsigned	2	XTPLOLEN Length of LPR options
186	(BA)	Unsigned	2	XTPPQLEN Length of value in XTPPRTQU
188	(BC)	Unsigned	2	XTPIALEN Length of value in XTIPIADR
190	(BE)	Unsigned	2	XTPMSGLN Length of message
192	(C0)	Character	256	XTPPRTQU Name of print queue or VTAM logmode
448	(1C0)	Character	256	XTIPIADR IP address/host name, URL, LU name, or email addresses
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	XTIPIPORT 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
=====				
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

**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 are merged. Parameters on the DD and OUTPUT override the values that are 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 that is 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 it processes 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 312.

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 that is specified by job submitters on the OUTPUT JCL statement. To do this, the Routing exit can inspect the values that

are specified in the RETAINS and RETAINF JCL parameters and change them if they exceed the maximum time that is defined for your installation.

- Restrict the use of certain printers to specific users. To do this, the Routing exit can 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 can 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 250.

## Input to the Routing exit

At entry to the Routing exit, the registers contain:

### Register 1

Pointer to the address of the ANFUEXTTP 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 ANFUEXTTP control block are valid when the Routing exit is called. The format of ANFUEXTTP is shown in “Sample format of ANFUEXTTP” on page 254.

### XTPSAVE

A register save area for use by the exit.

### XTPWORK1

Address of a work area, or zeros. For information about how to use work areas, see “Programming considerations” on page 251.

### XTPWORK2

Address of a work area, or zeros. For information about how to use work areas, see “Programming considerations” on page 251.

### 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 ANFUEXTTP 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 that is embedded in the print data or specified in the printer definition. If none is specified, this field 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 meanings 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 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 that is specified in the PRTQUEUE JCL parameter or in a printer definition; or the VTAM logmode that is 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:

- Host name or IP address of the target printer or print server that is specified in the DEST=IP: JCL parameter, the **printer-ip-address** job attribute, or in the printer definition
- URL of the target printer that is specified in the printer definition
- LU name of the target printer that is specified in the printer definition
- Email addresses of the primary recipients that are 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.

**Tip:** IP PrintWay does not pass the email addresses of the copy or blind copy recipients to the exit.

#### XTPRETNS

Amount of time IP PrintWay retains 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 set a default value of zeros. For the format of this field, see the note on page 256.

#### **XTPRETNF**

Amount of time IP PrintWay retains the data set on the JES spool after all retransmissions failed. This time was specified in the RETAINF JCL parameter or in the printer definition. IP PrintWay might set a default value of zeros. For the format of this field, see the note on page 256.

#### **XTPRTRYL**

Number of times IP PrintWay tries to retransmit the data set. This number was specified in the RETRYL JCL parameter or in the printer definition. IP PrintWay might set a default value of zero.

#### **XTPRTRYT**

Amount of time IP PrintWay waits between retransmissions. This time was specified in the RETRYT JCL parameter or in the printer definition. IP PrintWay might set a default value of zero. For the format of this field, see the note on page 256.

#### **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 that is 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 that is 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 that is 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 that is 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 253.

## 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 252.

- 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

Flag containing miscellaneous information 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. Table 42 lists the values that identify the protocols and field contents.

*Table 42. XTP\_ADDR\_TYPE values*

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 uses to obtain printing options:

##### **Value    Meaning**

- 0**      IP PrintWay selected the printer definition by matching the DEST, CLASS, and FORMS fields that are specified in JCL parameters with the values in the printer definition.
- 1**      IP PrintWay selected the printer definition by using the printer definition name that is 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 that is 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 that is 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 that is 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 zeros. For information about how to use work areas, see “Programming considerations” on page 251.

### **XTPWORK2**

Address of a work area, or zeros. For information about how to use work areas, see “Programming considerations” on page 251.

### **XTPDSFLG**

Flags indicating the action that you want IP PrintWay to take upon return. IP PrintWay sets these flags to B'0' before it calls the exit. To transmit the data set, do not set any of these flags.

**Note:** Set only one of these flags 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 that is created by this exit. IP PrintWay sets these flags to B'0' before it calls the exit. Put the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN.

Flag meanings 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 a message that is 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 that you make to fields that contain routing parameters, such as fields XTPOPTNM, XTPPRTQU, XTPIPADR, XTPRETNS, XTPRETNE, 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 that is 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 that is 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 that is 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 1 - 255.

#### **XTP\_ROUTING\_KEY\_TYPE**

Indicates how you want IP PrintWay to select the new printer definition:

<b>Value</b>	<b>Meaning</b>
<b>0</b>	IP PrintWay selects the printer definition by matching the values in fields XTPCLASS, XTPDEST, and XTPFORMS with the values in the printer definition.
<b>1</b>	IP PrintWay selects the printer definition by 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 that you make to fields XTPKEY, XTPCLASS, XTPFORMS, and XTPDEST.)

#### **XTOPTNM**

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:**

1. The Routing exit cannot change the email addresses of the copy or blind copy recipients.
2. If an email address is specified in a header in the document, the email address in the document overrides the email address that is 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**

Amount of time IP PrintWay retains the data set on the JES spool after a successful transmission. For the format of this field, see the note on page 256.

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**

Amount of time IP PrintWay retains the data set on the JES spool after all retransmissions failed. For the format of this field, see the note on page 256.

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 tries to retransmit the data set. Specify a number 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**

Amount of time IP PrintWay waits between retransmissions. For the format of this field, see the note on page 256.

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 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, 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 312.

## 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 that is contained in the scheduler work block text unit (SWBTU). See “Scheduler work block text unit (SWBTU)” on page 253.
  - 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 310.
  - 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 an extra 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 that is 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 IIISi 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:**

1. 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.
2. 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 250.

## **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 ANFUEXTTP 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 “Sample format of ANFUEXTP” on page 254.

#### **XTPSAVE**

A register save area for use by the exit.

#### **XTPWORK1**

Address of a work area, or zeros. For information about how to use work areas, see “Programming considerations” on page 251.

#### **XTPWORK2**

Address of a work area, or zeros. For information about how to use work areas, see “Programming considerations” on page 251.

#### **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.

<b>Value</b>	<b>Meaning</b>
--------------	----------------

<b>0</b>	IP PrintWay basic mode called the exit.
----------	-----------------------------------------

<b>1</b>	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 that is 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 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

Flag containing data set information 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 that is specified in the PRTQUEUE JCL parameter, in a printer definition, or by the Routing exit; or the VTAM logmode that is 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 that is 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 that is specified in the printer definition or by the Routing exit
- LU name of the printer that is specified in the printer definition or by the Routing exit
- Email addresses of the primary recipients that are 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

Amount of time IP PrintWay retains 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 set a default value of zeros. For the format of this field, see the note on page 256.

#### XTPRETNF

Amount of time IP PrintWay retains the data set on the JES spool after all retransmissions failed. This time was specified in the RETAINF JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might set a default value of zeros. For the format of this field, see the note on page 256.

### XTPRTRYL

Number of times IP PrintWay tries to retransmit the data set. This number was specified in the RETRYL JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might set a default value of zero.

### XTPRTRYT

Amount of time IP PrintWay waits between retransmissions. This time was specified in the RETRYT JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might set a default value of zero. For the format of this field, see the note on page 256.

### 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 that is 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 that is 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 that is specified in a 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 that is 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 253.

**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 252.

- 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**

Flags containing miscellaneous information 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. Table 43 lists the values that identify the protocols and field contents.

Table 43. XTP\_ADDR\_TYPE values

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 uses to obtain printing options:

Value	Meaning
-------	---------

- |   |                                                                                                                                                                             |
|---|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 | IP PrintWay selected the printer definition by matching the DEST, CLASS, and FORMS fields that are specified in JCL statements with the values in the printer definition.   |
| 1 | IP PrintWay selected the printer definition by using the printer definition name that is 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 250.

Value	Meaning
-------	---------

- |   |                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|---|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 | The <b>resubmit for filtering</b> option is not selected in the printer definition.                                                                                                                                                                                                                                                                                                                                                                       |
| 2 | The <b>resubmit for filtering</b> 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 already transformed the input data into the format that is required by the printer. For example, if the <b>afpxpcl</b> 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 that is 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 that is 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 that is 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 that are 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:**

1. XTP\_DOC\_HEADER\_LENGTH contains the length of the document header.
2. XTP\_TRANS\_DOC\_HEADER indicates whether IP PrintWay translates the document header to the printer code page before it sends 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:**

1. XTP\_DOC\_TRAILER\_LENGTH contains the length of the document trailer.
2. XTP\_TRANS\_DOC\_TRAILER indicates whether IP PrintWay translates the document trailer to the printer code page before it sends 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 zeros. 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 zeros.

### **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 zeros. For information about how to use work areas, see "Programming considerations" on page 251.

#### **XTPWORK2**

Address of a work area, or zeros. For information about how to use work areas, see "Programming considerations" on page 251.

#### **XTPRCFLG**

Flags to request that IP PrintWay add a record and specify characteristics of the new record.

#### **Tips:**

1. IP PrintWay sets all flags to B'0' before it calls the exit.
2. 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.
3. 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 meanings 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 that is created by this exit. IP PrintWay sets these flags to B'0' before it calls the exit. Put the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN.

Flag meanings 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 a message that is 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 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 it sends 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 it sends 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 translates 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 that is 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 that is 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 that 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, 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 must 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:

- a. Use similar instructions to specify a document trailer.
- b. 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 312.

## 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:

- Is 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:

1. 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.
2. 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 250.

## 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 "Sample format of ANFUEXTP" on page 254.

**XTPSAVE**

A register save area for use by the exit.

**XTPWORK1**

Address of a work area, or zeros. For information about how to use work areas, see "Programming considerations" on page 251.

**XTPWORK2**

Address of a work area, or zeros. For information about how to use work areas, see "Programming considerations" on page 251.

**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 that is 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 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

Flag containing record information 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.

#### 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 that is specified in the PRTQUEUE JCL parameter, in a printer definition, or by the Routing exit; or the VTAM logmode that is 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 that is 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 that is specified in the printer definition or by the Routing exit
- LU name of the printer that is specified in the printer definition or by the Routing exit
- Email addresses of the primary recipients that are 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

Amount of time IP PrintWay retains 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 set a default value of zeros. For the format of this field, see the note on page 256.

#### XTPRETNF

Amount of time IP PrintWay retains the data set on the JES spool after all retransmissions failed. This time was specified in the RETAINF JCL

parameter, in the printer definition, or by the Routing exit. IP PrintWay might set a default value of zeros. For the format of this field, see the note on page 256.

#### **XTPRTRYL**

Number of times IP PrintWay tries to retransmit the data set. This number was specified in the RETRYL JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might set a default value of zero.

#### **XTPRTRYT**

Amount of time IP PrintWay waits between retransmissions. This time was specified in the RETRYT JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might set a default value of zero. For the format of this field, see the note on page 256.

#### **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 that is 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 that is 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 that is 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 253.

#### **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 252.

- 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 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. Table 44 lists the values that identify the protocols and field contents.

Table 44. XTP\_ADDR\_TYPE values

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 uses to obtain printing options:

#### Value    Meaning

- |   |                                                                                                                                                                             |
|---|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 | IP PrintWay selected the printer definition by matching the DEST, CLASS, and FORMS fields that are specified in JCL statements with the values in the printer definition.   |
| 1 | IP PrintWay selected the printer definition by using the printer definition name that is 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 that is 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 that is 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 that is 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 that are 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 zeros. For information about how to use work areas, see “Programming considerations” on page 251.

**XTPWORK2**

Address of a work area, or zeros. For information about how to use work areas, see “Programming considerations” on page 251.

**XTPRCFLG**

Flags to request that IP PrintWay add the original record or a new record and to specify characteristics of the record.

**Tips:**

1. IP PrintWay sets all flags to B'0' before it calls the exit.
2. 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.
3. 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 meanings 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:**

1. To avoid a blank line when the exit adds a printer instruction, code the printer instruction in hexadecimal and do not set this flag.
2. This flag does not apply to the original record.

**XTPRCCC**

The new record contains a carriage control character in column 1. This flag does not 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 not apply to the original record.

**XTPMSGFL**

Flags indicating the destination of a message that is created by this exit. Put the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN.

Flag meanings 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 a message that is 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 record after the original record in the data set, set these fields in ANFUEXTP:
  - Set flag XTPRCORG=1 to request that IP PrintWay transmits the original record.
  - Set flag XTPRLAST=0 to indicate that IP PrintWay calls 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 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 that 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 changed, put the new record length in XTPPRLLEN.
  - 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, 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 312.

## 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 that is contained in the scheduler work block text unit (SWBTU). See “Scheduler work block text unit (SWBTU)” on page 253.
  - 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 310.
  - 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 a 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:**

1. 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.
2. 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 250.
3. To specify a document trailer that contains printer commands, use the Begin Data Set exit.
4. 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 “Sample format of ANFUExTP” on page 254.

**XTPSAVE**

A register save area for use by the exit.

**XTPWORK1**

Address of a work area, or zeros. For information about how to use work areas, see “Programming considerations” on page 251.

**XTPWORK2**

Address of a work area, or zeros. For information about how to use work areas, see “Programming considerations” on page 251.

**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 that is 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 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

Flag containing data set information 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 that is specified in the PRTQUEUE JCL parameter, in a printer definition, or by the Routing exit; or the VTAM logmode that is 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 that is 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 that is specified in the printer definition or by the Routing exit
- LU name of the printer that is specified in the printer definition or by the Routing exit
- Email addresses of the primary recipients that are 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**

Amount of time IP PrintWay retains 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 set a default value of zeros. For the format of this field, see the note on page 256.

#### **XTPRETNF**

Amount of time IP PrintWay retains the data set on the JES spool after all retransmissions failed. This time was specified in the RETAINF JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might set a default value of zeros. For the format of this field, see the note on page 256.

#### **XTPRTRYL**

Number of times IP PrintWay tries to retransmit the data set. This time was specified in the RETRYL JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might set a default value of zero.

#### **XTPRTRYT**

Amount of time IP PrintWay waits between retransmissions. This time was specified in the RETRYT JCL parameter, in the printer definition, or by the Routing exit. IP PrintWay might set a default value of zero. For the format of this field, see the note on page 256.

#### **XTPKEY**

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.

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 that is 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 that is 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 that is 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 that is 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 253.

## 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 252.

- 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 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 250.

### Value    Meaning

- |   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|---|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 | The <b>resubmit for filtering</b> option is not selected in the printer definition.                                                                                                                                                                                                                                                                                                                                                                                             |
| 2 | The <b>resubmit for filtering</b> 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 already transformed the input data into the format that is required by the printer. For example, if the <b>afpxpcl</b> 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 that is 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 that is 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 that is 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.

### **XTOPTNS**

Transmission options. These options were specified in the printer definition or in the components that are 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:**

1. XTP\_DOC\_HEADER\_LENGTH contains the length of the document header.
2. 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:**

1. XTP\_DOC\_TRAILER\_LENGTH contains the length of the document trailer.

2. 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 zeros. For information about how to use work areas, see "Programming considerations" on page 251.

**XTPWORK2**

Address of a work area, or zeros. For information about how to use work areas, see "Programming considerations" on page 251.

**XTPRCFLG**

Flags to request that IP PrintWay add a record and to specify characteristics of the new record.

**Tips:**

1. IP PrintWay sets all flags to B'0' before it calls the exit.
2. 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.
3. 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 meanings 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 that is created by this exit. IP PrintWay sets these flags to B'0' before it calls the exit. Put the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN.

Flag meanings 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 a message that is 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 that you are adding.
2. To transmit the data set with no changes, 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 312.

## SMF exit (ANFUXSMF)

The SMF exit lets you:

- Replace the SMF type 6 records that are written by IP PrintWay.
- Suppress the SMF type 6 records that are 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 312.

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 250. For an example of how to suppress the first SMF record, see “Examples -- SMF exit” on page 301.

**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 ANFUEXTTP 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 “Sample format of ANFUEXTP” on page 254.

**XTPSAVE**

A register save area for use by the exit.

**XTPWORK1**

Address of a work area, or zeros. For information about how to use work areas, see “Programming considerations” on page 251.

**XTPWORK2**

Address of a work area, or zeros. For information about how to use work areas, see “Programming considerations” on page 251.

**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**

- |          |                                            |
|----------|--------------------------------------------|
| <b>0</b> | IP PrintWay basic mode called the exit.    |
| <b>1</b> | 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 that is 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 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.

### XTPPRLN

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 that is 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 that is 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 that is 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 253.

### 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 252.

- 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 uses to obtain printing options:

##### Value    Meaning

- |          |                                                                                                                                                                             |
|----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>0</b> | IP PrintWay selected the printer definition by matching the DEST, CLASS, and FORMS fields that are specified on JCL statements with the values in the printer definition.   |
| <b>1</b> | IP PrintWay selected the printer definition by using the printer definition name that is 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 250.

##### Value    Meaning

- |          |                                                                                                                                                                                                                                                                                                                                                                                                   |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>0</b> | The <b>resubmit for filtering</b> option is not selected in the printer definition.                                                                                                                                                                                                                                                                                                               |
| <b>1</b> | The <b>resubmit for filtering</b> 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.                                                                                                                                                            |
| <b>2</b> | The <b>resubmit for filtering</b> option is selected and either this is the second data set that 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 that is 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 that is 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 that is 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 zeros. For information about how to use work areas, see "Programming considerations" on page 251.

#### **XTPWORK2**

Address of a work area, or zeros. For information about how to use work areas, see "Programming considerations" on page 251.

#### **XTPRCFLG**

Flag to request that IP PrintWay write or suppress the SMF record. IP PrintWay sets this flag to B'0' before it calls 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 that is created by this exit. IP PrintWay sets these flags to B'0' before it calls the exit. Put the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN.

Flag meanings 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 a message that is 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

1. 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 changed, put the new length in field XTPPRLN.
2. 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.
3. To suppress the SMF record, return control to IP PrintWay.
4. 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.
5. 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 312.

## 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 that are 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 312.

**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 “Sample format of ANFUEXTP” on page 254.

#### 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 zeros 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 that is 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 that is 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 that is 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 writes the message. IP PrintWay sets this flag to B'0' before it calls 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 has a message ID, followed by the message text.

### **Examples -- Message exit**

1. 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.
2. To suppress the message, return control to IP PrintWay.
3. 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 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 312.

## 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 312.

### 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 “Sample format of ANFUEXTP” on page 254.

#### XTPSAVE

A register save area for use by this exit.

#### XTPWORK1

Address of a work area, or zeros. For information about how to use work areas, see “Programming considerations” on page 251.

#### XTPWORK2

Address of a work area, or zeros. For information about how to use work areas, see “Programming considerations” on page 251.

#### XTPDSNAM

Fully qualified name of the data set being processed, padded to the right with blanks.

#### 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 that is 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 in a JCL parameter:

- If the flag is set to 1, the indicated parameter was specified in 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, IP PrintWay obtained the value from the PrintWay default printer definition. If no PrintWay default printer definition exists, IP PrintWay set a default value.

Flag meanings 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**

Flags containing data set information 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 that is specified in the PRTQUEUE JCL parameter, in the printer definition, or by the Routing exit.
- Name of the VTAM logmode that is 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**

Flag containing miscellaneous information 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 that is specified in the DEST=IP: JCL parameter, in the printer definition, or by the Routing exit
- URL of the target printer that is specified in the printer definition or by the Routing exit
- LU name of the target printer that is specified in the printer definition or by the Routing exit
- Email addresses of the primary recipients that are 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**

Amount of time IP PrintWay retains 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 set a default value of zeros. For the format of this field, see the note on page 256. XTPRETNS might not be available to the Response Notification exit if the Response code is X'08'.

#### **XTPRETNF**

Amount of time IP PrintWay retains the data set on the JES spool after all retransmissions failed. This time was specified in the RETAINF JCL parameter or in the printer definition. IP PrintWay might set a default

value of zeros. For the format of this field, see the note on page 256. XTPRETNF might not be available to the Response Notification exit if the Response code is X'08'.

#### **XTPRTRYL**

The number of times IP PrintWay tries to retransmit the data set. This time was specified in the RETRYL JCL parameter or in the printer definition. IP PrintWay might 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 amount of time IP PrintWay waits between retransmissions. This time was specified in the RETRYT JCL parameter or in the printer definition. IP PrintWay might set a default value of zeros. For the format of this field, see the note on page 256. 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 that is 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 that is 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 that is 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 that is 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 253.

#### 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 252.

#### 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 250.

Value	Meaning
-------	---------

- |   |                                                                                                                                                                                                                                    |
|---|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 | The <b>resubmit for filtering</b> option is not selected in the printer definition.                                                                                                                                                |
| 1 | The <b>resubmit for filtering</b> 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 <b>resubmit for filtering</b> option is selected in the printer definition and either this is the second data set that 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 that is 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 that is 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 that is 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 that are 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 312.

### **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 it calls the exit. Put the address of the message in field XTPMSGP and the length of the message in field XTPMSGLN.

Flag meanings 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 a message that is created by this exit. IP PrintWay adds message ID ANFM155I to the beginning of this message.

## **Examples -- Response Notification exit**

1. To write a message to the IP PrintWay message log when IP PrintWay 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 is printed to the log.
2. To write a message to the operator's console when IP PrintWay 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 is 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-aligned. 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 by 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 you

change 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 the Block Letter program (ANFUBLK)**

When you call 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 "Sample format of ANFUEXTP" on page 254.

### **XTPBIFLG**

Flags set to B'1' indicating the type of block letters you want:

#### **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 align 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, 1 - 8 characters, padded to the right with blanks.

## **Output from Block Letter program (ANFUBLK)**

ANFUBLK sets register 15 - 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 that contains 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 that contains 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 used only with the LPR printing protocol.

- ANFUXED1 adds a separator page that contains 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 that is 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. For information, see “Setting environment variables for IP PrintWay (extended mode)” on page 224.

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 333.

## 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.

End programming interface information

## 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:

#### MAXPROCUSER

The maximum number of processes that a single z/OS UNIX user ID can have concurrently active.

**Guideline:** Use at least 70 for Infoprint Server processes, and then add the number of FSSs and the total number of FSAs. For example, if you have 10 IP PrintWay FSSs that each have three FSAs, use:

$70+10+30=110$

#### 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 extra 200 for processes that are used by both NetSpool and Print Interface.

**Example:** If you create 10 IP PrintWay FSSs that each have three FSAs, and also use Print Interface or NetSpool, specify these statements in the BPXPRMxx member:

```
MAXPROCUSER(110) /*70(Infoprint Server)+40(IP PrintWay)*/  
MAXPROCSYS(1100) /*700(IP PrintWay)+200(Print Interface/NetSpool)+200(initial)*/
```

#### Tips:

1. To see how many processes a user ID is using, do these:

- a. Enter **D OMVS,A=ALL**.
  - b. Find the PID value for the user ID, and then enter **D OMVS,L,PID=nnnnnnnnnn**, where the *nnnnnnnnnn* is the PID value.
2. 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*.
3. You can use BPMPRMxx to set limits for all user IDs on the system. However, to allow one user ID to have a higher limit than the MAXPROCUSER setting, you can use the RACF ADDUSER or ALTUSER command to specify a PROCUSERMAX value for that user ID. For example:  

```
ALTUSER userid OMVS(PROCUSERMAX(500))
```
4. The LIMMSG statement in the BPXPRMxx member 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 by using the SETOMVS command. For a description of LIMMSG, see the SETOMVS command in *z/OS MVS System Commands*.

---

## 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 LNKLIST statement.
- If the LNKAUTH parameter that is put in the IEASYSxx member of SYS1.PARMLIB does not specify LNKLIST, also add the data sets to the APF statement in the PROGxx member.

As an alternative to updating the PROGxx member, you can update the LNKLISTxx and IEAAPFxx members of SYS1.PARMLIB. However, IBM 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 you code 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:**

1. JES lets you create up to 2000 FSSs.
2. For improved throughput and more efficient use of system resources, define several FSSs, with several FSAs per FSS.
3. 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 329.
4. To avoid system abends that can occur because of out-of-storage conditions:
  - Define more 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 workload. For information, see “Determining work-selection criteria (basic mode)” on page 316.
  - 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 acquired from the JES spool. Acquired data sets include data sets that IP PrintWay is currently processing and data sets that IP PrintWay retained on the JES spool. JES uses about 4 K 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 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 can 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:**

1. You can define up to 64 FSAs per FSS. However, for optimal performance do not define more than 35 FSAs per FSS.
2. 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 retransmissions to a printer are attempted before printing to another printer.
3. 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.
4. To make sure that critical printers are always available for printing, 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 can 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 shares the transmission-queue data set with no more than 9 other FSSs. For information, see “Sharing the transmission-queue data set (basic mode)” on page 329.

**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 313.

## 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 PRT\*\*\*\*\* statement or the JES3 DEVICE statement. JES2 and JES3 let you specify various 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 that are 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 that the administrator defined with CLASS P and DEST MYPRT. For more information on using DEST, CLASS, and FORMS to select a printer definition, see *z/OS Infoprint Server Operation and Administration*.

Some considerations that are 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 that is specified as an OUTPUT JCL parameter when it determines whether an output data set meets the work-selection criteria that are 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 that is specified in a JCL statement to select any printer definitions in the Printer Inventory. This is because JES3 passes the form name that is 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 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 (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 that is 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 the NIUCS parameter on the JES2 PRINTDEF statement or the UCS parameter on the JES2 PRTnnnnnn statement for the IP PrintWay FSA. If you do not specify a default font, JES2 uses GF10 as the default font.
  - Specify a default page definition in the NIFCB parameter on the JES2 PRINTDEF statement or the FCB parameter on the JES2 PRTnnnnnn statement for the IP PrintWay FSA.
- JES3:
  - Specify a default font in the CHARS parameter on the JES3 OUTSERV statement or 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 the CARRIAGE parameter on the JES3 OUTSERV statement or the CARRIAGE parameter on the JES3 DEVICE statement for the IP PrintWay FSA. If you do not 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 that is 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 PRTnnnnnn statement. When UCS=0, JES2 ignores any font that is 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 PRTnnnnnn 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 PRTnnnnnn 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 PRTnnnnnn 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 317.

### Sample JES2 initialization statements

Figure 25 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 25. 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 that are 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 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, which is specified as a 1 - 8 character name. In the example, the FSS name is PRINTWAY. This FSS name must match the FSS name in the PRT~~nnnnn~~ 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 332. This parameter is required.

#### **AUTOSTOP=YES | NO**

Specifies whether the FSS address space is 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 - 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=HASPFSM.

### **PRT~~nnnnn~~ statement**

A PRT~~nnnnn~~ statement defines the FSA to JES. This section describes the PRT~~nnnnn~~ 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 1 - 32767. JES2 lets you define more than one FSA with one PRT~~nnnnn~~ statement. For more information about how to define a range of FSAs with one PRT~~nnnnn~~ 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 PRT~~nnnnn~~ in the FSA definition.

PRT~~nnnnn~~ statement parameters include:

#### **CLASS | QUEUE=(classes)**

Specifies the output classes that are 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 that are specified here.

#### **FCB=fcb/pagedefinition**

Specifies the 1 - 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 that is specified in a printer definition or in the transform configuration file for output that is 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 317.

**FORMS={*form\_names*}**

Specifies the 1 - 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.

**PRESELECT=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 PRESELECT=NO to make sure that data sets print in the correct order.

**ROUTECD={*destinations*}**

Specifies the destinations that are 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 automatically starts 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 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 - 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 does 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 that is specified in the printer definition or transform configuration file for jobs that are 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 317.

**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 316.

**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 26 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 26. 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 that are shown in Figure 26 on page 321, 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 - 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 that concern 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 332. This parameter is required.

**SYSTEM**=*system\_name*

Specifies the JES3 processor on which the FSS runs. This parameter is optional. If you omit it, JES determines the default from the DEVICE statement.

**TERM**=**YES** | **NO**

YES specifies that the FSS terminates if the JES3 global address space is ended 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 - 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 that is 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 starts and stops this FSA dynamically.

**YES** Specifies that JES3 starts 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 starts and stops 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 that is 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 - 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 does not supply a default font name to IP PrintWay when the WS=U parameter is specified on the DEVICE statement.

**FCB**

Indicates that JES3 does 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 supplies 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 that is specified in the printer definition or, if none is specified, in the transform configuration file for jobs that are 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 317.

**WC=(classes)**

Specifies the output classes that are 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 that are 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 that is 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 316.
- F** Requests that JES3 pass the forms name that is specified on a JCL statement to the FSA.
- U** Requests that JES3 pass the font name that is specified on a JCL statement to the FSA.

**Notes:**

1. If you want IP PrintWay to use the form name that is specified in the FORMS JCL parameter to select printer definitions in the Printer Inventory, specify WS=F as one of the work-selection criteria. If you do not specify WS=F, JES3 does not pass the form name that is 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 that is specified in JCL to the IP PrintWay FSA.
3. If you want IP PrintWay or any data stream transforms to use the page definition that is specified in the PAGEDEF (or FCB) JCL parameter, specify WS=C as one of the work-selection criteria. If you do not specify WS=C, JES3 does not pass the page definition name that is specified in JCL to the IP PrintWay FSA.
4. If you want IP PrintWay or any data stream transforms to use the font that is 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 that is 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 that are specified in this definition when the IP PrintWay FSS starts.
- FSA definition. IP PrintWay uses values that are 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 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 45 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 45. IP PrintWay FSS definition

ISPF field name	Description	Default
<b>Applid</b>	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 that is defined to VTAM. <b>Note:</b> Specify this field only after you installed Infoprint Coaxial Printer Support for z/OS. Otherwise, IP PrintWay ends abnormally.	No default. This field is required to use the IP PrintWay VTAM protocol.
<b>Description</b>	A description of the FSS definition, for information purposes only.	No default.

Table 45. IP PrintWay FSS definition (continued)

ISPF field name	Description	Default
<b>Document code page</b>	<p>The default EBCDIC code page that is 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 that is 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 the document code page to the printer code page and 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.
<b>FSS name</b>	<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"> <li>• In JES2, the name of the JES FSS initialization statement</li> <li>• In JES3, the name of the JES FSSDEF initialization statement</li> </ul> <p>If the IP PrintWay startup procedure contains a JOB JCL statement, specify either:</p> <ol style="list-style-type: none"> <li>1. The name of the EXEC JCL statement (step name) in the IP PrintWay startup procedure (default name is IEFPROC)</li> <li>2. If no step name is specified, the name of the PROC JCL statement in the IP PrintWay startup procedure (default is ANFWPROC)</li> </ol>	No default. This field is required.
<b>Hiperspace blocks</b>	<p>The maximum number of 4 KB blocks that each FSA can use in hiperspace. The number that 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 GB.</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 <b>Maximum document size</b> or <b>Delete form feed</b> option is selected in the printer definition. IP PrintWay does not write data sets to hiperspace when it transmits data to VTAM controlled printers.</p>	8,000 (32 MB).

Table 45. IP PrintWay FSS definition (continued)

ISPF field name	Description	Default
<b>National language</b>	The language IP PrintWay uses for messages. You can select either English or Japanese.	English messages.
<b>Old style translation</b>	<p>An indicator of how IP PrintWay converts 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 either a document or a printer code page, a customized TCP/IP translate table, or a DBCS translate table.</p> <ul style="list-style-type: none"> <li>• If this field is selected, IP PrintWay uses the standard TCP/IP translate table to convert data from EBCDIC to ASCII.</li> <li>• 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 that is specified in the <b>Document code page</b> field of the IP PrintWay FSS definition in the Printer Inventory. The ASCII code page is IBM-850.</li> </ul> <p>This field is ignored when IP PrintWay prints on VTAM controlled printers or to email destinations.<sup>1</sup></p>	See the “Description” column.
<b>TCP/IP job name</b>	The name of the TCP/IP startup procedure.	The default TCP/IP name is used.
<b>Trace mode</b>	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 that is specified in an FSA definition overrides this trace mode.	No tracing is done.
<b>Trace prompt</b>	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.
<b>Trace table size</b>	The number of 4 KB pages of storage (above the 16 MB line) to allocate for each internal FSA trace table.	32 (128 KB).
<p>1. When you are 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.</p>		

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 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 defined to JES. In each IP PrintWay FSA definition, you can specify the type of tracing to start for the FSA.

Creating FSA definitions is optional. If an FSA definition does not exist, IP PrintWay uses the tracing value that is 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 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 transmission-queue data set. Multiple IP PrintWay FSSs, on the same or different z/OS systems, can also share the 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 tries the connection again. 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 retransmissions 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 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:

1. Do not share the transmission-queue data set among more than 10 - 15 FSSs. If you have a large workload, many FSAs per FSS, or critical printers that require high performance, do not share the transmission-queue data set among more than 10 FSSs.
2. If FSAs within multiple FSSs can transmit output data sets to the same printers, these FSSs can share the 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.

3. 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.
4. If you allocate more than one transmission-queue data set, put each data set on a separate volume. The volume is not heavily used.
5. 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 314.

## 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:

1. 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.
2. 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 that are 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.

3. If you do not periodically use the VSAM REPRO command to reclaim space, IP PrintWay might abnormally end (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 27. 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 zeros. `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 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 302.

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 timeout value
- Name of the Printer Inventory
- Location of the IP PrintWay transmission-queue and message-log data sets
- Environment variables

Table 46 summarizes the customization tasks that are 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.

*Table 46. Customization tasks for IP PrintWay startup procedure (basic mode)*

Task	Condition
“Creating a user ID to assign to the IP PrintWay startup procedure (basic mode)” on page 333	Required
“Defining the IP PrintWay startup procedure to RACF (basic mode)” on page 333	Required
“Editing the ANFWPROC startup procedure (basic mode)” on page 333	Optional: To change default settings

## 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 because of 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 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 included in SYS1.IBM.PROCLIB. However, during installation ANFWPROC might be 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 LNKLIST, 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 you restart 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 28 on page 335 shows the ANFWPROC procedure. JES initialization statements for an FSS that uses this startup procedure are shown in Figure 25 on page 318 (JES2) and in Figure 26 on page 321 (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 28. IP PrintWay startup procedure — SYS1.IBM.PROCLIB(ANFWPROC)

The JCL statements that 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:

1. 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 4 M.
2. Do not specify 0 M. If you specify 0 M 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 0 M and available storage becomes exhausted, an abend other than system F02 occurs. JES is not affected and you do not need to restart your z/OS system.

**TIME=NOLIMIT**

TIME=NOLIMIT prevents TIMEOUT abends.

**PARM='INV=inventory'**

The name that is 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=&TCPLQ..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 241.

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
```

**//ANFMMSG 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 413.

#### Rules:

1. If you specify the environment variables in an MVS data set, it must have these DCB attributes: RECFM=VB and LRECL=255.
2. 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 29 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 29. 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*.

---

## Customizing AOPCMND for starting and stopping printers

You can use the TSO/E command, AOPCMND, to start and stop IP PrintWay extended mode printers from TSO instead of from the Infoprint Central component that runs under the IBM HTTP Server - Powered by Apache.

To use the AOPCMND command, you must first do these customization steps:

1. Authorize the TSO/E command processor by editing the IKJTSOxx member of SYS1.PARMLIB and adding AOPCMND to the AUTHCMD list.
2. Enable dynamic configuration.

**Rule:** If you move AOPCMND from SYS1.LINKLIB to another library, make sure that it is in an APF-authorized library.

For information about using the AOPCMND command, see *z/OS Infoprint Server Operation and Administration*.

For information about customizing TSO/E, see *z/OS TSO/E 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 that are 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 must 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 with 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 alternative 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 are 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 23.

Table 47 on page 340 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 47. Customization tasks for Infoprint Central

Task	Condition
"Identifying software requirements for Infoprint Central"	Required
"Editing the aopd.conf file for Infoprint Central" on page 341	Required
"Customizing the IBM HTTP Server - Powered by Apache" on page 343	Required
"Setting up security for Infoprint Central" on page 352	Required
"Editing the system configuration definition for Infoprint Central" on page 374	Optional: To specify dynamic attributes
"Customizing web browsers" on page 376	Required
"Logging on to Infoprint Central from the web" on page 377	Required
"Customizing JES3 console standards (CONSTD statement)" on page 378	Optional: To change JES3 console editing characters
"Re-creating the search database" on page 378	Optional
Chapter 10, "Customizing PSF for z/OS to use the Printer Inventory," on page 385	Optional: To work with PSF printers from Infoprint Central

Table 48 lists other components of Infoprint Server you might need to customize:

Table 48. Other components to customize for Infoprint Central

Task	Condition
Chapter 3, "Customizing the Printer Inventory Manager," on page 27	Required
Chapter 7, "Customizing IP PrintWay," on page 207	Optional: To work with IP PrintWay printers from Infoprint Central

**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 IBM HTTP Server - Powered by Apache 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

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, V7.1 (5655-W43)
  - IBM 64-bit SDK for z/OS, Java Technology Edition, V7.1 (5655-W43)

For information, see Java products for z/OS (<http://www-03.ibm.com/systems/z/os/zos/tools/java/>).

- Microsoft Internet Explorer 9.0 or later
- Mozilla Firefox 24 Extended Support Release (ESR) or later

## 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 when the ISPF **Operating mode** field is set to z/OS 2.1, 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 you restart 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}

You can also specify these 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:

1. 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.
2. Do not use these reserved names: HC, INSTREAM, INTERNAL, OPERLOG, SYSIOSRS, SYSLOG, UNKNOWN.
3. 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*.

4. 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.
5. 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 you restart Infoprint Server.

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

**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 are removed from the JES spool.

You can specify a value 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 are removed from the JES spool.

**Guidelines:**

1. 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.
2. 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 34.
3. 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.
4. 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 122.
5. If you use **aoplogu** to view messages, the date and time format that is displayed in messages is controlled by the TZ and either the LC\_TIME or LC\_ALL environment variables. See “Setting environment variables for the Printer Inventory Manager” on page 46.

If you use Infoprint Central to view messages, the date and time format for the messages is controlled by the operating system that runs your web browser.

**Rules:**

1. 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.
2. 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 you restart 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 395.
- How to edit the Infoprint Server configuration file, see “Steps for editing the Infoprint Server configuration file (`aopd.conf`)” on page 45.
- How to enable dynamic configuration, see “Enabling dynamic configuration” on page 78.

---

## Customizing the IBM HTTP Server - Powered by Apache

In z/OS V2R1 and earlier versions, the Infoprint Central component of Infoprint Server was designed to work only with the IBM HTTP Server powered by Domino®. In z/OS V2R2, IBM removed support for the Domino HTTP Server and only the IBM HTTP Server - Powered by Apache, a base element of z/OS V2R2, is supported. You must now use the Apache 31-bit IBM HTTP Server with Infoprint Central.

To display Infoprint Central web pages, you must customize and start the IBM HTTP Server - Powered by Apache.

### Guidelines:

1. Start a separate IBM HTTP Server - Powered by Apache 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.
2. Start an IBM HTTP Server - Powered by Apache 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.
3. Customize the IBM HTTP Server - Powered by Apache 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 install the IBM HTTP Server - Powered by Apache, see *z/OS V2R2.0 HTTP Server - Powered by Apache User's Guide*. For more information about the HTTP Server - Powered by Apache, see WebSphere Application Server Library (<http://www.ibm.com/software/webservers/appserv/was/library/>).



Table 49 summarizes the customization tasks for using the HTTP Server - Powered by Apache with Infoprint Central. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies to your installation.

*Table 49. Customization tasks for using the HTTP Server - Powered by Apache with Infoprint Central*

Task	Condition
"Configuring the IBM HTTP Server - Powered by Apache"	Required
"Editing the configuration file for the IBM HTTP Server - Powered by Apache (conf/httpd.conf)" on page 345	Required
"Customizing the environment variables file for the IBM HTTP Server - Powered by Apache (bin/envvars)" on page 346	Required
"Creating a started task to run the IBM HTTP Server - Powered by Apache" on page 351	Required

## Configuring the IBM HTTP Server - Powered by Apache

The IBM HTTP Server - Powered by Apache files are installed with z/OS V2R2; however, to display Infoprint Central web pages, you must configure and start the HTTP server. The following steps describe how to set up a default configuration for the HTTP server that is used exclusively by Infoprint Central:

1. Create an installation directory for the server configuration files. For example:
 

```
cd install_directory
mkdir -p etc/websrv1
```
2. Change the directory to the IBM HTTP Server - Powered by Apache product directory:
 

```
cd /usr/lpp/ihsa_zos/.31bit
```
3. Run the installer program, bin/install\_ihs, to install the HTTP server product files in your installation directory, do the initial configuration, and create symbolic links from your home directory to the product directory. Use these parameters to run the installer program:
  - The installation directory for the configuration files.
  - The non-SSL port for the web server. Select a port other than the default port of 80; for example, 8081 or 8082. You might also want to change the port on the **Listen** directive.

For example:

```
./bin/install_ihs install_directory/etc/websrv1 8081
```

4. Confirm that you successfully installed an operating version of the IBM HTTP Server - Powered by Apache in your installation directory:
  - a. Change to the server's installation directory:
 

```
cd install_directory/etc/websrv1
```
  - b. Run these commands to verify that the IBM HTTP Server - Powered by Apache was successfully installed:
 

```
bin/apachectl -v
bin/apachectl configtest
```

You see output similar to this example when an installation is successful:

```
# bin/apachectl -v
Server version: IBM_HTTP_Server/8.5.5.2 (Unix)
Server built:   Jul 11 2014 18:07:04
```

```
# bin/apachectl configtest
Syntax OK
```

5. Use this command to start the IBM HTTP Server - Powered by Apache:  
bin/apachectl start
6. Verify that the IBM HTTP Server - Powered by Apache is running successfully by loading the default Infoprint Central web page with the host name or IP address of your z/OS system and the non-SSL port number you specified in step 3 on page 344. For example:  
http://bldserv.bldev.infoprint.com:8081/

If you have problems opening the web page, see *z/OS V2R2.0 HTTP Server - Powered by Apache User's Guide*.

7. Run this command to stop the IBM HTTP Server - Powered by Apache before you edit the HTTP server configuration file and environment variables file:  
bin/apachectl stop

## Editing the configuration file for the IBM HTTP Server - Powered by Apache (conf/httpd.conf)

The configuration file for the IBM HTTP Server - Powered by Apache, `conf/httpd.conf`, contains directives that customize the HTTP server. You must add directives to the configuration file so that the HTTP server can display Infoprint Central web pages.

To edit the configuration file, add the following information to the end of the `conf/httpd.conf` file.

**Tip:** Be sure to uncomment the last two lines if you ordered the Japanese feature.

```
# Make the following changes to the httpd.conf file
# for Infoprint Central.
# Normally, these can be added to the end of the file.

# Modify the HTTP response header
# -----
# Modification of the HTTP request and response headers is
# provided by the mod_headers module. This module is used
# to set the HTTP response header option to prevent clickjacking.
LoadModule headers_module modules/mod_headers.so

# SAF (RACF) authorization
# -----
# SAF authentication is provided by the mod_authnz_saf
# module. The mod_authz_default and mod_auth_basic modules
# provide basic authentication and authorization support
# which is needed in mod_authnz_saf configurations.
LoadModule authnz_saf_module modules/mod_authnz_saf.so

# Infoprint Central module
# -----
LoadModule IpC_module /usr/lpp/Printsrv/lib/aopipc.so

<Location /Infoprint/>
    AuthName "Infoprint Central"
    AuthType Basic
    AuthBasicProvider saf
    # Infoprint Central requires an authorized user
```

```

| Require valid-user
| SAFRunAs %%CLIENT%%
|
| # Use the following directive to use a specific SAF APPL class resource.
| # During user signon authentication, this class is checked for user permission.
| # The < > are added for demarcation and should not be present if you uncomment the directive.
| # SAFAPPLID <name of APPLID>
| # Example:
| # SAFAPPLID INFOPRINT
|
| # Handler for Infoprint Central requests
| SetHandler IpC
|
| # No translation by default for location /Infoprint/
| CharsetSourceEnc ISO8859-1
| CharsetDefault ISO8859-1
|
| # Prevent clickjacking
| Header append X-Frame-Options "SAMEORIGIN"
|
| </Location>
|
| <Location /Infoprint/help/EN_US/>
| # Infoprint Central help is in EBCDIC
| CharsetSourceEnc IBM-1047
| CharsetDefault ISO8859-1
| </Location>
|
| # Map URIs to directories
| Alias /Infoprint/Scripts/ /usr/lpp/Printsrv/InfoprintCentral/Scripts/
| Alias /Infoprint/css/ /usr/lpp/Printsrv/InfoprintCentral/css/
| Alias /Infoprint/Images/ /usr/lpp/Printsrv/InfoprintCentral/Images/
| Alias /Infoprint/help/En_US/ /usr/lpp/Printsrv/InfoprintCentral/help/En_US/
| Alias /Infoprint/En_US/ /usr/lpp/Printsrv/InfoprintCentral/
| # Japanese feature: uncomment these lines:
| #AliasMatch ^/Infoprint/help/Ja_JP/(.*)\.htm /usr/lpp/Printsrv/InfoprintCentral/help/Ja_JP/$1.htm.binary
| #Alias /Infoprint/Ja_JP/ /usr/lpp/Printsrv/InfoprintCentral/

```

## Customizing the environment variables file for the IBM HTTP Server - Powered by Apache (bin/envvars)

When you install the IBM HTTP Server - Powered by Apache, the environment variables file, bin/envvars, is created. To use the IBM HTTP Server - Powered by Apache with Infoprint Central, you must customize the bin/envvars file by adding environment variables at the end of the file. For example, you might add these variables:

```

| export AOPCENTRAL=/usr/lpp/Printsrv/InfoprintCentral
| export AOPCONF=/etc/Printsrv/aopd.conf # Location of aopd.conf
| export CLASSPATH=/usr/lpp/Printsrv/classes/penguin.jar:\
| /usr/lpp/Printsrv/classes/ipa.jar:/usr/lpp/Printsrv/classes/snmp.jar:\
| /usr/lpp/Printsrv/classes/modelplugin.jar
| export JAVA_HOME=/usr/lpp/java/J7.1
| export LANG=C
| export LC_ALL=En_US.IBM-1047
| export LIBPATH=$LIBPATH:/usr/lpp/Printsrv/lib:/usr/lpp/ixm/IBM/xm14c-5_7/lib:\
| /usr/lpp/ixm/IBM/xslt4c-1_11/lib
| export NLS_PATH=/usr/lpp/Printsrv/%L/%N:/usr/lib/nls/msg/%L/%N:\
| /usr/lpp/internet/%L/%N:/usr/lpp/ldap/lib/nls/msg/%L/%N

```

Follow these guidelines when you customize the bin/envvars file:

1. If you use the AOPCONF, LC\_ALL, and LC\_CTYPE environment variables in the HTTP server environment variables file, they must match the variables that are specified in the environment for the start command or the JCL startup procedure.

2. Start each environment variable on a separate line.
3. Use a backslash, \, to continue an environment variable on another line.
4. Do not include any blanks after the last character on a line.
5. After you update the HTTP server environment variables file, do these steps:
  - a. Restart the HTTP server with this command so that your changes take effect:  

```
bin/apachectl restart
```
  - b. Verify that you can open the main Infoprint Central web page. For example, enter:  

```
http://bldserv.bldev.infoprint.com:8081/Infoprint/En_US/IPS.html
```
  - c. If you have problems opening the web page, see the logs/error\_log file. You can use the LogLevel directive in conf/httpd.conf to adjust the number of the messages that are recorded in the Apache error logs. You can also set GUI\_DEBUG=1, AOPTRACEON=1, or both in the bin/envvars file to get more information.

Table 50 summarizes the variables that you might need to set in the 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 50. Environment variables in bin/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	/usr/lpp/java/J7.1.
LANG	Optional	C
LC_ALL	Optional	C
LIBPATH	Required	None
NLSPATH	Required	None

### **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:**

1. 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.
2. You need to 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.
3. 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.
4. 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:**

1. For information about how to assign an SNMP write community name to printers, see the documentation for the printer or contact the printer manufacturer.
2. 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.
3. To do printer actions, Infoprint Central users must have CONTROL access to the RACF profile in the PRINTSRV class that protects the printer.
4. 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, *envvar* file, or JCL startup procedure. This environment variable is optional.

Specify this environment variable only in the IBM HTTP Server - Powered by Apache environment variables file (bin/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 IBM HTTP Server - Powered by Apache environment variables file (bin/envvars) and in the **aopstart** EXEC or *envvar* file must match.

**Default:** /etc/Printsrv/aopd.conf

## AOPLIMIT

The maximum number of objects that Infoprint Central displays after a search. This limit prevents the IBM HTTP Server - Powered by Apache 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 1 - 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:

1. 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.
2. If the HTTP server times out occasionally with the default limit, ask users to narrow their searches.
3. If the HTTP server times out repeatedly with the default limit, lower the limit or try increasing the HTTP server's timeout value.
4. 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 1 - 9999. If you specify an incorrect value, the default value is used.

For more information, see the AOPLIMIT environment variable.

**Default:** The limit that is 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 1 - 9999. If you specify an incorrect value, the default value is used.

For more information, see the AOPLIMIT environment variable.

**Default:** The limit that is 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, do not include any blanks in the value, and specify the entire value on one line.

### Default:

```
CLASSPATH=/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 that is used to locate Java files. Infoprint Central requires Java 7.1. This environment variable is optional. However, you must set it if you did not install the Java files in the default Java directory.

**Default:** /usr/lpp/java/J7.1.

**Example:** JAVA\_HOME=/usr/lpp/java/J7.1

## LANG

The language that is 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 that is used to format time and date information in messages. This locale overrides the locale in the LC\_TIME variable.

**Rule:** In the IBM HTTP Server - Powered by Apache environment variables file (bin/envvars), do not specify a value that is associated with a code page other than IBM-1047.

### Examples:

1. **aopstart** EXEC: LC\_ALL=Fr\_FR.IBM-297
2. bin/envvars file: LC\_ALL=Fr\_FR.IBM-1047

**Default:** C (also called POSIX)

## LIBPATH

The path that is 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:

1. Separate directories with a colon, do not include any blanks in the value, and specify the entire value on one line.
2. 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, do not include any blanks in the value, and specify the entire value on one line.

**Default:** None

### Example:

```
NLSPATH=existing values:/usr/lpp/Printsrv/%L/%N:/usr/lpp/Printsrv/En_US/%N
```

## Creating a started task to run the IBM HTTP Server - Powered by Apache

To create a started task to run the IBM HTTP Server - Powered by Apache:

1. Define the user ID for the started task that you want to run the server under.
2. Define a RACF (or other SAF) STARTED rule to map a started task to the user ID.

3. Create a procedure similar to this example:

```

/*-----
//IHS AE001 PROC ACTION='start',
// DIR='install_directory/etc/websrv1',
// CONF='conf/httpd.conf'
/*-----
//IHS EXEC PGM=BPXBATCH,
// PARM='SH &DIR/bin/apachectl -k &ACTION -f &CONF -DNO_DETACH',
// REGION=512M
//STDOUT DD PATH='&DIR/logs/proc.output',
// PATHOPTS=(OWRONLY,OCREAT,OTRUNC),
// PATHMODE=(SIRUSR,SIWUSR,SIRGRP,SIWGRP)
//STDERR DD PATH='&DIR/logs/proc.errors',
// PATHOPTS=(OWRONLY,OCREAT,OTRUNC),
// PATHMODE=(SIRUSR,SIWUSR,SIRGRP,SIWGRP)
// PEND

```

4. Run this command to start the IBM HTTP Server - Powered by Apache:

```
START procname
```

where *procname* is the procedure name; for example, IHSAE001.

## 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*.

Table 51 summarizes the customization tasks that are related to setting up security for Infoprint Central. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies to your installation.

Table 51. Customization tasks for Infoprint Central security

Task	Condition
"Authorizing the AOOPER group to RACF profiles" on page 353	Required
"Defining Infoprint Central users as z/OS UNIX users" on page 355	Required
"Activating the PRINTSRV class" on page 355	Optional: To define profiles in the PRINTSRV class
"Authorizing users to read the Printer Inventory" on page 355	Optional: To let Infoprint Central users view objects in the Printer Inventory
"Setting up security for printers" on page 356	Optional: To restrict who can work with printers
"Setting up security for NetSpool logical units (LUs)" on page 364	Optional: To restrict who can work with NetSpool LUs
"Setting up security for IP PrintWay job selection rules" on page 365	Optional: To restrict who can work with job selection rules
"Setting up security for print jobs" on page 367	Optional: To restrict who can work with print jobs
"Setting up security for Infoprint Server daemons" on page 373	Optional: To restrict who can view the status of daemons
"Setting up security for printer and printer pool definitions" on page 373	Optional: To restrict who can view printer definitions

Table 51. Customization tasks for Infoprint Central security (continued)

Task	Condition
“Requesting RACF notification and auditing” on page 373	Optional: To change default RACF notification and auditing

## Authorizing the AOOPER group to RACF profiles

You must give the user ID that starts Infoprint Server access to the RACF profiles in the OPERCMDS class that protect JES operator commands, and profiles in the JESSPOOL class that protect data sets. These profiles are optional. However, if you defined any of these profiles, you must give the user ID that starts Infoprint Server the minimum access.

The profile in the JESSPOOL class for JES2 and JES3 is `nodeid.userid.jobname.jobid.Ddsid.dsname`; the minimum access required for the AOOPER group is UPDATE. Table 52 and Table 53 list the profiles in the OPERCMDS class and the minimum access required.

Table 52. Profiles in the OPERCMDS class and minimum access required for the AOOPER group— JES2 systems. For *jesname*, substitute the name of the JES system. For example, substitute JES2.

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> .DISPLAY.TSUOUT	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> .MODIFY.TSUOUT	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 53. Profiles in the OPERCMDS class and minimum access required for the AOOPER group— JES3 systems. For *jesname*, substitute the name of the JES system. For example, substitute JES3.

Profile name in OPERCMDS class	Minimum access
<i>jesname</i> .CALL.WTR	UPDATE
<i>jesname</i> .CANCEL.DEV.dev	UPDATE

Table 53. Profiles in the OPERCMDS class and minimum access required for the AOPOPER group— JES3 systems (continued). For *jesname*, substitute the name of the JES system. For example, substitute JES3.

Profile name in OPERCMDS class	Minimum access
<i>jesname</i> .DISPLAY.D	READ
<i>jesname</i> .DISPLAY.U	READ
<i>jesname</i> .MODIFY.U	UPDATE
<i>jesname</i> .RESTART.DEV.dev	UPDATE
<i>jesname</i> .START.DEV.dev	UPDATE
<i>jesname</i> .VARY.DEV	UPDATE
MVS.MODIFY	UPDATE
MVS.VARY.NET	UPDATE

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.

**Guideline:** To limit access, use a JCL startup procedure, instead of a start command from the z/OS UNIX command line, to start Infoprint Server because you can associate a user ID with a JCL startup 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:

a. 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)
PERMIT N1.*.*.D*. * CLASS(JESSPOOL) ACCESS(UPDATE) ID(AOPOPER)
```

b. 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:**

- a. If you define more specific profiles in the OPERCMDS and JESSPOOL classes, give the AOPOPER group access to those profiles as well.
  - b. For information about the fields in the profile in the JESSPOOL class, see “Protecting print jobs with profiles in the JESSPOOL class” on page 370.
2. Refresh the OPERCMDS and JESSPOOL classes.

#### Examples:

- a. JES2: SETROPTS RACLIST(OPERCMDS JESSPOOL) REFRESH
- b. JES3: SETROPTS RACLIST(OPERCMDS JESSPOOL) REFRESH
3. If you authorize the AOOPER 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*.

**Tip:** Infoprint Central users do not need to be authorized to use TSO.

### Step 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 can 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 authorize 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, with 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 52.

**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.
- **OPERCMDS class:** Profiles in the OPERCMDs 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:

1. **IP PrintWay printers:** Define profiles in the PRINTSRV class to protect printers. Otherwise, any Infoprint Central user can work with any IP PrintWay printer.
2. **PSF printers:** Define profiles in the PRINTSRV class to protect printers, or define profiles in the OPERCMDs class to protect printer actions. Otherwise, any Infoprint Central user can work with any PSF printer.

Define profiles in the OPERCMDs 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 can 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 54 on page 357 lists the printer actions users can do in Infoprint Central and the minimum access that is required to the printer's profile in the PRINTSRV class.

Table 54. 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 that is specified in Printer Inventory	UPDATE
Change job selection - PSF printers only	Profile that is specified in Printer Inventory	UPDATE
Find and display printers <sup>1</sup>	Profile that is specified in Printer Inventory	READ
Interrupt - PSF printers only <sup>2</sup>	Profile that is specified in Printer Inventory	CONTROL
Pause - PSF printers only <sup>2</sup>	Profile that is specified in Printer Inventory	CONTROL
Ping and run traceroute	Profile that is specified in Printer Inventory	READ
Redirect - IP PrintWay printers only <sup>5</sup>	Profile that is specified in Printer Inventory	CONTROL
Repeat - PSF printers only <sup>2</sup>	Profile that is specified in Printer Inventory	CONTROL
Reset <sup>1, 2, 4, 5</sup>	Profile that is specified in Printer Inventory	CONTROL
Restore - IP PrintWay printers only	Profile that is specified in Printer Inventory	CONTROL
Space - PSF printers only <sup>2</sup>	Profile that is specified in Printer Inventory	CONTROL
Start	Profile that is specified in Printer Inventory	CONTROL
Stop printer and delete or hold the current print job <sup>2</sup>	Profile that is specified in Printer Inventory	CONTROL
Stop printer after the current print job completes	Profile that is specified in Printer Inventory	CONTROL
Turn offline <sup>1, 2, 4, 5</sup>	Profile that is specified in Printer Inventory	CONTROL
Turn online <sup>1, 2, 4, 5</sup>	Profile that is specified in Printer Inventory	CONTROL
View log	Profile that is specified in Printer Inventory	READ
View properties	Profile that is specified in Printer Inventory	READ



Table 54. 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
<ol style="list-style-type: none"> <li>1. 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 “Customizing the environment variables file for the IBM HTTP Server - Powered by Apache (bin/envvars)” on page 346.</li> <li>2. 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.</li> <li>3. To redirect an IP PrintWay printer to an alternative printer, CONTROL access is required to the profiles for both printers.</li> <li>4. 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 “Customizing the environment variables file for the IBM HTTP Server - Powered by Apache (bin/envvars)” on page 346.</li> <li>5. 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 356.</li> </ol>		

### 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:

- a. 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 now or in the future. Profile names can be up to 64 characters.
- b. Give the profile universal READ access if you want all Infoprint Central users to be able to view properties of the printer.
- c. For information about specifying the NOTIFY and AUDIT parameters on the RDEFINE command, see “Requesting RACF notification and auditing” on page 373.

**Example:** To define a profile that is 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 54 on page 357. 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:

- a. 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.
- b. 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*.
- c. 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.
- d. 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:**

- a. 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)

:
:
:

```

- b. 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.

- 1) 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**.

- 2) Inspect the /tmp/defs file to make sure the **modify** commands are acceptable.
- 3) 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:

- a. 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.
- b. 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*.
- c. 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:

- a. 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:

:

- b. 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.

- 1) 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.

- 2) Inspect the /tmp/defs file to make sure the **modify** commands are acceptable.
- 3) 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 355.

### 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 already defined profiles to protect printers in the OPERCMDS class, these profiles are suitable for Infoprint Central. However, be sure to give Infoprint Central users and the AOOPER 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 55 and Table 56 on page 362 list 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 that is required to the profiles.

*Table 55. Infoprint Central printer actions and minimum access required to profiles in the OPERCMDS class — JES2 systems.* For *jesname*, substitute the name of the JES system. For example, substitute JES2.

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	<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	<i>jesname</i> .DISPLAY.DEV	READ
Space - backward	<i>jesname</i> .BACKSP.DEV	UPDATE
Space - forward	<i>jesname</i> .FORWARD.DEV	UPDATE

Table 55. Infoprint Central printer actions and minimum access required to profiles in the OPERCMDS class — JES2 systems (continued). For *jesname*, substitute the name of the JES system. For example, substitute JES2.

To do this action in Infoprint Central	Access to this profile in the OPERCMDS class is required	This minimum access is required
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	<i>jesname</i> .DISPLAY.DEV	READ
Turn printer online	<i>jesname</i> .DISPLAY.DEV	READ
View properties	<i>jesname</i> .DISPLAY.DEV	READ
<b>Notes:</b> <ol style="list-style-type: none"> <li>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.</li> <li>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 “Customizing the environment variables file for the IBM HTTP Server - Powered by Apache (bin/envvars)” on page 346.</li> <li>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 “Customizing the environment variables file for the IBM HTTP Server - Powered by Apache (bin/envvars)” on page 346.</li> </ol>		

Table 56. Infoprint Central printer actions and the minimum access required to profiles in the OPERCMDS class — JES3 systems. For *jesname*, substitute the name of the JES system. For example, substitute JES3.

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

Table 56. Infoprint Central printer actions and the minimum access required to profiles in the OPERCMDS class — JES3 systems (continued). For *jesname*, substitute the name of the JES system. For example, substitute JES3.

To do this action in Infoprint Central	Access to this profile in the OPERCMDS class is required	This minimum access is required
Stop printer and delete the current print job	<i>jesname</i> .CANCEL.DEV. <i>dev</i> <i>jesname</i> .RESTART.DEV. <i>dev</i>	UPDATE UPDATE
Stop printer after the current print job completes	<i>jesname</i> .RESTART.DEV. <i>dev</i>	UPDATE
View properties	<i>jesname</i> .DISPLAY.D	READ
<b>Notes:</b> 1. 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 that is defined to JES. 2. 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 55 on page 361 and Table 56 on page 362.

**Guideline:** For information about specifying the NOTIFY and AUDIT parameters on the RDEFINE command, see “Requesting RACF notification and auditing” on page 373.

### Examples:

- a. 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)
```

- b. 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 that is required to the JES resource profiles, see Table 55 on page 361 and Table 56 on page 362.

### Examples:

- a. 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)
```

- b. 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(OPERCMDS) 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 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 353.
4. Refresh the OPERCMDS class.

**Example:** SETROPTS RACLIST(OPERCMDS) 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 355.

## 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 57 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 57. 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"> <li>AOP.NETSPOOL<sup>1</sup></li> <li>AOP.ADMINISTRATOR</li> </ul>	<ul style="list-style-type: none"> <li>CONTROL</li> <li>UPDATE</li> </ul>
Stop	<ul style="list-style-type: none"> <li>AOP.NETSPOOL<sup>1</sup></li> <li>AOP.ADMINISTRATOR</li> </ul>	<ul style="list-style-type: none"> <li>CONTROL</li> <li>UPDATE</li> </ul>
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 373.

**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 that is required, see Table 57 on page 364.

**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 353.

## 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 58 lists the job selection rule actions users can do in Infoprint Central and the minimum access that is required to the job selection rule's profile.

*Table 58. 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 that is specified in Printer Inventory	CONTROL
Stop	Profile that is 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:

- a. 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.
- b. For information about specifying the NOTIFY and AUDIT parameters on the RDEFINE command, see “Requesting RACF notification and auditing” on page 373.

**Example:** To define a profile that is 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:

- a. 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*.
- b. 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 355.

## 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 that you already defined in the OPERCMDS class to protect data sets on the JES spool are suitable for use with Infoprint Central. If profiles exist in the OPERCMDS class, give Infoprint Central users and the AOOPER 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 59 and Table 60 on page 369 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 that is required to the profile.

*Table 59. Infoprint Central print job actions and minimum access required to profiles in the OPERCMDS class - JES2.* For *jesname*, substitute the name of the JES system. For example, substitute 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 <i>jesname</i> .MODIFY.TSUOUT	UPDATE
Delete	<i>jesname</i> .MODIFY.BATOUT <i>jesname</i> .MODIFY.STCOUT <i>jesname</i> .MODIFY.TSUOUT	UPDATE
Edit properties	<i>jesname</i> .MODIFY.BATOUT <i>jesname</i> .MODIFY.STCOUT <i>jesname</i> .MODIFY.TSUOUT	UPDATE
Find and display print jobs	<i>jesname</i> .DISPLAY.BATOUT <i>jesname</i> .DISPLAY.STCOUT <i>jesname</i> .DISPLAY.TSUOUT	READ
Hold	<i>jesname</i> .MODIFY.BATOUT <i>jesname</i> .MODIFY.STCOUT <i>jesname</i> .MODIFY.TSUOUT	UPDATE
Move	<i>jesname</i> .MODIFY.BATOUT <i>jesname</i> .MODIFY.STCOUT <i>jesname</i> .MODIFY.TSUOUT	UPDATE
Release	<i>jesname</i> .MODIFY.BATOUT <i>jesname</i> .MODIFY.STCOUT <i>jesname</i> .MODIFY.TSUOUT	UPDATE

Table 59. Infoprint Central print job actions and minimum access required to profiles in the OPERCMDS class - JES2 (continued). For *jesname*, substitute the name of the JES system. For example, substitute JES2.

To do this action in Infoprint Central	Access to this profile in the JES2 OPERCMDS class is required	This minimum access is required
View log	<i>jesname</i> .DISPLAY.BATOUT <i>jesname</i> .DISPLAY.STCOUT <i>jesname</i> .DISPLAY.TSUOUT	READ
View properties	<i>jesname</i> .DISPLAY.BATOUT <i>jesname</i> .DISPLAY.STCOUT <i>jesname</i> .DISPLAY.TSUOUT	READ

Table 60. Infoprint Central print job actions and the minimum access required to profiles in the OPERCMDS class - JES3. For *jesname*, substitute the name of the JES system. For example, substitute 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

## 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 59 on page 368 and Table 60.

**Guideline:** For information about specifying the NOTIFY and AUDIT parameters on the RDEFINE command, see “Requesting RACF notification and auditing” on page 373.

### Examples:

- a. 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)
RDEFINE OPERCMDS (JES2.*.TSUOUT) 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)
```

- b. 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 that is required to the JES resource profiles, see Table 59 on page 368 and Table 60 on page 369.

#### Examples:

- a. JES2: To give users READ or UPDATE access to the broad profiles, enter these RACF commands:

```
PERMIT JES2.*.BATOUT CLASS(OPERCMDS) ACCESS(READ) ID(userid or groupid)
PERMIT JES2.*.STCOUT CLASS(OPERCMDS) ACCESS(READ) ID(userid or groupid)
PERMIT JES2.*.TSUOUT CLASS(OPERCMDS) ACCESS(READ) ID(userid or groupid)
```

```
PERMIT JES2.*.BATOUT CLASS(OPERCMDS) ACCESS(UPDATE) ID(userid or groupid)
PERMIT JES2.*.STCOUT CLASS(OPERCMDS) ACCESS(UPDATE) ID(userid or groupid)
PERMIT JES2.*.TSUOUT CLASS(OPERCMDS) ACCESS(UPDATE) ID(userid or groupid)
```

To give users UPDATE access to the specific profiles, enter these RACF commands:

```
PERMIT JES2.MODIFY.BATOUT CLASS(OPERCMDS) ACCESS(UPDATE) ID(userid or groupid)
PERMIT JES2.MODIFY.STCOUT CLASS(OPERCMDS) ACCESS(UPDATE) ID(userid or groupid)
PERMIT JES2.MODIFY.TSUOUT CLASS(OPERCMDS) ACCESS(UPDATE) ID(userid or groupid)
```

- b. JES3: To give some users READ access and other users UPDATE access to the broad profiles, enter these RACF commands:

```
PERMIT JES3.*.U CLASS(OPERCMDS) ACCESS(READ) ID(userid or groupid)
PERMIT JES3.*.U CLASS(OPERCMDS) ACCESS(UPDATE) ID(userid or groupid)
```

To give users UPDATE access to the specific profile, enter this RACF command:

```
PERMIT JES3.MODIFY.U CLASS(OPERCMDS) 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 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 353.
4. Refresh the OPERCMDS class.

**Example:** SETROPTS RACLIST(OPERCMDS) 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:

1. 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.
2. The profiles that exist in the JESSPOOL class are 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 61 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 that is required to the profile.

*Table 61. 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 61 on page 371.

**Guideline:** For information about specifying the NOTIFY and AUDIT parameters on the RDEFINE command, see “Requesting RACF notification and auditing” on page 373.

**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 that is required to the JES resource profiles, see Table 61 on page 371.

**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 373.

**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 62 lists the daemon action that users can do in Infoprint Central and the minimum access that is required to the AOP.DAEMON profile in the PRINTSRV class.

*Table 62. Infoprint Central daemon action 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 355.

## 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 needs to:

- 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 that is 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:

1. 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.
2. Do not use these reserved names: HC, INSTREAM, INTERNAL, OPERLOG, SYSIOSRS, SYSLOG, UNKNOWN.
3. 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*.

4. 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.
5. 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 you restart Infoprint Server.

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

**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 are 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 that is specified in the system logger log stream definition is used instead.

You can specify a value 0 - 59. A value of 0 means that Infoprint Central cannot display any messages and cannot display information about print jobs are 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:**

1. 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.
2. If you use the system logger for messages and also use Infoprint Central, the value needs to 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.
3. 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.
4. 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 34.
5. 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.
6. 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 122.
7. If you use **aoplogu** to view messages, the date and time format that is displayed in messages is controlled by the TZ and either the LC\_TIME or LC\_ALL environment variables. See “Setting environment variables for the Printer Inventory Manager” on page 46.

If you use Infoprint Central to view messages, the date and time format for the messages is controlled by the operating system that runs your web browser.

**Rules:**

1. 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.
2. 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 you restart Infoprint Server.

**Default:** `log-retention = 1`

**Example:** `log-retention = 3`

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 78.
- How to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 91.

---

## Customizing web browsers

Infoprint Central supports only Microsoft Internet Explorer 9.0 or later or Mozilla Firefox 24 Extended Support Release (ESR) or later web browsers.

You must customize the browser's Internet settings:

**Required settings:**

- Enable JavaScripts.
- Enable style sheets.

**Guidelines:**

1. Request that your browser always retrieve the most recent pages from the web so that the Infoprint Central refresh function works correctly. Do not use the default setting if it checks for newer versions of stored pages automatically.
2. Do not block or disable all cookies. Infoprint Central stores cookies on your system to save your search values.
3. Do not disable the meta refresh security option. Infoprint Central uses the meta refresh function to display your search results automatically.
4. For better performance, set your browser to cache files.

---

## Logging on to Infoprint Central from the web

To use Infoprint Central, you must log on with an authorized user ID and password. When you are done, you can log off Infoprint Central.

To start using Infoprint Central:

1. Enter a URL in the browser:

- For the English version:

`http://hostname:port/Infoprint/En_US/IPS.html`

If the IBM HTTP Server - Powered by Apache 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 IBM HTTP Server - Powered by Apache uses Secure Sockets Layer (SSL):

`https://hostname:port/Infoprint/Ja_JP/IPS.html`

*hostname*

The address of the z/OS system where the IBM HTTP Server - Powered by Apache is running.

*port*

The port where the IBM HTTP Server - Powered by Apache 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 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`

2. Log on to Infoprint Central with your user name and password for the host z/OS system.

**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.

To stop using Infoprint Central:

- Select your user name on the title bar.
- Select **Sign Out**.

---

## 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 that are 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 file names. 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:

1. You must code two quotation marks in the first position because the quotation mark is a special character that is 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.
2. 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*.

---

## Re-creating 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. If the search database becomes corrupted, delete the database and run the **sdbu** command to re-create 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.

To re-create the search database:

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. Repopulate the search database from the Printer Inventory:

`sdbu -z`

**Result:** The search database files are created in the `/var/Printsrv/searchdb` directory. If you specified a different base directory in the **base-directory** attribute, the search database files are created in that base directory instead of in the `/var/Printsrv` base directory.



## 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 63 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 63. Comparison of Windows printing methods

Function	Infoprint Port Monitor	SMB	IPP	LPR/LPD
Specify Infoprint Server job attributes	Yes	No	Yes <sup>1</sup>	No
Obtain print job status	Yes <sup>2</sup>	Yes <sup>3</sup>	Yes	No
Windows systems supported	Windows Server 2008, Windows 7, Windows Vista (Business and Enterprise editions) <sup>4</sup>		Any Windows systems that support IPP	Any Windows systems with an LPR command
Infoprint Port Monitor must be installed on Windows system	Yes <sup>5</sup>	No	No	No
Infoprint Server daemon that must be started	<b>aoplpd</b> (LPD)	None <sup>6</sup>	<b>aopipdd</b> (IPP Server)	<b>aoplpd</b> (LPD)
<div><div></div><div>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 153.</div><div>2. You can see the documents that are waiting to print. If an error occurred, you can view the Windows Event Log.</div><div>3. When the Windows job status window is open, z/OS system performance might be degraded.</div><div>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.</div><div>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.</div><div>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 149.</div></div>				

To use the Infoprint Port Monitor:

1. Make sure that 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 (**aop1pd**).
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 Infoprint Port Monitor (<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 Server is started and the Print Interface line printer daemon (**aop1pd**) 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, 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 (**aop1pd**) 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 they print 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 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 or later and the AFP Download Plus feature to write messages for print jobs and printers in the Infoprint Server common message log. 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.

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 64 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 64. Customization tasks for PSF for z/OS

Task	Condition
"Identifying PSF software requirements"	Required
"Creating PSF FSS and FSA definitions in the Printer Inventory"	Required
"Customizing PSF for the Printer Inventory" on page 386	Required
"Customizing PSF for z/OS to write messages to the common message log" on page 386	Optional: To write PSF messages in the common message log

---

### Identifying PSF software requirements

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:**

1. 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.
2. To submit print jobs to PSF printers with 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 with 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 the Printer Inventory

After you create FSS and FSA definitions in the Printer Inventory, you must customize PSF to use the Printer Inventory.

**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 processed and can view PSF messages for PSF printers.

- Using Infoprint Central, operators can view information about print jobs that PSF 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 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 V4R4 is running, you must restart the PSF functional subsystem (FSS) before the change to the **Log messages** field can take effect. If PSF V4R5 is running, you do not need to restart the PSF FSS because PSF automatically picks up a change to this field.

**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 . . . \_\_\_\_\_

PINST 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 that are 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 V2R2 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

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. Customize 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 31
  - e. “Creating Infoprint Server directories” on page 32
  - f. “Creating and editing the Infoprint Server configuration file (aopd.conf)” on page 37
  - g. “Setting environment variables for the Printer Inventory Manager” on page 46
  - h. “Setting up security for the Printer Inventory and operator commands” on page 52
  - i. “Setting permissions for directories and executable files -- aopsetup” on page 59
  - j. “Customizing startup and shutdown commands and procedures” on page 61
  - k. “Enabling Infoprint Server ISPF panels” on page 76
  - l. “Enabling dynamic configuration” on page 78
  - m. “Editing the system configuration definition for Printer Inventory Manager” on page 87

- \_\_\_ n. "Sending non-console messages to the hardcopy log" on page 94
- \_\_\_ o. "Setting up the MVS system logger for the common message log" on page 98
- \_\_\_ p. "Backing up and restoring the Printer Inventory" on page 114
- \_\_\_ q. "Saving messages in the common message log in the z/OS UNIX file system" on page 120
- \_\_\_ r. "Reducing the size of the common message log in the z/OS UNIX file system" on page 122
- \_\_\_ s. "Setting Workload Manager goals" on page 123
- \_\_\_ 4. Customize Print Interface: See Chapter 4, "Customizing Print Interface," on page 125.
  - \_\_\_ a. "Identifying software requirements for Print Interface" on page 126
  - \_\_\_ b. "Common customization for Print Interface" on page 126
    - \_\_\_ 1) "Customizing z/OS Communications Server" on page 127
    - \_\_\_ 2) "Updating the BPXPRMxx member of SYS1.PARMLIB" on page 129
    - \_\_\_ 3) "Editing the aopd.conf file for Print Interface" on page 130
    - \_\_\_ 4) "Writing a filter program" on page 131
    - \_\_\_ 5) "Customizing JES to specify no blank truncation" on page 137
    - \_\_\_ 6) "Setting environment variables for Print Interface" on page 137
    - \_\_\_ 7) "Editing the system configuration definition for Print Interface" on page 138
  - \_\_\_ c. "Customizing the z/OS UNIX printing commands" on page 140
  - \_\_\_ d. "Customizing the AOPPRINT JCL procedure" on page 141
  - \_\_\_ e. "Customizing the Print Interface LPD" on page 142
    - \_\_\_ 1) "Customizing z/OS Communications Server (TCP/IP)" on page 143
    - \_\_\_ 2) "Editing the aopd.conf file for the LPD" on page 145
    - \_\_\_ 3) "Editing the system configuration definition for the LPD" on page 146
    - \_\_\_ 4) "Using the LPD compatibility filter" on page 147
    - \_\_\_ 5) "Customizing LPR clients" on page 147
  - \_\_\_ f. "Customizing the z/OS SMB server to use Print Interface" on page 149
  - \_\_\_ g. "Customizing the Print Interface IPP Server" on page 150
    - \_\_\_ 1) "Ensuring sufficient memory is available to start the IPP Server" on page 151
    - \_\_\_ 2) "Editing the aopd.conf file for the IPP Server" on page 151
    - \_\_\_ 3) "Setting environment variables for the IPP Server" on page 152
    - \_\_\_ 4) "Editing the system configuration definition for the IPP Server" on page 153
    - \_\_\_ 5) "Customizing the IPP workstation client" on page 153
  - \_\_\_ h. "Customizing SAP R/3" on page 155
 

If the SAP R/3 application server runs on the z/OS system:

    - \_\_\_ 1) "Defining an SAP user for the callback daemon" on page 156

- \_\_\_ 2) "Creating the SAP callback daemon configuration file (aopsapd.conf)" on page 157
- \_\_\_ 3) "Setting environment variables for the SAP OMS" on page 159
- \_\_\_ 4) "Defining a ROMS and LOMS to the SAP R/3 application server" on page 159
- \_\_\_ 5) "Modifying the LOMS to specify job attributes" on page 160
- \_\_\_ 6) "Creating an SAP R/3 output device" on page 161
- If the SAP R/3 application server does not run on the z/OS system:
  - \_\_\_ 1) "Customizing the Print Interface LPD" on page 142
  - \_\_\_ i. "Customizing the Print Interface subsystem" on page 163
- \_\_\_ 5. Customize Transform Interface: See Chapter 5, "Customizing Transform Interface," on page 167.
  - \_\_\_ a. "Identifying software requirements for Transform Interface" on page 168
  - \_\_\_ b. "Editing the aopd.conf file for the Transform Manager" on page 168
  - \_\_\_ c. "Creating the transform configuration file (aopxfd.conf)" on page 169
  - \_\_\_ d. "Creating the remote transform configuration file (aoprxf.conf)" on page 175
  - \_\_\_ e. "Setting environment variables for Transform Interface" on page 176
  - \_\_\_ f. "Customizing the TCP/IP firewall" on page 178
  - \_\_\_ g. "Setting up security for transform resources" on page 178
  - \_\_\_ h. "Creating a code page conversion table" on page 178
- \_\_\_ 6. Customize NetSpool: See Chapter 6, "Customizing NetSpool," on page 179.
  - \_\_\_ a. "Updating the BPXPRMxx member of SYS1.PARMLIB" on page 180
  - \_\_\_ b. "Editing the aopd.conf file for NetSpool" on page 181
  - \_\_\_ c. "Customizing the NetSpool message log" on page 182
  - \_\_\_ d. "Customizing the NetSpool startup procedure" on page 182
  - \_\_\_ e. "Changing Language Environment runtime options" on page 189
  - \_\_\_ f. "Writing NetSpool exit and filter programs" on page 189
  - \_\_\_ g. "Setting environment variables for NetSpool" on page 202
  - \_\_\_ h. "Editing the system configuration definition for NetSpool" on page 203
- \_\_\_ 7. Customize IP PrintWay basic mode: See Chapter 7, "Customizing IP PrintWay," on page 207.
  - \_\_\_ a. "Comparing IP PrintWay extended mode to basic mode" on page 210
  - \_\_\_ b. "Identifying software requirements for IP PrintWay" on page 213
  - \_\_\_ c. "Defining printer host names in the DNS" on page 240
  - \_\_\_ d. "Customizing z/OS Communications Server" on page 241
  - \_\_\_ e. "Creating a VTAM APPL resource definition for IP PrintWay" on page 245
  - \_\_\_ f. "Updating the BPXPRMxx member of SYS1.PARMLIB (basic mode)" on page 313
  - \_\_\_ g. "APF-authorizing SYS1.IMAGELIB (basic mode)" on page 314
  - \_\_\_ h. "Defining the IP PrintWay functional subsystem (basic mode)" on page 314

- \_\_\_ i. "Creating FSS and FSA definitions in the Printer Inventory (basic mode)" on page 325
- \_\_\_ j. "Creating the transmission-queue data set (basic mode)" on page 328
- \_\_\_ k. "Creating the IP PrintWay message-log data set (basic mode)" on page 331
- \_\_\_ l. "Customizing the IP PrintWay startup procedure (basic mode)" on page 332
- \_\_\_ m. "Writing IP PrintWay exit and filter programs" on page 246
- \_\_\_ n. "Changing Language Environment runtime options (basic mode)" on page 337
- \_\_\_ o. "Customizing PSF for sharing network printers" on page 337
- \_\_\_ 8. Customize PrintWay extended mode: See Chapter 7, "Customizing IP PrintWay," on page 207.
  - \_\_\_ a. "Comparing IP PrintWay extended mode to basic mode" on page 210
  - \_\_\_ b. "Identifying software requirements for IP PrintWay" on page 213
  - \_\_\_ c. "Editing the aopd.conf file for IP PrintWay (extended mode)" on page 213
  - \_\_\_ d. "Increasing the space available in /var/Printsrv directory (extended mode)" on page 215
  - \_\_\_ e. "Updating the BPXPRMxx member of SYS1.PARMLIB (extended mode)" on page 215
  - \_\_\_ f. "Creating IP PrintWay job selection rules (extended mode)" on page 216
  - \_\_\_ g. "Authorizing IP PrintWay to select data sets from the JES spool (extended mode)" on page 222
  - \_\_\_ h. "Authorizing IP PrintWay to write SMF type 6 records (extended mode)" on page 223
  - \_\_\_ i. "Setting environment variables for IP PrintWay (extended mode)" on page 224
  - \_\_\_ j. "Editing the system configuration definition for IP PrintWay (extended mode)" on page 232
  - \_\_\_ k. "APF-authorizing SYS1.IMAGELIB (extended mode)" on page 240
  - \_\_\_ l. "Customizing from AFP transforms for IP PrintWay (extended mode)" on page 240
  - \_\_\_ m. "Defining printer host names in the DNS" on page 240
  - \_\_\_ n. "Customizing z/OS Communications Server" on page 241
  - \_\_\_ o. "Creating a VTAM APPL resource definition for IP PrintWay" on page 245
  - \_\_\_ p. "Writing IP PrintWay exit and filter programs" on page 246
  - \_\_\_ q. "Customizing PSF for sharing network printers" on page 337
  - \_\_\_ r. "Customizing AOPCMND for starting and stopping printers" on page 337
- \_\_\_ 9. Customize Infoprint Central: See Chapter 8, "Customizing Infoprint Central," on page 339.
  - \_\_\_ a. "Identifying software requirements for Infoprint Central" on page 340
  - \_\_\_ b. "Editing the aopd.conf file for Infoprint Central" on page 341



- \_\_\_ c. "Customizing the IBM HTTP Server - Powered by Apache" on page 343
- \_\_\_ d. "Setting up security for Infoprint Central" on page 352
- \_\_\_ e. "Editing the system configuration definition for Infoprint Central" on page 374
- \_\_\_ f. "Customizing web browsers" on page 376
- \_\_\_ g. "Logging on to Infoprint Central from the web" on page 377
- \_\_\_ h. Customize PSF for z/OS: See Chapter 10, "Customizing PSF for z/OS to use the Printer Inventory," on page 385
- \_\_\_ i. "Customizing JES3 console standards (CONSTD statement)" on page 378
- \_\_\_ j. "Re-creating the search database" on page 378
- \_\_\_ 10. Customize Infoprint Port Monitor for Windows: See Chapter 9, "Installing and customizing the Infoprint Server Windows client," on page 381.
- \_\_\_ 11. Customize PSF for z/OS: See Chapter 10, "Customizing PSF for z/OS to use the Printer Inventory," on page 385.
  - \_\_\_ a. "Identifying PSF software requirements" on page 385
  - \_\_\_ b. "Creating PSF FSS and FSA definitions in the Printer Inventory" on page 385
  - \_\_\_ c. "Customizing PSF for the Printer Inventory" on page 386
  - \_\_\_ d. "Customizing PSF for z/OS to write messages to the common message log" on page 386



---

## Appendix B. Infoprint Server files

This appendix describes these Infoprint Server files that let you customize Infoprint Server:

- “aopd.conf -- Infoprint Server configuration file”
- “aopmsg.conf -- Infoprint Server message configuration file” on page 406
- “aoprxf.conf -- Infoprint Server remote transform configuration file” on page 408
- “aopxfd.conf -- Infoprint Server transform configuration file” on page 409
- “aopsapd.conf -- Infoprint Server SAP callback daemon configuration file” on page 410

---

### 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 ]*
[ start-daemons = {[ippd] [lpd] [netd] [outd] [ssid] [subd] [xfd]}}
[ xcf-group-qualifier = q]
```

**Note:** Attributes that are 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 that is 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 that is 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 that is 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:**

1. 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.
2. 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 you restart 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.

**Rule:** Do not change this attribute while Infoprint Server (including NetSpool, IP PrintWay basic mode, and Infoprint Central) or the Printer Inventory for PSF is running. If you change this attribute, you must stop and restart:

- All Infoprint Server daemons
- NetSpool
- IP PrintWay basic mode
- IBM HTTP Server - Powered by Apache (when running Infoprint Central)
- PSF for z/OS (when running Printer Inventory for PSF)

**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:**

1. 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.
2. Do not use these reserved names: HC, INSTREAM, INTERNAL, OPERLOG, SYSIOSRS, SYSLOG, UNKNOWN.
3. 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*.

4. 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.
5. 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 you restart Infoprint Server.

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

**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 78.

**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 you restart Infoprint Server.

**Default:** dynamic-configuration = no

**Example:** dynamic-configuration = yes

**ebcdic-codepage = codepage**

The name of an IBM-supplied or custom EBCDIC code page that is 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:

1. 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.
2. 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 you restart 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 four 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.

**Rule:** Do not change this attribute while Infoprint Server (including NetSpool, IP PrintWay basic mode, and Infoprint Central) or the Printer Inventory for PSF is running. If you change this attribute, you must stop and restart:

- All Infoprint Server daemons
- NetSpool

- IP PrintWay basic mode
- IBM HTTP Server - Powered by Apache (when running Infoprint Central)
- PSF for z/OS (when running Printer Inventory for PSF)

**Default:** inventory = A0P1

**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:**

1. Make sure that the port is not used by any other service on the same TCP/IP stack.
2. 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.
3. 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 you restart 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:**

1. 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 you restart Infoprint Server.
2. If you use Infoprint Central and change this attribute, restart the IBM HTTP Server - Powered by Apache.

**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:**

1. 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.
2. The job ID that Print Interface returns to the job submitter does not contain this prefix.

**Rules:**

1. 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.
2. 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 you restart 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 are removed from the JES spool.

You can specify a value 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 are removed from the JES spool.

**Guidelines:**

1. 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.
2. 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 34.
3. 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.

4. 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 122.
5. If you use **aoplogu** to view messages, the date and time format that is displayed in messages is controlled by the TZ and either the LC\_TIME or LC\_ALL environment variables. See “Setting environment variables for the Printer Inventory Manager” on page 46.  
  
If you use Infoprint Central to view messages, the date and time format for the messages is controlled by the operating system that runs your web browser.

#### Rules:

1. 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.
2. 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 you restart 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.

**Tip:** 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:

1. Make sure that the port is not used by any other service on the same TCP/IP stack.

2. 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 127.
3. 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.
4. 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 you restart 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 that are specified in the **log-retention** attribute limits the amount of information that Infoprint Server keeps in the Historical Inventory.

Table 65 shows the number of documents 1 MB in the Historical Inventory can contain.

*Table 65. Number of documents in the Historical Inventory*

If print jobs on the JES spool contain:	1 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 1 MB

For example:

- Documents processed in one day = 100,000
- Days to retain information = 3 (**log-retention** = 3)
- Documents in 1 MB = 1350 documents

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

**Rules:**

1. 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.
2. 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 you restart 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 by 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 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:**

1. 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 by using the IP address instead of the printer's host name.
2. 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 you restart 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:**

1. 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.
2. 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 you restart Infoprint Server.
3. This attribute does not apply to IP PrintWay basic mode.

**Default:** smf-recording = yes

**Example:** smf-recording = no

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

The Infoprint Server daemons that are started when you use the **aopstart** command. This attribute is ignored when the **Operating mode** field on the ISPF Configuration panel is set to **z/OS 2.2**.

You can specify one or more of these daemon names that are 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 an Internet Protocol network or to email destinations.

**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 daemon **aopd** always starts, regardless of which daemons you specify in this attribute. To start only the Printer Inventory Manager daemon, 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
```

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

1. You added value **ssid**.
2. Dynamic configuration is not enabled.
3. 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:

1. In this example, all daemons start:  

```
start-daemons = {ippd lpd netd outd ssid subd xfd}
```
2. In this example, only the Printer Inventory Manager daemon starts:  

```
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: AOPpppp[*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 you restart Infoprint Server.

**Default:** None

**Example:** xcf-group-qualifier = 1

## Usage notes

- For information about dynamic configuration, see “Enabling dynamic configuration” on page 78.
- The Infoprint Server configuration file is in the directory that is specified in the **AOPCONF** environment variable. If **AOPCONF** is not set, the default location is /etc/Printsrv/aopd.conf.
- 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**.
- 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 wanted.
  - 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 wanted.

---

## 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:**

1. 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.
2. 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 you restart 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:**

1. 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.
2. If your installation uses the MPFLSTxx Parmlib member for message automation, you might want to specify the same message IDs in MPFLSTxx.
3. If your installation uses an automated operations tool, the tool can process these messages.

**Rules:**

1. If you specify this attribute, also specify **hardcopy-messages = list**.
2. 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.
3. 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 you restart Infoprint Server.

**Default:** None.

**Example:** `hardcopy-message-list={A0P3614I A0P3803E}`

## Usage notes

1. For information about dynamic configuration, see “Enabling dynamic configuration” on page 78.
2. The Infoprint Server message configuration file is an optional file. If present, it is in the directory that is 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 wanted.
  - Start comments with the # symbol.
  - Include blank lines, if wanted.
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 you restart 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:

1. remote-transform-manager-ip-address = linux1.xyz.com
2. 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 is in the directory that is specified in the AOPRXF\_CONF environment variable. 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 wanted.
  - Type blank characters before or after equal signs, if wanted.
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.

---

## aopxofd.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.

### 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 is in the directory that is specified in the AOPXFD\_CONF environment variable. If this environment variable is not set, the default location is /etc/Printsrv/aopxofd.conf.
2. To create this configuration file, copy sample file /usr/lpp/Printsrv/samples/aopxofd.conf to /etc/Printsrv/aopxofd.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 wanted.
  - 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 that is enclosed in quotation marks, end the line with a backslash and do not type any characters, even blank characters, after the backslash. For example:  

```
environment = {AOP_FONTLIB -> "sys1.font300  \
sys1.sfntilib 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 that is
    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 that is defined to SAP for use by the Callback daemon.

userid = userid
    The user ID that is defined to SAP for use by the Callback daemon.

password = password
    The password that is defined to SAP for use by the user ID.

language = language
    The language that is used for messages from the SAP R/3 application server.
```

### Example

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 = xxxxxxx
  language = EN
;

```

## Usage notes

1. All attributes are required.
2. The configuration file is in the directory that is 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 you start 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 wanted.
  - 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 wanted.



---

## Appendix C. Environment variables

This appendix summarizes the environment variables that affect processing of Infoprint Server. Table 66 on page 414 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 46.

### **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 62.

**Note:** If dynamic configuration is enabled, Infoprint Server ignores some environment variables that are specified in the **aopstart** EXEC. These environment variables are marked with <sup>1</sup> in the table. You must specify the equivalent attribute in the system configuration definition in the Printer Inventory. For the equivalent attributes, see Table 14 on page 81.

### **aopdaemon *envvar* file**

This file is specified with the **aopdaemon** command or the VARFILE DD statement in the AOPDEMON JCL procedure. For information about how to set variables, see “Creating an environment variables file (*envvar*)” on page 65.

### **AOPSTART JCL procedure STDENV data definition (DD)**

For information about how to set variables, see “Customizing startup and shutdown commands and procedures” on page 61.

### **IP PrintWay basic mode startup procedure (ANFWPROC) STDENV DD**

For information about how to set variables, see “Editing the ANFWPROC startup procedure (basic mode)” on page 333.

### **IBM HTTP Server - Powered by Apache `bin/envvars` environment variables file**

You must set variables in this file only if you use Infoprint Central. For information about how to set variables, see “Customizing the environment variables file for the IBM HTTP Server - Powered by Apache (`bin/envvars`)” on page 346.

### **Rules:**

1. If you specify an environment variable in the AOPSTART JCL procedure STDENV DD, you do not need to specify it in the **aopstart** EXEC.
2. If you specify an environment variable in the AOPSTART JCL procedure STDENV DD, the **aopstart** EXEC, or the *envvar* file VARFILE DD, also specify it in the other files if applicable (`etc/profile` file, ANFWPROC STDENV DD, `bin/envvars` file). For example, you must set the AOPCONF environment variable in the **aopstart** EXEC or the *envvar* file, in the `/etc/profile` file, and in the `bin/envvars` file.

You must always set required variables. You must set optional variables if the default value is not suitable for your installation.



Table 66. Where to set environment variables

Environment variable	Location where set				
	/etc/profile file	aopstart EXEC or <i>envvar</i> file	AOPSTART STDENV DD	ANFWPROC STDENV DD	bin/envvars file
_BPX_SHAREAS	No	Optional	No	No	No
_BPX_UNLIMITED_OUTPUT	No	Optional	No	No	No
_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 <sup>1</sup>	No	Optional	No	No	No
AOP_APPLID <sup>1</sup>	No	Optional	No	No	No
AOP_BLANK_TRUNCATION_CLASSES <sup>1</sup>	No	Optional	No	No	No
AOP_IGNORE_DCF_ROUTING_ERRORS <sup>1</sup>	No	Optional	No	No	No
AOP_MAIL_DO_NOT_ADD_SUFFIX <sup>1</sup>	No	Optional	No	Optional	No
AOP_MAIL_PRESERVE_SUFFIXES <sup>1</sup>	No	Optional	No	Optional	No
AOP_MAIL_USE_FIRST_ADDRESS <sup>1</sup>	No	Optional	No	No	No
AOP_MAXTHREADTASKS <sup>1</sup>	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 <sup>2</sup>	Optional	Optional	No	No	No
AOP_SELECT_WORK_FROM_HOLD_QUEUE <sup>1</sup>	No	Optional	No	No	No
AOP_SUPPRESS_POST_UNIX_FILTER_FORMATTING <sup>1</sup>	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 <sup>3</sup>	Optional	Optional	Optional	No	No
AOPIPPD_MAXTHREADTASKS <sup>1</sup>	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 <sup>1</sup>	No	Optional	No	No	No
AOPMAILER <sup>1</sup>	No	Optional <sup>3</sup>	No	Optional	No
AOPMAILER_OPTIONS <sup>1</sup>	No	Optional	No	Optional	No
AOPMSG_CONF	Optional	Optional <sup>1</sup>	No	No	No
AOPNETD_MAXTHREADTASKS <sup>1</sup>	No	Optional	No	No	No
AOPNETD_USE_FIXED_JOBID <sup>1</sup>	No	Optional	No	No	No
AOPNETD_USE_UNALTERED_JOBID <sup>1</sup>	No	Optional	No	No	No
AOPTIONS	Optional	No	No	No	No
AOPOUTD_MAXTHREADTASKS <sup>1</sup>	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
AOPSSID_MAXTHREADTASKS <sup>1</sup>	No	Optional	No	No	No

Table 66. Where to set environment variables (continued)

Environment variable	Location where set				
	/etc/profile file	aopstart EXEC or <i>envvar</i> file	AOPSTART STDENV DD	ANFWPROC STDENV DD	bin/envvars file
AOPSUBD_MAXTHREADTASKS <sup>1</sup>	No	Optional	No	No	No
AOPTRACEBYTES <sup>3</sup>	Optional	Optional	Optional	No	No
AOPTRACEDIR <sup>3</sup>	No	Optional	No	No	Optional
AOPTRACEMAXFILES <sup>3</sup>	Optional	Optional	Optional	No	No
AOPTRACEON <sup>3</sup>	Optional	Optional	Optional	No	Optional
AOPTRACEON_DATA <sup>3</sup>	Optional	Optional	Optional	No	No
AOPTRACEON_FILTER <sup>3</sup>	Optional	Optional	Optional	No	No
AOPTRACEON_MSGLOG <sup>3</sup>	Optional	Optional	Optional	No	No
AOPVALIDATEDB	Optional	Optional	Optional	No	No
AOPWSMD_MAXTHREADTASKS <sup>1</sup>	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	Optional	No	Optional	Required
MANPATH	Required	No	No	No	No
NLSPATH	Required	Optional	No	Optional	Required
PATH	Required	Optional	No	Optional	No
PRINTER or LPDEST	Optional	No	No	No	No
STEPLIB	Optional	Optional	No	No	No
TZ	Optional	Optional	Optional	No	Optional
<p>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 <b>aopstart</b> EXEC or <i>envvar</i> file. Infoprint Server ignores it if specified. For the equivalent attributes in the system configuration definition, see the description of the environment variable.</p> <p>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>.</p> <p>3. These environment variables control tracing and dumping. For more information about them, see <i>z/OS Infoprint Server Messages and Diagnosis</i>.</p>					

**\_BPX\_SHAREAS**

Causes the spawn() callable service to create some Infoprint Server subtasks in the parent's address space. Do not remove this variable or change its value.

**\_BPX\_UNLIMITED\_OUTPUT**

Specifies that Infoprint Server daemons can 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 or **aopdemon** command. Do not remove this variable or change its value.

**\_BPXK\_SETIBMOPT\_TRANSPORT**

The TCP/IP job name that is associated with the TCP/IP stack you want to use. This variable is optional. However, you must set it if your installation 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 **i conv** uses customized TCP/IP code page translation tables. Set the value to **C** if you 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 must not define this variable because system performance might be degraded.

#### **AOP\_ALLOW\_ALL\_CHARACTERS\_IN\_LINE\_DATA**

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:**

1. 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.
2. 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.
3. 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:**

1. If dynamic configuration is enabled, specify the equivalent **allow-all-characters-in-line-data** attribute in the system configuration definition.
2. 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 that is 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:

1. If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **applid** attribute in the system configuration definition.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
3. 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.

**Value:** 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:

1. Specify this environment variable only if line or text data prints with extra blank characters at the end of lines.
2. This environment variable can slow performance and cause IP PrintWay to use more system resources.
3. Before JES gives 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 can 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:**

1. If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **blank-truncation-classes** attribute in the system configuration definition.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
3. 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 that are 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:**

1. Set this environment variable to YES if your IP PrintWay job-selection rules cause IP PrintWay to select many 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.
2. 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.
3. 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:**

1. 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.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
3. 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:

1. 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.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
3. 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 that are 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:

1. Specify the suffixes that users might specify in file names. For example, you might specify the .doc suffix for Microsoft Word documents.
2. 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:

1. Begin each suffix with a period.
2. 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.
3. If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **mail-preserve-suffixes** attribute in the system configuration definition.
4. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

5. 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).

##### **Values:**

- YES** IP PrintWay sends each document to the email address or addresses that are specified for the first document in the print job.
- NO** IP PrintWay sends each document to the email address or addresses that are specified for the document.

##### **Tips:**

1. 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 that are specified for the first document in the print job.
2. IP PrintWay basic mode ignores this environment variable. IP PrintWay basic mode always sends each document to the email address or addresses that are specified for the document unless the **Concatenate job** option is selected in the **Dataset grouping** field in the printer definition.

##### **Rules:**

1. 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.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart EXEC**.
3. 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: **aopipdd**, **aoplpd**, **aopnetd**, **aopoutd**, **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:**



1. 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 is reached, threads are 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 by using the SETOMVS command.

2. Do not set a value less than 32.
3. If you increase the MAXTHREADTASKS limit, you might need to increase the maximum region size because each concurrent task requires more storage:
  - If you use a JCL startup procedure to start Infoprint Server, increase the region size on the EXEC statement in the procedure.
  - If you use a start command from the z/OS UNIX command line 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). Be careful when you are using the SHRLIBRGNSIZE OMVS parameter in a 31-bit environment because it limits the size of usable regions across the system.
  - If you enter a start 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.
4. Do not set a high limit because each concurrent task requires some storage below the 16 MB line. If the limit is too high, an abend can occur.
5. If you exceed the default limit (200) when many printers are waiting for operator intervention at the same time, you might need to increase the limit in the AOPOUTD\_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:

1. If dynamic configuration is enabled, specify the equivalent **max-thread-tasks** attribute in the system configuration definition.
2. 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:

AOPOUTD\_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:

- |            |                                                                                                           |
|------------|-----------------------------------------------------------------------------------------------------------|
| <b>YES</b> | 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. |
| <b>NO</b>  | UNIX return codes:                                                                                        |
| 0          | Successful. However, a warning or data stream error might occur during the transform.                     |
| 1          | The command was not accepted, or the transform failed.                                                    |

#### Notes:

1. For information about whether a transform command supports this environment variable, see the documentation for the transform.
2. 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).

3. 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.
4. 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.
5. 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:

1. 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.

2. You need to 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.
3. 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.
4. 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:**

- |                   |                                                                                                                                                                                                      |
|-------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b><u>YES</u></b> | IP PrintWay extended mode selects new held output groups from the JES spool. You can use Infoprint Central to move the held output groups to different IP PrintWay printers before you release them. |
| <b>NO</b>         | IP PrintWay extended mode does not select new held output groups from the JES spool. You can use JES or SDSF commands to release individual data sets in an output group for printing.               |

##### **Rules:**

1. 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.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
3. 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 that is 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:**

1. 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.
2. 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 that is returned by DLL filters.

**Rules:**

1. 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.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
3. 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:**

1. For information about how to assign an SNMP write community name to printers, see the documentation for the printer or contact the printer manufacturer.
2. 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.
3. To do printer actions, Infoprint Central users must have CONTROL access to the RACF profile in the PRINTSRV class that protects the printer.
4. 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, *envvar* file, or JCL startup procedure. This environment variable is optional.

Specify this environment variable only in the IBM HTTP Server - Powered by Apache environment variables file (bin/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 IBM HTTP Server - Powered by Apache environment variables file (bin/envvars) and in the **aopstart** EXEC or *envvar* file must match.

**Default:** /etc/Printsrv/aopd.conf

### AOPIPPD\_MAXTHREADTASKS

The MAXTHREADTASKS limit for the Internet Printing Protocol (IPP) daemon, **aopipd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aopipd** can have concurrently active. This environment variable is optional.

**Tip:** For more information, see the “Tips” for the AOP\_MAXTHREADTASKS environment variable.

#### Rules:

1. If dynamic configuration is enabled, specify the equivalent **aopipd-max-thread-tasks** attribute in the system configuration definition.
2. If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

**Default:** The limit that is 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 IBM HTTP Server - Powered by Apache 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 1 - 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:**

1. 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.
2. If the HTTP server times out occasionally with the default limit, ask users to narrow their searches.
3. If the HTTP server times out repeatedly with the default limit, lower the limit or try increasing the HTTP server's timeout value.
4. 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 1 - 9999. If you specify an incorrect value, the default value is used.

For more information, see the AOPLIMIT environment variable.

**Default:** The limit that is 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 1 - 9999. If you specify an incorrect value, the default value is used.

For more information, see the AOPLIMIT environment variable.

**Default:** The limit that is 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, **aop1pd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aop1pd** can have concurrently active. This environment variable is optional.

**Tip:** For more information, see the "Tips" for the AOP\_MAXTHREADTASKS environment variable.

**Rules:**

1. If dynamic configuration is enabled, specify the equivalent **aoplpd-max-thread-tasks** attribute in the system configuration definition.
2. If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

**Default:** The limit that is specified in the AOP\_MAXTHREADTASKS environment variable is used. If that environment variable is not specified, the limit is 200.

## AOPMAILER

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:

1. Start the path name with a slash (/).
2. If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **mailer-path-name** attribute in the system configuration definition.
3. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
4. 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:

1. If you run IP PrintWay extended mode and dynamic configuration is enabled, specify the equivalent **mailer-options** attribute in the system configuration definition.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.
3. 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**. If dynamic configuration is enabled, you did not create this configuration file or 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 more information, see the “Tips” for the AOP\_MAXTHREADTASKS environment variable.

#### **Rules:**

1. If dynamic configuration is enabled, specify the equivalent **aopnetd-max-thread-tasks** attribute in the system configuration definition.
2. If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

**Default:** The limit that is 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 that is 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 large number of PSF FSAs or IP PrintWay basic mode FSAs. Contact IBM for more information.

#### **Rules:**

1. If dynamic configuration is enabled, specify the equivalent **netspool-use-fixed-jobid** attribute in the system configuration definition.
2. 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:**

1. If dynamic configuration is enabled, specify the equivalent **netspool-use-unaltered-jobid** attribute in the system configuration definition.
2. 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

## **APOPTIONS**

A string of job attributes and values that are in effect for each **lp** command. The **lp** command includes the attributes that are 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 APOPTIONS 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*.

## **APOUTD\_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:**

1. If you exceed the default limit (200) when many 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*.
2. See the "Tips" for the AOP\_MAXTHREADTASKS environment variable.

**Rules:**

1. If dynamic configuration is enabled, specify the equivalent **aopoutd-max-thread-tasks** attribute in the system configuration definition.

2. If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

**Default:** The limit that is specified in the AOP\_MAXTHREADTASKS environment variable is used. If that environment variable is not specified, the limit is 200.

### AOPPATH

The path where job attribute files, which are 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 is considered an authenticated z/OS user ID for doing RACF checks. This is the user ID that IP PrintWay sends to the Infoprint Server LPD during the resubmit for filtering function. 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 can 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:

1. For a secure system, do not set this environment variable.
2. 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.
3. 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 with 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`

### 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 more information, see the “Tips” for the AOP\_MAXTHREADTASKS environment variable.

#### Rules:

1. If dynamic configuration is enabled, specify the equivalent **aopssid-max-thread-tasks** attribute in the system configuration definition.
2. If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

**Default:** The limit that is 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 more information, see the “Tips” for the AOP\_MAXTHREADTASKS environment variable.

#### Rules:

1. If dynamic configuration is enabled, specify the equivalent **aopsubd-max-thread-tasks** attribute in the system configuration definition.
2. If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

**Default:** The limit that is 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 processing. 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 more information, see the “Tips” for the AOP\_MAXTHREADTASKS environment variable.

### Rules:

1. If dynamic configuration is enabled, specify the equivalent **aopwsmd-max-thread-tasks** attribute in the system configuration definition.
2. If dynamic configuration is not enabled, specify this environment variable in the **aopstart** EXEC.

**Default:** The limit that is 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 server environment variables file.

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:

1. If you start the IPP Server, set the CLASSPATH environment variable in the **aopstart** EXEC or the *envvar* file.
2. If you start Infoprint Central, set the CLASSPATH environment variable in the HTTP server environment variables file.

#### 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 that is 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 126
- Infoprint Central: “Identifying software requirements for Infoprint Central” on page 340

#### Rules:

1. IPP Server: Set the JAVA\_HOME environment variable in the **aopstart** EXEC or the *envvar* file.
2. Infoprint Central: Set the JAVA\_HOME environment variable in the HTTP server environment variables file.

**Default:** The IPP Server and Infoprint Central use Java files in this directory: `/usr/lpp/java/J7.1`.

### LANG

The language that is 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 that is used to format time and date information in messages. This locale overrides the locale in the `LC_TIME` variable.

**Rule:** In the IBM HTTP Server - Powered by Apache environment variables file (`bin/envvars`), do not specify a value that is associated with a code page other than IBM-1047.

**Examples:**

1. **aopstart** EXEC: LC\_ALL=Fr\_FR.IBM-297
2. bin/envvars file: LC\_ALL=Fr\_FR.IBM-1047

**Default:** C (also called POSIX)

**LC\_CTYPE**

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

**Rules:** In the IBM HTTP Server - Powered by Apache environment variables file (bin/envvars):

1. Do not specify the LC\_CTYPE environment variable.
2. 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 that is 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 that is 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 HTTP server environment variables file:

/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 and with the **aopdemon** command. 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 bin/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 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 and with the **aopdemon** command. You must specify this environment variable in the /etc/profile file.

**PATH** The path that is used to locate executable files. This environment variable is required. If you installed Infoprint Server executable files 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 and with the **aopdemon** command. 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 that is named in either LPDEST or PRINTER overrides the printer that is named on the ISPF configuration panel.

#### STEPLIB

The names of any step libraries that are not in the system LNKLIST, 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 that is specified by the STEPLIBLIST statement in the BPXPRMxx PARMLIB member. See “Updating the BPXPRMxx member of SYS1.PARMLIB (extended mode)” on page 215.

**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 that is used to format date and time information in messages that are displayed by **aoplogu**.

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

---

## 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 with 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 that is 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 zeros, 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:**

1. 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.
2. 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.
3. 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:**

1. 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.
2. If dynamic configuration is not enabled, specify the equivalent AOP\_ALLOW\_ALL\_CHARACTERS\_IN\_LINE\_DATA environment variable in the **aopstart** EXEC.
3. 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

#### **aopipd-max-thread-tasks = nnn**

The MAXTHREADTASKS limit for the Internet Printing Protocol (IPP) daemon, **aopipd**. The MAXTHREADTASKS limit is the maximum number of MVS tasks that **aopipd** can have concurrently active.

**Tip:** See the “Tips” for the **max-thread-tasks** attribute.

#### **Rules:**

1. If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop **aopipd** and restart Infoprint Server.
2. If dynamic configuration is not enabled, specify the equivalent AOPIPD\_MAXTHREADTASKS environment variable in the **aopstart** EXEC.

**Default:** The limit that is 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:**

1. If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop **aoplpd** and restart Infoprint Server.
2. If dynamic configuration is not enabled, specify the equivalent AOPLPD\_MAXTHREADTASKS environment variable in the **aopstart** EXEC.

**Default:** The limit that is 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:**

1. If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop **aopnetd** and restart Infoprint Server.
2. If dynamic configuration is not enabled, specify the equivalent AOPNETD\_MAXTHREADTASKS environment variable in the **aopstart** EXEC.

**Default:** The limit that is 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:**

1. 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*.
2. Also see the “Tips” for the **max-thread-tasks** attribute.

**Rules:**

1. 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.
2. If dynamic configuration is not enabled, specify the equivalent AOPOUTD\_MAXTHREADTASKS environment variable in the **aopstart** EXEC.

**Default:** The limit that is 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
```

**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:**

1. If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop **aopssid** and restart Infoprint Server.
2. If dynamic configuration is not enabled, specify the equivalent AOPSSID\_MAXTHREADTASKS environment variable in the **aopstart** EXEC.

**Default:** The limit that is 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:**

1. If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop **aopsubd** and restart Infoprint Server.
2. If dynamic configuration is not enabled, specify the equivalent AOPSUBD\_MAXTHREADTASKS environment variable in the **aopstart** EXEC.

**Default:** The limit that is 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:**

1. 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.
2. If dynamic configuration is not enabled, specify the equivalent AOPWSMD\_MAXTHREADTASKS environment variable in the **aopstart** EXEC.

**Default:** The limit that is 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 that is 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:**

1. 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.
2. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOP\_APPLID environment variable in the **aopstart** EXEC.
3. 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 that is 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 that is 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 that is 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:

1. 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.
2. 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 you restart 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.

**Value:** 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:

1. Specify this attribute only if line or text data prints with extra blank characters at the end of lines.
2. This attribute can slow performance and cause IP PrintWay to use more system resources.
3. Before JES gives 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 can 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:**

1. 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.
2. 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.
3. 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:**

1. 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.
2. Do not use these reserved names: HC, INSTREAM, INTERNAL, OPERLOG, SYSIOSRS, SYSLOG, UNKNOWN.
3. 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*.

4. 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.
5. 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 you restart Infoprint Server.

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

**Example:** console-name = PRINTCON

**ebcdic-codepage = codepage**

The name of an IBM-supplied or custom EBCDIC code page that is 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:

1. 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.
2. 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 you restart 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:**

1. 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.
2. 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 you restart 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:**

1. 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.
2. If your installation uses the MPFLSTxx Parmlib member for message automation, you might want to specify the same message IDs in MPFLSTxx.
3. If your installation uses an automated operations tool, the tool can process these messages.

**Rules:**

1. If you specify this attribute, also specify **hardcopy-messages = list**.
2. 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.
3. 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 you restart 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 that are 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:**

1. Set **ignore-dcf-routing-errors = yes** if your IP PrintWay job-selection rules cause IP PrintWay to select many 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.
2. 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.
3. 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:**

1. 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.
2. 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.
3. 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:**

1. Make sure that the port is not used by any other service on the same TCP/IP stack.
2. 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.
3. 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 you restart Infoprint Server.

**Default:** ipp-port-number = 631

**ipsmode** = **z201** | **z202**

Indicates the product function level that you want Infoprint Server to operate with.

**z201** Infoprint Server operates with z/OS 2.1 functions (default).

**z202** Infoprint Server operates with z/OS 2.2 functions.

**Rule:** You can specify this attribute only in the system configuration definition.

**Default:** ipsmode=z201

**Example:** ipsmode=z202

#### **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:**

1. 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.
2. The job ID that Print Interface returns to the job submitter does not contain this prefix.

#### **Rules:**

1. 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.
2. 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 you restart 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 are 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 that is specified in the system logger log stream definition is used instead.

You can specify a value 0 - 59. A value of 0 means that Infoprint Central cannot display any messages and cannot display information about print jobs are 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:

1. 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.
2. If you use the system logger for messages and also use Infoprint Central, the value needs to 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.
3. 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.
4. 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 34.
5. 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.
6. 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 122.
7. If you use **aoplogu** to view messages, the date and time format that is displayed in messages is controlled by the TZ and either the LC\_TIME or LC\_ALL environment variables. See “Setting environment variables for the Printer Inventory Manager” on page 46.  
  
If you use Infoprint Central to view messages, the date and time format for the messages is controlled by the operating system that runs your web browser.

#### Rules:

1. 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.
2. 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 you restart 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.

**Tip:** 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 **eng** 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:**

1. Make sure that the port is not used by any other service on the same TCP/IP stack.
2. 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 127.
3. 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.
4. 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 you restart 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 that are 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:

1. 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.
2. 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.
3. 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 that are 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:

1. Specify the suffixes that users might specify in file names. For example, you might specify the .doc suffix for Microsoft Word documents.
2. If the **mail-do-not-add-suffixes = yes**, IP PrintWay ignores the **mail-preserve-suffixes** attribute.

#### Rules:

1. Begin each suffix with a period.
2. To specify more than one suffix, separate the suffixes with a blank and enclose the suffixes in single or double quotation marks.
3. 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.

4. 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.
5. 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 that are specified for the first document in the print job.

**yes** IP PrintWay sends each document to the email address or addresses that are specified for the first document in the print job.

**no** IP PrintWay sends each document to the email address or addresses that are specified for that document.

**Rules:**

1. 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.
2. 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.
3. 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 that are 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:**

1. To specify more than one option, separate the options with a blank and enclose the options in single or double quotation marks.
2. 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.

3. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOPMAILER\_OPTIONS environment variable in the **aopstart** EXEC.
4. 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:**

1. Start the path name with a slash (/).
2. 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.
3. If you run IP PrintWay extended mode and dynamic configuration is not enabled, specify the equivalent AOPMAILER environment variable in the **aopstart** EXEC.
4. 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 that are specified in the **log-retention** attribute limits the amount of information that Infoprint Server keeps in the Historical Inventory.

Table 67 shows the number of documents 1 MB in the Historical Inventory can contain.

*Table 67. Number of documents in the Historical Inventory*

If print jobs on the JES spool contain:	1 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 1 MB

For example:

- Documents processed in one day = 100,000
- Days to retain information = 3 (**log-retention** = 3)
- Documents in 1 MB = 1350 documents

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

#### Rules:

1. 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.
2. 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 you restart 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: **aopipdd**, **aoplpd**, **aopnetd**, **aopoutd**, **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:

1. 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 is reached, threads are 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 by using the SETOMVS command.

2. Do not set a value less than 32.
3. If you increase the MAXTHREADTASKS limit, you might need to increase the maximum region size because each concurrent task requires more storage:
  - If you use the JCL startup procedure to start Infoprint Server, increase the region size on the EXEC statement in the procedure.

- If you use the start command from the z/OS UNIX command line 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). Be careful when you are using the SHRLIBRGNSIZE OMVS parameter in a 31-bit environment because it limits the size of usable regions across the system.
  - If you enter the start 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.
4. Do not set a high limit because each concurrent task requires some storage below the 16 MB line. If the limit is too high, an abend can occur.
  5. If you exceed the default limit (200) when many 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:

1. 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.
2. 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 that is 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 large number of PSF FSAs or IP PrintWay basic mode FSAs. Contact IBM for more information.

#### Rules:

1. If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop **aopnetd** and restart Infoprint Server.

2. 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:**

1. If dynamic configuration is enabled, specify this attribute in the system configuration definition. If you change this attribute, stop **aopnetd** and restart Infoprint Server.
2. 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. You can use Infoprint Central to move the held output groups to different IP PrintWay printers before you release them.

**no** IP PrintWay extended mode does not select new held output groups from the JES spool. You can use JES or SDSF commands to release individual data sets in an output group for printing.

**Rules:**

1. 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.
2. 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.
3. 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:**

1. 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.
2. 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 you restart Infoprint Server.
3. 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 that is 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:**

1. 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.
2. 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 that is returned by DLL filters.

**Rules:**

1. 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.



2. 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.
3. 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 79.
- How to edit the system configuration definition, see “Steps for enabling dynamic configuration” on page 82.
- 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 is 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 is 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 create 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 creates the process in a separate address space and none of the DD statements are visible to the process.

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 is in a z/OS UNIX file. The program name is case-sensitive. The file must be in one of the directories that are 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 that are 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 that is 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 exist, type UNNUM to remove them and set number mode off.
- Sequence numbers in columns 73 - 80 in data that is 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 can use AOPBATCH to run a shell script if the first line of the script contains *#!/script interpreter*, where *script interpreter* is the program that processes the script. For example, this is the first line in a script:  

```
#!/bin/sh
```

Otherwise, 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 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 correctly sets it.

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 can not create the process.                                   |
| <i>nn</i> | The exit value from the created 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

These examples show how to use the AOPBATCH program. The environment variables that are 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 uses 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 that are 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 465
- “Programming considerations” on page 479
- “Customization requirements” on page 480

---

### API files

Table 68 summarizes the files that are provided with the API. For each file, the table lists the default directory into which the file is installed.

*Table 68. API files*

File	Directory	Description
aopapi.dll	/usr/lpp/Printsrv/lib	Executable code
aopapi.h	/usr/lpp/Printsrv/samples	Include file that contains data structures and function declarations
libaopapi.a	/usr/lpp/Printsrv/lib	Static library that contains function stubs

---

### API data structures

Table 69 summarizes the data structures that are provided with the Infoprint Server API. After the table, each data structure is described more fully in alphabetical order.

*Table 69. API data structures*

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

Table 69. API data structures (continued)

Data structure	Include file	Description
PrinterInfo	aopapi.h	Information about an Infoprint Server printer
JobToken	aopapi.h	A token that uniquely identifies an Infoprint Server print job
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 “GetAPIVersionNumber -- return the current API version number” on page 473.

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 end 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 that is specified in the printer definition or a NULL value if no description is specified.

**location**

The location that is 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 PrinterStatus 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 that is used to send input and receive output from the transform daemon.

## API functions

Table 70 summarizes the C library functions that are provided with the API. Each function is listed according to the type of function it does. Each function is described more fully in alphabetical order.

Table 70. API functions

Type	Function	Include file	Description
Determining the API version number	GetAPIVersionNumber	aopapi.h	Returns the version number of this API
Initializing and terminating the API	InitAPI	aopapi.h	Initializes the API
	TermAPI	aopapi.h	Terminates the API
Creating a print job	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	CancelJob	aopapi.h	Cancels an Infoprint Server print job
Querying print jobs	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	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	RequestTransform	aopapi.h	Establishes a connection with a transform daemon
	ReleaseTransform	aopapi.h	Releases a connection with a transform daemon

Table 70. API functions (continued)

Type	Function	Include file	Description
Handling errors	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 is not yet closed by the ClosePrintFile function. The *token* argument must be a valid job token that is 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 nonzero 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 you call BeginEnumJobs, call these functions:

1. EnumJobs to obtain the enumeration of print jobs.
2. EndEnumJobs to end the enumeration.

Input parameters are:

*printer\_name*

The name of a printer definition in the Infoprint Server Printer Inventory whose jobs you want to enumerate. Printer definition names are case-sensitive.

*out\_token*

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 nonzero 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 that are defined by printer definitions in the Infoprint Server Printer Inventory. After you call 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 that is 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 nonzero 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 because of 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 is closed on the JES spool but is not yet selected for printing. It removes all output data sets for the job on the JES spool. Input parameters are:

*job\_id* The job ID returned by the ClosePrintFile function or the job ID returned by Infoprint Server if the job was created by some other method.

*user\_name*

The user name that is passed to the CreatePrintFile 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 nonzero 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 is called. Input parameters are:

*token* The job token that is returned by the CreatePrintFile function.

*job\_id* A job ID assigned by Infoprint Server and returned by this function. If you call the GetJobInfo or CancelJob 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 with 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 1 - 65535, define the environment variable AOP\_64K\_JOBID before you call the CreatePrintFile function.

### Returned value

ClosePrintFile returns:

- Zero and sets *job\_id* if the output data set was closed successfully.
- A nonzero 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 you call 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:

*printer\_name*

The name of a printer definition in the Infoprint Server Printer Inventory. Printer definition names are case-sensitive.

*user\_name*

The name of the owner of the print job. The name can be 1 - 31 characters. If *user\_name* is NULL, CreatePrintFile uses the current user login name that is returned by the `_getlogin1()` function. The user name becomes the job name of the output data set on the JES spool.

*file\_name*

The name of the document. The name can be any length. If *file\_name* is NULL, the job identifier is used.

*attributes*

A null-terminated string of job attributes. If *attributes* is NULL, values specified in the printer definition, or default values, are used for all job attributes. Specify attributes in the format *attribute=value*, separated by spaces. For more information about the valid job attributes and how to specify them, see the description of the `-o` option of the `lp` command and the *Using Job Attributes* section of the *z/OS Infoprint Server User's Guide*.

**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.

*out\_token*

A job token that is returned by this function. You must pass this job token as an argument to the WritePrintFile, ClosePrintFile, and AbortPrintFile functions. You can pass the returned job token to functions from threads other than the thread that called the CreatePrintFile function.



### Returned value

CreatePrintFile returns:

- Zero and *out\_token* if the print job was successfully created.
- A nonzero 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 that is returned by the `BeginEnumJobs` function.

### Returned value

EndEnumJobs returns:

- Zero if the function was successful.
- A nonzero 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 that is returned by the `BeginEnumPrinters` function.

### Returned value

EndEnumPrinters returns:

- Zero if the function was successful.
- A nonzero 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 that are 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:

*token* A valid query token that is returned by the BeginEnumJobs function.

*buffer* A buffer in which this function can return the JobInfo data structures.

*buffer\_size*

The number of bytes in the buffer.

*max\_count*

The maximum number of jobs for which you want JobInfo structures to be returned.

*job\_list*

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 is freed or modified. The returned value indicates the number of pointers in the array.

## Returned value

EnumJobs returns:

- The number of JobInfo structures that are 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 that are 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 is returned. You must call the EndEnumPrinters function to end the enumeration of printers.

Input parameters are:

*token* A valid query token that is returned by the BeginEnumPrinters function.

*buffer* A buffer in which this function can return the PrinterInfo data structures.

*buffer\_size*

The number of bytes in the buffer.

*max\_count*

The maximum number of printers for which you want PrinterInfo structures to be returned.

*printer\_list*

A pointer that is 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 is freed or modified. The returned value indicates the number of pointers in the array.

## Returned value

EnumPrinters returns:

- The number of PrinterInfo structures that are created if the function was successful. Zero indicates that information about all printers is 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 that is returned by the API for the calling thread. Table 71 lists names that are defined in the aopapi.h file for the error numbers and describes the error. The table does not list all possible error numbers that are returned by this function. For all errors, you can use the ErrorString function to obtain a description of the error.

Table 71. ErrorNumber names and error descriptions

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.

Table 71. *ErrorNumber names and error descriptions (continued)*

Name	Description
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.
AOP_BUFFER_OVERFLOW	The buffer that is provided to an API function was not large 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 cannot be initialized. Make sure that <code>aopapi.dll</code> can be loaded from one of the directories that are 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 that is 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.

## GetAPIVersionNumber -- return the current API version number

### Format

```
#include <aopapi.h>
int GetAPIVersionNumber();
```

### Description

The `GetAPIVersionNumber` function returns the current version number of the API. Use this function to determine what fields and functions are available in the API based on the version number that is returned.

### Returned value

`GetAPIVersionNumber` returns:

	0	API Version 0
	1	API Version 1
		In Version 1, the DocumentInfo data structure contains the high_size field.
		This field is not present in API Version 0. See "DocumentInfo - information
		about a document in a job" on page 462.

## 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 that is 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:

*job\_id* The job ID returned by the ClosePrintFile function or the job ID returned by Infoprint Server if the job was created by some other method.

*buffer* A buffer in which this function returns the JobInfo data structure.

*buffer\_size*  
The number of bytes in the buffer.

### Returned value

GetJobInfo returns:

- A pointer to a JobInfo structure if the function was successful. The address is in the buffer that is 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 that is 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:

*printer\_name*

The name of the printer definition in the Infoprint Server Printer Inventory.

*buffer* A buffer in which this function returns the PrinterInfo data structure.

*buffer\_size*

The number of bytes in the buffer.

### Returned value

GetPrinterInfo returns:

- A pointer to a PrinterInfo structure if the function was successful. The address is in the buffer that is 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 is not called explicitly. Use this function if your application needs to determine whether the API can be successfully initialized separately from other calls.

### Returned value

InitAPI returns:

- Zero if the API was successfully initialized.
- A nonzero 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; for example, because of an unexpected transform failure or a communication failure.

### Returned value

ReleaseTransform returns:

- Zero if the function was successful.



- A nonzero 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 that is 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. For information about the transform configuration file, see “Format of an entry in the transform configuration file” on page 169.

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 “Programming considerations” on page 479.

### Returned value

`SetTerminationHandler` returns:

- Zero if the function was successful.

- A nonzero 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 that are 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 nonzero 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 it writes 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 it writes 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 is 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 that is associated with *token* is ended and the output data set is removed from the JES spool as if the `AbortPrintFile` function is called.

Input parameters are:

*token* A valid job token that is returned by the CreatePrintFile function.

*buffer* A buffer that contains the data to be written to the output data set on the JES spool.

*buffer\_size*  
The number of bytes in the buffer.

### Returned value

WritePrintFile returns:

- Zero if the data is written successfully.
- A nonzero value if data is not successfully written.

---

## Programming considerations

These are the programming 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 (by 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 you include the API header file (aopapi.h). For example, you can 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 clean up 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:

*handler\_argument*

The argument that was passed to the SetTerminationHandler function.

*cleanup\_handler*

A pointer to a cleanup handler function that your termination handler must call.

*cleanup\_argument*

An argument that the termination handler must pass to the cleanup handler.

Your termination handler function must do these functions:

- Produce a dump or equivalent diagnostic of the process to help diagnose problems in your application and the API.
- Do any wanted cleanup actions. All actions that are permitted by a 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 call the `exit` function to terminate the process or 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 suboption 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 125.
- (Optional) Customize the Transform Interface and a transform product. See Chapter 5, “Customizing Transform Interface,” on page 167.

In addition, do these customization tasks that are related to the API:

- (Optional) Define environment variable `A0P_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 must not define this variable because system performance might be degraded.

After customizing Infoprint Server, do these tasks that are 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 207.

### 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 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 14 on page 81. For information about how to edit the system configuration definition, see “Steps for editing the system configuration definition” on page 91.

- Increase the space available to the `/var/Printsrv` directory if necessary. See “Increasing the space available in `/var/Printsrv` directory (extended mode)” on page 215.
- 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 215.
- 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 216.
- 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 222.
- 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 223.
- Edit the **aopstart** EXEC or `envvar` file 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 `LNKLST`. 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 224.

- Recompile the IP PrintWay SMF exit.
  - Review other IP PrintWay exits to see whether changes are required. See “Comparing IP PrintWay basic mode and extended mode exits” on page 248.
  - 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 `ANFUEXT` control block instead of in records. See “Begin Data Set exit” on page 267.
2. Customize Infoprint Central. See Chapter 8, “Customizing Infoprint Central,” on page 339.
  3. Restart the IBM HTTP Server - Powered by Apache 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 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 that is supplied by JES as 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 whether 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 can select output data sets in class Q. (See “Creating job selection rules in the Printer Inventory (extended mode)” on page 220.)
- 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 245.)

---

## Appendix H. Benefiting from new functions in z/OS V2R2

This section describes how to benefit from the Infoprint Server functions that are introduced in z/OS V2R2.

**Note:** All installations must perform some actions when upgrading to z/OS V2R2. For information about the actions that installations must take to maintain the same functional behavior in z/OS V2R2 as in previous releases, see *z/OS Migration*.

---

### Infoprint Central enhancements

The Infoprint Central user interface is updated with these enhancements:

- The user interface is redesigned to follow new IBM accessibility and usability standards, and now includes support for Mozilla Firefox.
- The IBM HTTP Server powered by Domino that Infoprint Central previously used is replaced with the IBM HTTP Server - Powered by Apache. The IBM HTTP Server - Powered by Apache is the strategic HTTP server that z/OS V2R2 supports and includes security capabilities that are similar to the ones provided in the Domino HTTP Server.
- You can now display TSO output jobs in the list of JES print jobs from a search. This function is available when the **Operating mode** field is set to z/OS 2.2 on the ISPF System Configuration panel.
- The System Status page now displays a table with the component, XCF member name, job name, system, and state when a new operating mode data element is set to z/OS Version 2 Release 2.
- Job selection rules are now accessed from the menu bar rather than the System Status page.

---

### Serviceability enhancements

To minimize migration actions when a new z/OS release is installed, Infoprint Server now supports a data element that you can use to activate new product functions during or after installation. This data element is set with the **Operating mode** field on the ISPF System Configuration panel.

---

### Start and stop Infoprint Server daemons

Infoprint Server runs as multiple daemons on a single processor with a master daemon that starts and stops the other daemons as necessary. Infoprint Server now gives you the ability to start and stop individual daemons on a single processor when the **Operating mode** field is set to z/OS 2.2 on the ISPF System Configuration panel.

---

### TSO/E command for controlling printers

Infoprint Central was previously the primary method that is used to start and stop Infoprint Server printers. Infoprint Server now supports a TSO/E command, AOPCMND, that can start and stop IP PrintWay extended mode printers from TSO.

Some advantages of using the TSO/E AOPCMND command are:

- You can control IP PrintWay extended mode printers without implementing the Infoprint Central component that runs under the IBM HTTP Server - Powered by Apache.
- You can migrate from a non-IBM print application to Infoprint Server.

---

## Variable inline email message

You can now use Infoprint Server to customize email messages with personalized inline text.

Some advantages of using a variable inline email message are:

- By using an existing text job attribute that corresponds to a JCL keyword, such as **name-text**, you can include a text string at the beginning of each email before an inline message. For example, if `name-text="Mr. Jones"`, Mr. Jones is included on the first line in the email.
- You can also now specify the existing **mail-inline-message** printer attribute as a job attribute so you can send a unique message for the job.

---

## Appendix I. Accessibility

Accessible publications for this product are offered through IBM Knowledge Center (<http://www.ibm.com/support/knowledgecenter/SSLTBW/welcome>).

If you experience difficulty with the accessibility of any z/OS information, send a detailed message to the "Contact us" web page for z/OS (<http://www.ibm.com/systems/z/os/zos/webqs.html>) or use the following mailing address.

IBM Corporation  
Attention: MHVRCFS Reader Comments  
Department H6MA, Building 707  
2455 South Road  
Poughkeepsie, NY 12601-5400  
United States

---

### Accessibility features

Accessibility features help users who have physical disabilities such as restricted mobility or limited vision use software products successfully. The accessibility features in z/OS can help users do the following tasks:

- Run assistive technology such as screen readers and screen magnifier software.
- Operate specific or equivalent features by using the keyboard.
- Customize display attributes such as color, contrast, and font size.

---

### Consult assistive technologies

Assistive technology products such as screen readers function with the user interfaces found in z/OS. Consult the product information for the specific assistive technology product that is used to access z/OS interfaces.

---

### Keyboard navigation of the user interface

You can access z/OS user interfaces with TSO/E or ISPF. The following information describes how to use TSO/E and ISPF, including the use of keyboard shortcuts and function keys (PF keys). Each guide includes the default settings for the PF keys.

- *z/OS TSO/E Primer*
- *z/OS TSO/E User's Guide*
- *z/OS V2R2 ISPF User's Guide Vol I*

---

### Dotted decimal syntax diagrams

Syntax diagrams are provided in dotted decimal format for users who access IBM Knowledge Center with a screen reader. In dotted decimal format, each syntax element is written on a separate line. If two or more syntax elements are always present together (or always absent together), they can appear on the same line because they are considered a single compound syntax element.

Each line starts with a dotted decimal number; for example, 3 or 3.1 or 3.1.1. To hear these numbers correctly, make sure that the screen reader is set to read out

punctuation. All the syntax elements that have the same dotted decimal number (for example, all the syntax elements that have the number 3.1) are mutually exclusive alternatives. If you hear the lines 3.1 USERID and 3.1 SYSTEMID, your syntax can include either USERID or SYSTEMID, but not both.

The dotted decimal numbering level denotes the level of nesting. For example, if a syntax element with dotted decimal number 3 is followed by a series of syntax elements with dotted decimal number 3.1, all the syntax elements numbered 3.1 are subordinate to the syntax element numbered 3.

Certain words and symbols are used next to the dotted decimal numbers to add information about the syntax elements. Occasionally, these words and symbols might occur at the beginning of the element itself. For ease of identification, if the word or symbol is a part of the syntax element, it is preceded by the backslash (\) character. The \* symbol is placed next to a dotted decimal number to indicate that the syntax element repeats. For example, syntax element \*FILE with dotted decimal number 3 is given the format 3 \\* FILE. Format 3\* FILE indicates that syntax element FILE repeats. Format 3\* \\* FILE indicates that syntax element \* FILE repeats.

Characters such as commas, which are used to separate a string of syntax elements, are shown in the syntax just before the items they separate. These characters can appear on the same line as each item, or on a separate line with the same dotted decimal number as the relevant items. The line can also show another symbol to provide information about the syntax elements. For example, the lines 5.1\*, 5.1 LASTRUN, and 5.1 DELETE mean that if you use more than one of the LASTRUN and DELETE syntax elements, the elements must be separated by a comma. If no separator is given, assume that you use a blank to separate each syntax element.

If a syntax element is preceded by the % symbol, it indicates a reference that is defined elsewhere. The string that follows the % symbol is the name of a syntax fragment rather than a literal. For example, the line 2.1 %OP1 means that you must refer to separate syntax fragment OP1.

The following symbols are used next to the dotted decimal numbers.

**? indicates an optional syntax element**

The question mark (?) symbol indicates an optional syntax element. A dotted decimal number followed by the question mark symbol (?) indicates that all the syntax elements with a corresponding dotted decimal number, and any subordinate syntax elements, are optional. If there is only one syntax element with a dotted decimal number, the ? symbol is displayed on the same line as the syntax element, (for example 5? NOTIFY). If there is more than one syntax element with a dotted decimal number, the ? symbol is displayed on a line by itself, followed by the syntax elements that are optional. For example, if you hear the lines 5 ?, 5 NOTIFY, and 5 UPDATE, you know that the syntax elements NOTIFY and UPDATE are optional. That is, you can choose one or none of them. The ? symbol is equivalent to a bypass line in a railroad diagram.

**! indicates a default syntax element**

The exclamation mark (!) symbol indicates a default syntax element. A dotted decimal number followed by the ! symbol and a syntax element indicate that the syntax element is the default option for all syntax elements that share the same dotted decimal number. Only one of the syntax elements that share the dotted decimal number can specify the ! symbol. For example, if you hear the lines 2? FILE, 2.1! (KEEP), and 2.1 (DELETE), you know that (KEEP) is the



default option for the FILE keyword. In the example, if you include the FILE keyword, but do not specify an option, the default option KEEP is applied. A default option also applies to the next higher dotted decimal number. In this example, if the FILE keyword is omitted, the default FILE(KEEP) is used. However, if you hear the lines 2? FILE, 2.1, 2.1.1! (KEEP), and 2.1.1 (DELETE), the default option KEEP applies only to the next higher dotted decimal number, 2.1 (which does not have an associated keyword), and does not apply to 2? FILE. Nothing is used if the keyword FILE is omitted.

**\* indicates an optional syntax element that is repeatable**

The asterisk or glyph (\*) symbol indicates a syntax element that can be repeated zero or more times. A dotted decimal number followed by the \* symbol indicates that this syntax element can be used zero or more times; that is, it is optional and can be repeated. For example, if you hear the line 5.1\* data area, you know that you can include one data area, more than one data area, or no data area. If you hear the lines 3\* , 3 HOST, 3 STATE, you know that you can include HOST, STATE, both together, or nothing.

**Notes:**

1. If a dotted decimal number has an asterisk (\*) next to it and there is only one item with that dotted decimal number, you can repeat that same item more than once.
2. If a dotted decimal number has an asterisk next to it and several items have that dotted decimal number, you can use more than one item from the list, but you cannot use the items more than once each. In the previous example, you can write HOST STATE, but you cannot write HOST HOST.
3. The \* symbol is equivalent to a loopback line in a railroad syntax diagram.

**+ indicates a syntax element that must be included**

The plus (+) symbol indicates a syntax element that must be included at least once. A dotted decimal number followed by the + symbol indicates that the syntax element must be included one or more times. That is, it must be included at least once and can be repeated. For example, if you hear the line 6.1+ data area, you must include at least one data area. If you hear the lines 2+, 2 HOST, and 2 STATE, you know that you must include HOST, STATE, or both. Similar to the \* symbol, the + symbol can repeat a particular item if it is the only item with that dotted decimal number. The + symbol, like the \* symbol, is equivalent to a loopback line in a railroad syntax diagram.



---

## Notices

This information was developed for products and services 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.

#### COPYRIGHT LICENSE:

This information might contain sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

---

## 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: IBM Lifecycle Support for z/OS (<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 Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available at Copyright and Trademark information (<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.

Intel and Itanium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and 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 that are used in Infoprint Server documentation. If you do not find the term that you are looking for, see the index or see the IBM Glossary of Computing Terms (<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.

**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** A UNIX operating system developed by IBM that is designed and optimized to run on POWER<sup>®</sup> 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 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 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, and PSF. 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 retrieve 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.

**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 defines an aspect of the operating environment for a process. For example, environment variables can define the home directory, the command search path, the terminal in use, or 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 execute in a hiperspace. Unlike a

data space, data in a hyperspace 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 that 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. The host name might be a fully qualified domain name such as mycomputer.city.company.com, or it might be a specific subname such as 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.

**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.

**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 JES3, Job Entry Subsystem.

**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 JES2, Job Entry Subsystem.

**job control language (JCL)**

A command language that identifies a job to an operating system and describes the job requirements.

**Job Entry Subsystem (JES)**

An IBM licensed program that receives jobs into the system and processes all output data that is produced by those 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.

**KB** See kilobyte.

**kilobyte (KB)**

For processor storage, real and virtual storage, and channel volume, 2 to the power of 10 or 1,024 bytes. For disk storage capacity and communications volume, 1,000 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** See megabyte.

**megabyte (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.

## **migrate**

To install a new version or release of a program to replace an earlier version or release.

## **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 Printer Manager (NPM)**

An application that lets network administrators monitor, control, and configure IBM 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.

## **Network Print Facility (NPF)**

A feature that routes VTAM, JES2, or JES3 printer output to printers in a TCP/IP network.

**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.

**pel** See picture element.

## **picture element (pel, 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.

## **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.
- 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.
- 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 attempts.
- 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 based on their successive physical positions, such as on magnetic tape. See also partitioned data set.

**Server Message Block (SMB)**

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, software programs, or devices on a network that allows the two elements to communicate and exchange data for the duration of the session.

**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.

**SMB** See Server Message Block.

**SMF** See System Management Facilities.

**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.

**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 Facilities (SMF)**

A component of z/OS 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. Telnet enables 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 transmits to the remote system or that IP PrintWay retains on the Job Entry Subsystem (JES) spool.

**transparent data**

1. 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.
2. Data that can contain any hexadecimal value.

**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** A highly portable operating system that features multiprogramming in a multiuser environment. The UNIX operating system was originally developed for use on minicomputers, but was adapted for mainframes and microcomputers. The AIX operating system is IBM's implementation of the UNIX operating system.

**UNIX file**

An object that exists in a hierarchical file system. Examples of UNIX files are 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 IBM mainframe operating system that uses 64-bit real storage.

**z/OS file system (zFS)**

A type of file system that resides in a Virtual Storage Access Method (VSAM) linear data set (LDS) and has a hierarchical organization of files and directories with a root directory.



---

# Index

## Special characters

- \_BPX\_JOBNAME environment variable 170
- \_BPX\_SHAREAS environment variable 415
- \_BPX\_UNLIMITED\_OUTPUT environment variable 48, 415
- \_BPXK\_SETIBMOPT\_TRANSPORT environment variable 144, 224, 415
- \_ICONV\_MODE environment variable 416
- /etc/Printsrv directory 33
- /etc/Printsrv/aopd.conf, creating 37
- /etc/Printsrv/aopmsg.conf, creating 96
- /etc/Printsrv/aopsapd.conf, creating 157
- /etc/profile, setting environment variables in 46
- /etc/rc file
  - setting STEPLIB variable 31
  - starting Infoprint Server daemons 75
  - starting sendmail 244
- /usr/lpp/Printsrv/bin, specifying in PATH variable 51, 435
- /var/Printsrv directory 33
- % operator in filter options 132
- & in configuration files 45, 46, 158, 406, 409, 411
- # in configuration files 45, 46, 158, 406, 409, 411

## Numerics

- 161 and 162 ports, unrestricting in TCP/IP data set 128
- 3270 data stream support in NetSpool 15
- 3270 data streams, NetSpool exits for 189
- 33E abend 61
- 515 port number
  - default 145
  - specified as default 146
  - unrestricting in TCP/IP data set 128
- 631 port number
  - specified as default 152, 153
  - unrestricting in TCP/IP data set 128

## A

- abend D37 182, 331
- AbortPrintFile function 466
- accessibility 489
  - contact IBM 489
  - features 489
- address spaces, running NetSpool in multiple 183
- administrator group, for security 55
- AFP Download Plus
  - Printer Inventory 5, 385
- AFP Printer Driver for Windows, overview 6
- AFP to PCL transform, customizing for PJJ functions 240
- AFP to PostScript transform, customizing for PJJ functions 240
- AFP Viewer plug-in for Windows, overview 7
- AIX, increasing timeout value 148
- aliases file, sendmail 244
- allow-all-attributes-in-line-data attribute 138, 203, 232
- ANFDEAL sample JCL 330
- ANFMFILE program 182, 331
- ANFMIJCL sample JCL 331
- ANFMMSG DD statement 336
- ANFQINIT sample JCL 331

- ANFQUEUE DD statement 336
- ANFUETBL, tables for block letters 310
- ANFUEXTP control block 257, 269, 280, 288, 297, 302
- ANFULINK sample JCL 312
- ANFUXRSP exit 304
- ANFUXxxx exits, sample 312
- ANFWPROC startup procedure 333
- AOP\_64K\_JOBID environment variable 416, 480
- AOP\_ALLOW\_ALL\_CHARACTERS\_IN\_LINE\_DATA environment variable 137, 202, 225, 420
- AOP\_APPLID environment variable 225, 420
- AOP\_BLANK\_TRUNCATION\_CLASSES environment variable 225
- AOP\_IGNORE\_DCF\_ROUTING\_ERRORS environment variable 219, 225, 420
- AOP\_MAIL\_DO\_NOT\_ADD\_SUFFIX environment variable 225, 420
- AOP\_MAIL\_PRESERVE\_SUFFIXES environment variable 225, 420
- AOP\_MAIL\_USE\_FIRST\_ADDRESS environment variable 225, 420
- AOP\_MAXTHREADTASKS environment variable 420
- AOP\_MVS\_RETURN\_CODES environment variable 422
- AOP\_PJL environment variable 240
- AOP\_READ\_COMMUNITY environment variable 349
- AOP\_SELECT\_WORK\_FROM\_HOLD\_QUEUE environment variable 225
- AOP\_SUPPRESS\_POST\_UNIX\_FILTER\_FORMATTING environment variable 135, 225
- AOP\_WRITE\_COMMUNITY environment variable 349
- AOP.ADMINISTRATOR profile 53, 56, 355
- AOP.CONFIGURATION profile 54, 56
- AOP.DAEMON profile 373
- AOP.NETSPOOL profile 364
- AOP.PRIORITY profile 372
- AOP192E 120
- AOP3201E, suppressing 219, 225
- AOPADMIN group 54, 55, 59
- AOPAPI 461
- AOPBATCH 457
  - examples 460
  - exit values 460
  - format 457
- AOPCENTRAL environment variable 349
- AOPCENTRAL\_CODEPAGE environment variable 349
- AOPCMND, customizing 337
- AOPCONF environment variable 48, 349
- AOPCONF group 54, 55
- AOPCPETC sample job 44, 96, 172, 176
- aopd.conf
  - attributes 395
  - creating 37
  - editing 45
  - naming in AOPCONF environment variable 48
- AOPDEMON procedure 67
- aopfilt.so 137
- AOPINIT EXEC 77
- AOPIPPD\_MAXTHREADTASKS environment variable 425
- aopipdd-max-thread-tasks 438
- AOPLIMIT environment variable 349, 426
- AOPLIMIT\_DOCS environment variable 349, 426

- AOPLIMIT\_JOBS environment variable 350, 426
- AOPLPD\_MAXTHREADTASKS environment variable 426
- aoplpd-max-thread-tasks 438
- AOPMAILER environment variable 225, 428
- AOPMAILER\_OPTIONS environment variable 225, 428
- AOPMSG\_CONF environment variable 49, 428
- aopmsg.conf
  - creating 96
  - format 406
- AOPNETD\_MAXTHREADTASKS environment variable 428
- AOPNETD\_USE\_FIXED\_JOBID environment variable 428
- AOPNETD\_USE\_UNALTERED\_JOBID environment variable 428
- aopnetd-max-thread-tasks 438
- aopoms command 160
- AOOPER group 54, 55, 59
- AOOPTIONS environment variable 140, 429
- AOPOUTD\_MAXTHREADTASKS environment variable 429
- aopoutd-max-thread-tasks 438
- AOPPATH environment variable 141, 430
- AOPPRINT JCL procedure, customizing 141
- AOPRACF 55, 65, 66
- AOPRESUBMITUSER environment variable 431
- AOPRXF\_CONF environment variable 49, 177, 431
- aoprxf.conf
  - creating 175
  - format 175, 408
- aoprxf.so 137
- AOPSAPD\_CONF environment variable 159, 431
- aopsapd.conf
  - client attribute 158, 410
  - creating 157
  - example 158
  - format 157, 410
  - initial-target attribute 158, 410
  - language attribute 158, 410
  - naming in environment variable 159
  - password attribute 158, 410
  - syntax rules 158
  - system attribute 158, 410
  - userid attribute 158, 410
- aopsetup shell script 59
- AOPSSID\_MAXTHREADTASKS environment variable 431
- aopssid-max-thread-tasks 439
- aopssid, starting 341
- AOPSTAR2 procedure 67
- aopstart EXEC
  - editing 62
  - setting environment variables in 46
- AOPSTART procedure 67
- AOPSTAT procedure 76
- AOPSTOP procedure 72
- AOPSTOP2 procedure 72
- AOPSUBD\_MAXTHREADTASKS environment variable 431
- aopsubd-max-thread-tasks 439
- AOPTRACEDIR environment variable 49
- AOPTRACEON environment variable, setting in AOPSTART procedure 67
- AOPVALIDATEDB environment variable
  - setting for Printer Inventory Manager 49
  - setting in AOPSTART procedure 67
- AOPWSMD\_MAXTHREADTASKS environment variable 432
- aopwsmd-max-thread-tasks 440
- AOPXFD\_CONF environment variable 49, 177, 432
- aopxfd.conf
  - creating 169
  - environment attribute 170

- aopxfd.conf (*continued*)
  - example 171
  - format 169, 409
  - maximum-active attribute 171
  - maximum-idle-time attribute 171
  - minimum-active attribute 171
  - start-command attribute 170
  - start-daemons attribute 168
  - transform attribute 170
- APF-authorizing C++ applications that use API 479
- APF-authorizing NetSpool exits 190
- APF-authorizing SYS1.IMAGELIB 240, 314
- API 461
- APIJPJCL startup procedure for NetSpool 186
- APIJP1D1 201
- APIJP1D2 201
- APIM1JCL sample JCL 182
- APIMSG DD statement 187
- APIPP1X2 control block 191
- APIPP1X2 macro 201
- APIPP1D1 exit, NetSpool 192
- APIPP1D2 exit, NetSpool 197
- APIUBF3 exit, NetSpool 194
- APIUGEX exit, NetSpool 199
- APPL ID, specifying for IP PrintWay basic mode 325
- APPL statement for IP PrintWay, creating 245
- Application Programming Interface 461
- applid attribute 232, 440
- ascii-codepage attribute 40, 87, 440
- assistive technologies 489
- attributes, job, specifying in environment variable 140
- AUDIT parameter, RACF 373
- AUTH parameter 246
- authorizing users to Printer Inventory, common message log, and operator commands 52
- AUTOLOG statement in TCP/IP 128
- automation, message 94
- AUTOSTOP parameter (JES2) 319

## B

- backend option, AIX 149
- backing up the Printer Inventory 114
- banner page
  - adding in exit 267, 287
  - controlling in sample exit 312
- base-directory attribute 37
- Begin Data Set exit, IP PrintWay
  - examples 278
  - input 268
  - installing 312
  - modifying 313
  - output 275
  - overview 267
  - programming considerations 251
  - sample 312
- BeginEnumJobs function 466
- BeginEnumPrinters function 467
- Beginning of File exit for 3270 data, NetSpool (APIPUBF3)
  - description 194
  - examples 196
  - input 195
  - installing 201
  - modifying 202
  - output 196
- Beginning of File exit for SCS data, NetSpool (APIPP1D1)
  - description 192

Beginning of File exit for SCS data, NetSpool (APIPPTD1)

(continued)

examples 194

input 192

installing 201

modifying 202

output 193

programming considerations 190

sample 201

benefiting from new functions 487

bin/envvars file, customizing 346

binary data support in NetSpool 16

blank truncation 137

IP PrintWay extended mode 225

blank-truncation-classes attribute 232, 440

BLNKTRNC parameter (OUTCLASS) 137

Block Letter routine

description of 310

input 311

link-editing with exit 312

output 311

sample exit for 312

tables (ANFUBTBL) 310

BPX.SMF profile 223

BPXPRMxx member, updating

for IP PrintWay basic mode 313

for IPP Server 151

for NetSpool 180

for Print Interface 129

MAXPROCUSER statement for IP PrintWay 215

MAXSOCKETS parameter 28

STEPLIBLIST facility, for SCEERUN and SCLBDLL  
libraries 31

broadcasting data 16

browsers, customizing for Infoprint Central 376

BufferSizeRequired function 467

## C

C++ runtime options

changing in IP PrintWay 337

changing in NetSpool 189

callable services 461

CancelJob function 468

carriage control characters

input to Begin Data Set exit 271

input to End Data Set exit 291

input to Routing exit 261

CARRIAGE parameter 317

cfilter.c 133

checklist of customization tasks 389

CICS, customization required to print from 25

class

changing in exit 256

CLASS parameter (JES2) 319

QUEUE parameter (JES2) 319

specifying as JES2 work-selection criteria 319, 321

specifying as JES3 work-selection criteria 324

WC parameter (JES3) 324

classes for transforms 169

CLASSPATH environment variable 49, 152, 350, 432

client attribute 158, 410

ClosePrintFile function 468

Coaxial Printer Support 213

code pages

custom code pages, specifying in configuration file 40, 87

default ASCII code page in configuration file 40, 87

code pages (continued)

default EBCDIC code page in configuration file 40, 87

defaults for IP PrintWay basic mode 325

for Infoprint Central 349

Japanese ISPF panels 77

coexistence, regarding Printer Inventory 27

COMMNDxx parmlib member 75

common message log

allocating space 36

MVS system logger 98

NetSpool 182

overview 5

reducing size of 122

saving 120

security for 52

community name for Infoprint Central 349

compatibility filter 147

conf/httpd.conf, editing 345

configuration file aopd.conf

attributes 395

creating 37

editing 45

naming in AOPCONF environment variable 48

configuration file aopmsg.conf

creating 96

format 406

configuration file aoprxf.conf

creating 175

format 175, 408

configuration file aopsapd.conf

client attribute 158, 410

creating 157

example 158

format 157, 410

initial-target attribute 158, 410

language attribute 158, 410

naming in environment variable 159

password attribute 158, 410

syntax rules 158

system attribute 158, 410

userid attribute 158, 410

configuration file aopxfd.conf

creating 169

environment attribute 170

example 171

format 169, 409

maximum-active attribute 171

maximum-idle-time attribute 171

minimum-active attribute 171

start-command attribute 170

start-daemons attribute 168

transform attribute 170

CONSNAM parameter, for NetSpool 187

console-name attribute 341, 374, 440

consoles, for NetSpool 184, 187

CONSTD statement 378

contact

z/OS 489

control file first, sending with LPR 148

COPIES parameter, changing in exit 256

cp command, example of 157

CreatePrintFile function 469

customizing

AOPPRINT JCL procedure 141

Infoprint Central 339

Infoprint Port Monitor for Windows 381

IP PrintWay 207

customizing (*continued*)

- LPD 142
- NetSpool 179
- Print Interface 125
- Print Interface IPP Server 150
- Print Interface SAP R/3 OMS 155
- Printer Inventory Manager 27
- PSF 385
- Transform Interface 167
- Transform Manager 167
- Windows client 381

## D

- D37 abend 182, 331
- daemons to start 37, 145, 151, 164, 168, 181, 213
- DATASETPREFIX statement 242
- DBCS, customizing for
  - default ASCII code page, specifying 40, 87
  - default EBCDIC code page, specifying 40, 87
- default printer, specifying 141, 435
- destination
  - changing in exit 256
  - DGROUP parameter (JES3) 322
  - ROUTECD parameter (JES2) 320
  - specifying as JES2 work-selection criteria 321
  - specifying as JES3 work-selection criteria 324
- DEVICE statement (JES3) 322
- DGROUP parameter (JES3) 322
- diagram of system flow 8
- directories for Infoprint Server 32
- directory name for files, specifying in configuration file 37
- DLLs, specifying directory for 50, 159, 434
- DNS
  - defining printers 240
  - email protocol 243
  - lookup order 242
  - resolve-printway-printers attribute 214
- document code page, changing default 40, 87
- document header
  - Begin data set exit 267
  - example 278
  - XTP\_DOC\_HEADER\_LENGTH field 275
  - XTP\_DOC\_HEADER\_PTR field 274
  - XTP\_TRANS\_DOC\_HEADER field 272
- document trailer
  - Begin data set exit 267
  - example 278
  - XTP\_DOC\_TRAILER\_LENGTH field 275
  - XTP\_DOC\_TRAILER\_PTR field 274
  - XTP\_TRANS\_DOC\_TRAILER field 272
- DocumentInfo data structure 462
- domain name server (DNS)
  - defining printers 240
  - email protocol 243
  - lookup order 242
  - resolve-printway-printers attribute 214
- downloading Infoprint Port Monitor 381
- DTYPE parameter (JES3) 322
- dynamic attributes 79
- dynamic configuration, enabling 78
- dynamic enablement of Infoprint Server 28
- DYNAMIC parameter (JES3) 322
- dynamic-configuration attribute 37

## E

- ebcdic-codepage attribute 40, 87, 440
- EC6 abend 61
- echo command, example of 47
- email addresses for multi-document print jobs 228, 420
- email protocol, IP PrintWay
  - changing email address in Routing exit 267
  - customization tasks specific to protocol 209
  - environment variables 336
  - sendmail customization 243
- enabling Infoprint Server 28
- End Data Set exit, IP PrintWay
  - examples 295
  - input 288
  - installing 312
  - modifying 313
  - output 294
  - overview 287
  - programming considerations 251
  - sample 312
- EndEnumJobs function 470
- EndEnumPrinters function 470
- EnumJobs function 470
- EnumPrinters function 471
- envvar file
  - creating 65
  - setting environment variables in 46
- environment attribute 170
- environment variables
  - \_BPX\_JOBNAME 170
  - \_BPX\_SHAREAS variable 415
  - \_BPX\_UNLIMITED\_OUTPUT variable 48, 415
  - \_BPXK\_SETIBMOPTR\_TRANSPORT variable 144, 224, 415
  - \_ICONV\_MODE variable 416
  - AOP\_64K\_JOBID variable 416, 480
  - AOP\_ALLOW\_ALL\_CHARACTERS\_IN\_LINE\_DATA variable 137, 202, 225, 420
  - AOP\_APPLID variable 225, 420
  - AOP\_BLANK\_TRUNCATION\_CLASSES variable 225
  - AOP\_IGNORE\_DCF\_ROUTING\_ERRORS variable 225, 420
  - AOP\_MAIL\_DO\_NOT\_ADD\_SUFFIX variable 225, 420
  - AOP\_MAIL\_PRESERVE\_SUFFIXES variable 225, 420
  - AOP\_MAIL\_USE\_FIRST\_ADDRESS variable 225, 420
  - AOP\_MAXTHREADTASKS variable 420
  - AOP\_MVS\_RETURN\_CODES variable 422
  - AOP\_READ\_COMMUNITY variable 349
  - AOP\_SELECT\_WORK\_FROM\_HOLD\_QUEUE variable 225
  - AOP\_SUPPRESS\_POST\_UNIX\_FILTER\_FORMATTING variable 225
  - AOP\_WRITE\_COMMUNITY variable 349
  - AOPCENTRAL variable 349
  - AOPCENTRAL\_CODEPAGE variable 349
  - AOPCONF variable 48, 349
  - AOPIPPD\_MAXTHREADTASKS variable 425
  - AOPLIMIT variable 349, 426
  - AOPLIMIT\_DOCS variable 349, 426
  - AOPLIMIT\_JOBS variable 350, 426
  - AOPLPD\_MAXTHREADTASKS variable 426
  - AOPMAILER variable 225, 428
  - AOPMAILER\_OPTIONS variable 225, 428
  - AOPMSG\_CONF variable 49, 428
  - AOPNETD\_MAXTHREADTASKS variable 428
  - AOPNETD\_USE\_FIXED\_JOBID variable 428
  - AOPNETD\_USE\_UNALTERED\_JOBID variable 428
  - AOPOPTIONS variable 140, 429



environment variables (*continued*)

- AOPOUTD\_MAXTHREADTASKS variable 429
- AOPPATH variable 141, 430
- AOPRESUBMITUSER variable 431
- AOPRXF\_CONF variable 49, 177, 431
- AOPSAPD\_CONF variable 159, 431
- AOPSSID\_MAXTHREADTASKS variable 431
- AOPSUBD\_MAXTHREADTASKS variable 431
- AOPTRACEDIR 49
- AOPVALIDATEDB variable 49
- AOPWSMD\_MAXTHREADTASKS variable 432
- AOPXFD\_CONF variable 49, 177, 432
- CLASSPATH variable 49, 152, 350, 432
- dynamic attributes, equivalent 81
- envar file, creating 65
- example of displaying with echo command 47
- examples of setting with export command 47
- for IP PrintWay 336
- for IP PrintWay extended mode 224
- for IPP Server 152
- for NetSpool 202
- for Print Interface 137
- for Print Interface IPP Server 152
- for Printer Inventory Manager 46
- for SAP OMS 159
- for Transform Interface 176
- for transforms 170
- JAVA\_HOME variable 50, 152, 350, 433
- LANG variable 50, 351, 434
- LC\_ALL variable 50, 434
- LC\_CTYPE variable 50, 434
- LC\_TIME variable 50, 434
- LIBPATH variable 50, 159, 351, 434
- LPDEST variable 141, 435
- MANPATH variable 51, 434
- NLSPATH variable 51, 351, 435
- PATH variable 51, 435
- PRINTER variable 141, 435
- setting in aopstart EXEC 62
- STEPLIB variable 31, 224, 435
- summary of 413
- TZ variable 52, 437

ErrorNumber function 472

ErrorString function 473

examples

- Begin Data Set exit 278
- End Data Set exit 295
- JES2 initialization 318
- JES3 initialization 321
- Message exit 303
- Record exit 286
- Routing exit 266
- SAP callback daemon configuration file 158
- SMF exit 301
- transform configuration file 171

exit programs 189, 246

- APF-authorizing NetSpool exits 190
- APIPP1X2 control block 191
- filter programs 190, 248
- installing
  - IP PrintWay exits 312
  - NetSpool exits 201
- IP PrintWay Begin Data Set exit 267
- IP PrintWay End Data Set exit 287
- IP PrintWay Message exit 302
- IP PrintWay Record exit 279
- IP PrintWay Response Notification exit 304

exit programs (*continued*)

- IP PrintWay Routing exit 256
- IP PrintWay SMF exit 296
- modifying
  - IP PrintWay exits 313
  - NetSpool exits 202
- NetSpool Beginning of File exit
  - for 3270 data streams 194
  - for SCS data 192
- NetSpool Graphic Escape exit 199
- NetSpool Transparent Data Control exit 197
- order of, IP PrintWay 247
- Print Interface filters 131
- programming considerations
  - IP PrintWay 251
  - NetSpool 190
- sample IP PrintWay exits 312
- sample NetSpool exits 201
- STEPLIB for NetSpool exits 187
- export command, example of 47
- exporting the Printer Inventory 114

## F

FCB parameter (JES2) 319

FCB work-selection criterion 324

FCB, APF authorization for SYS1.IMAGELIB 240, 314

FCB, default for IP PrintWay (JES3) 323

filter programs

- examples of % operator 132
- Infoprint Server filters 137
- input 133, 135
- LPD compatibility 147
- modifying 136
- options for 132
- output 134, 135
- programming considerations 135
- REXX examples 132
- sample 137
- source code, sample 133
- testing 136
- writing a DLL filter 133
- writing a UNIX filter 134

firewall for remote transforms 178

firewalls, email considerations 244

fixed job ID, NetSpool 428, 453

font, default for IP PrintWay

- JES2 317, 320
- JES3 317, 323

form name

- changing in exit 256
- FORMS parameter (JES2) 320
- specifying as JES2 work-selection criteria 321
- specifying as JES3 work-selection criteria 324

FSA definitions for IP PrintWay 328

FSS and FSA definitions, IP PrintWay 325

FSS definitions for IP PrintWay 325

FSS initialization statement

- JES2 319
- JES3 322

FSSNAME parameter (JES3) 322, 323

functional subsystem (FSS), IP PrintWay

- defining multiple FSAs 315
- defining multiple FSSs 314
- JES2 initialization statements 318
- JES3 initialization 321



- functional subsystem (FSS), IP PrintWay *(continued)*
  - name, specifying in
    - JES2 319
    - JES3 322
  - planning 314
- functional subsystem application (FSA), IP PrintWay
  - determining work-selection criteria 316
  - how many IP PrintWay FSAs to create 315
  - IP PrintWay configuration information 328
  - name, specifying in
    - JES2 319
    - JES3 323

## G

- GetAPIVersionNumber function 473
- GetJobInfo function 474
- GetPrinterInfo function 475
- GF10 default font (JES2) 317, 321
- Graphic Exit, NetSpool (APIUGEX)
  - description 199
  - examples 201
  - input 200
  - installing 201
  - modifying 202
  - output 201
  - sample 201
- GS10 default font (JES3) 317

## H

- hardcopy log 94
- hardcopy-message-list attribute 95, 407, 440, 444
- hardcopy-messages attribute 94, 406, 440, 443
- header, document
  - Begin data set exit 267
  - example 278
  - XTP\_DOC\_HEADER\_LENGTH field 275
  - XTP\_DOC\_HEADER\_PTR field 274
  - XTP\_TRANS\_DOC\_HEADER field 272
- HFS directories 32
- hiperspace blocks for IP PrintWay basic mode 325
- Historical Inventory, overview 5
- HTTP Server - Powered by Apache
  - configuring 344
  - creating a started task 351

## I

- IAZJSPA control block 252
- IBM HTTP Server - Powered by Apache
  - configuring 344
  - creating a started task 351
- IBM-1047, specified as default 40, 87
- IDCAMS parameters to allocate transmission-queue data sets 330
- IFAPRDxx member of SYS1.PARMLIB 28
- ignore-dcf-routing-errors attribute 232, 440
- IMS, customization required to print from 25
- Infoprint Central
  - authorizing AOPOPER group to JES commands 353
  - CONSTD statement 378
  - customizing 339
    - IBM HTTP Server - Powered by Apache 343
    - PSF for z/OS 385
    - web browsers 376

- Infoprint Central *(continued)*
  - defining users to RACF 355
  - overview 23
  - RACF notification 373
  - re-creating the search database 378
  - security 352
  - setting up security for
    - actions on PSF printers 361
    - changing print job priority 372
    - daemons 373
    - IP PrintWay job selection rules 365
    - NetSpool logical units 364
    - print jobs 367
    - printer and printer pool definitions 373
    - printers 356
    - software requirements 340
    - starting from the web 377
- Infoprint Port Monitor for Windows
  - compared to other printing protocols 381
  - installing and customizing 381
  - overview 6
- Infoprint Server configuration file aopd.conf
  - attributes 395
  - creating 37
  - editing 45
  - naming in AOPCONF environment variable 48
- InfoPrint Transform Manager for Linux
  - code page conversion table 178
  - configuration file 175
  - customization tasks 167
  - firewall 178
  - resource security 178
  - software requirement 168
- InitAPI function 476
- initial-target attribute 158, 410
- input data streams, types supported 14
- interfaces, programming 131
- INTERVAL parameter 128
- INV parameter
  - in IP PrintWay startup procedure 335
  - in NetSpool startup procedure 187
- inventory attribute 37, 164
- Inventory, Printer
  - authorizing users 52
  - backing up 114
  - FSS and FSA definitions, IP PrintWay 325
  - naming in IP PrintWay startup procedure 335
  - naming in NetSpool startup procedure 187
  - restoring 114
  - sharing 27
- IP address, changing in exit 256
- IP PrintWay
  - comparing basic and extended modes 210
  - customizing 207
  - overview 16
- IP PrintWay basic mode
  - APF-authorizing SYS1.IMAGELIB 314
  - APPL statement 245
  - authorizing to select data sets 222
  - exits, writing 246
  - filters, writing 248
  - FSA definitions 328
  - FSA, defining to JES 314
  - FSS definitions 325
  - FSS, defining to JES 314
  - increasing allowed number of processes 313
  - Language Environment runtime options 337

- IP PrintWay basic mode (*continued*)
  - migrating from basic mode to extended mode 483
  - sendmail, customizing 243
  - sharing AFP printers with PSF 337
  - software requirements 213
  - space requirements in hiperspace 325
  - startup procedure for 332
  - TCPIP.DATA data set, customizing for 241
  - transmission-queue data set, allocating 328
- IP PrintWay extended mode
  - APF-authorizing SYS1.IMAGELIB 240
  - APPL statement 245
  - authorization to write SMF type 6 records 223
  - environment variables 224
  - exits, writing 246
  - filters, writing 248
  - increasing allowed number of processes 215
  - job selection rules 216
  - sendmail, customizing 243
  - setting up the DNS 240
  - sharing AFP printers with PSF 337
  - software requirements 213
  - space required in /var/Printsrv 35
  - starting and stopping printers 337
  - TCPIP.DATA data set, customizing for 241
  - temporary files in /var/Printsrv 35
- IPP protocol, IP PrintWay
  - customization tasks specific to protocol 209
  - environment variables 336
- IPP Server in Print Interface, customizing 150
  - environment variables for 152
  - IPP workstation client, customizing 153
  - specifying port number for 152, 153
  - starting daemon for 151
  - URI used by IPP workstation client 153
- ipp-port-number attribute 152, 153, 440
- ISO8859-1, specified as default 40, 87
- ISPF panels, Infoprint Server
  - defining libraries in TSO logon procedure 77
  - displaying Infoprint Server option 77
  - enabling 76
  - modifying AOPINIT EXEC 77

## J

- Japanese, ISPF panels 77
- Japanese, messages 77, 183
  - for IP PrintWay basic mode 325
- Java software requirements 126, 340
- JAVA\_HOME environment variable 50, 152, 350, 433
- JES work-selection criteria
  - determining 316
  - specifying in JES2 321
  - specifying in JES3 324
- jes-name attribute 37
- JES2
  - FSS statement example 319
  - initialization statements 318
  - PRTnnnnn statement 319
- JES3
  - DEVICE statement 322
  - FSS statement 322
  - initialization statements 321
- JESNEWS data set
  - indicator in ANFUEXT control block 254
  - suppressing SMF type 6 record for 296, 312
- JNAME parameter (JES3) 323

- job attributes, specifying in environment variable 140
- job control language (JCL)
  - sample to link-edit exit 312
  - startup, for Infoprint Server daemons 67
  - startup, IP PrintWay 334
  - startup, NetSpool 182
  - status, for Infoprint Server daemons 76
- job ID, NetSpool 428, 453, 454
- job name for transforms, changing 170
- job selection rules
  - creating 216
  - security 365
- job separator page data area (IAZJSPA) 252
- job-prefix attribute 130, 138, 181, 203, 440
- JobInfo data structure 463
- JobToken data structure 463
- JSPA control block 252

## K

- keyboard
  - navigation 489
  - PF keys 489
  - shortcut keys 489

## L

- LANG environment variable 50, 351, 434
- language attribute 158, 410
- Language Environment runtime libraries 31
- Language Environment runtime options
  - changing in IP PrintWay 337
  - changing in NetSpool 189
- language for IP PrintWay basic mode messages 325
- LANGUAGE parameter, in NetSpool startup procedure 187
- large files, sending to LPD 148
- LC\_ALL environment variable 50, 434
- LC\_CTYPE environment variable 434
- LC\_TIME environment variable 50, 434
- LIBPATH environment variable 50, 159, 351, 434
- libraries, defining in TSO logon procedure (Print Interface) 77
- limit of objects Infoprint Central displays 349, 426
- limits on spooled output 129, 188
- Linux, Infoprint Transform Manager
  - code page conversion table 178
  - configuration file 175
  - customization tasks 167
  - firewall 178
  - resource security 178
  - software requirement 168
- loadprinterinventory command, restore Printer Inventory 120
- log stream
  - deciding on naming convention 100
  - deciding to use 99
  - deciding type 99
  - defining coupling-facility type 106
  - defining DASD-only type 104
  - defining system logger 104
  - determining retention period 100
  - disabling 112
  - enabling 109
  - handling errors 113
  - parameters 105
  - security for 101
  - size of data sets 101
  - switching log streams 111

- log stream (*continued*)
  - viewing messages 113
- log-retention attribute 40, 87, 341, 374, 440
- log.v2db files, deleting 122
- logon procedure for ISPF panels 77
- logstream-name attribute 440
- LOMS for SAP R/3
  - defining 159
  - modifying 160
  - selecting for an output device 161
- LOOKUP statement 242
- LPD compatibility filter 147
- LPD control file first, sending with LPR 148
- lpd\_compat.so 147
- lpd-port-number attribute 145, 146
- LPD, customizing 142
  - command codes supported in 147
  - running Communications Server LPD on same system 144
  - specifying port number for 146
  - starting daemon 145
- LPDEST environment variable 141, 435
- lprfp command, downloading 8
- LUCLASS parameter in NetSpool startup procedure 186

## M

- mail-do-not-add-suffixes attribute 232, 440
- mail-preserve-suffixes attribute 232, 440
- mail-use-first-address attribute 232, 440
- mailer-options attribute 232, 440
- mailer-path-name attribute 232, 440
- man pages, specifying directory for 51, 434
- MANPATH environment variable 51, 434
- max-historical-inventory-size 40, 87, 440
- max-thread-tasks 452
- MAXASSIZE parameter 151
- maximum-active attribute 171
- maximum-idle-time attribute 171
- MAXPROCSYS parameter 129, 180, 216, 313
- MAXPROCUSER parameter 129, 180, 215, 313
- MAXSOCKETS parameter 28
- MAXTHREADTASKS limit
  - AOP\_MAXTHREADTASKS environment variable 420
  - AOIPPD\_MAXTHREADTASKS environment variable 425
  - AOPLPD\_MAXTHREADTASKS environment variable 426
  - AOPNETD\_MAXTHREADTASKS environment variable 428
  - AOPOUTD\_MAXTHREADTASKS environment variable 429
  - AOPSSID\_MAXTHREADTASKS environment variable 431
  - AOPSUBD\_MAXTHREADTASKS environment variable 431
  - AOPWSMD\_MAXTHREADTASKS environment variable 432
- MCS console 184
- message automation 94
- message configuration file, aopmsg.conf
  - creating 96
  - format 406
- Message exit, IP PrintWay
  - examples 303
  - input 302
  - installing 312
  - modifying 313

- Message exit, IP PrintWay (*continued*)
  - output 303
  - overview 302
  - programming considerations 251
  - sample 312
- message log, common
  - allocating space 36
  - MVS system logger 98
  - NetSpool 182
  - reducing size of 122
  - saving 120
  - security for 52
- message-log data set, IP PrintWay
  - allocating space 331
  - defining in startup procedure 336
  - initializing 331
- message-log data set, NetSpool 182
- messages, sending to hardcopy log 94
- migration program, overview 6
- minimum-active attribute 171
- missing system configuration definition 120
- MODE parameter
  - in DEVICE statement (JES3) 323
  - in PRTnnnnn statement (JES2) 320
- MSGDEST parameter (JES3) 322
- multiple TCP/IP stacks 144
- MVS STOP command 73
- MVS system logger
  - deciding on naming convention 100
  - deciding to use 99
  - deciding type 99
  - deciding type of log stream 99
  - defining coupling-facility log stream 106
  - defining coupling-facility type 106
  - defining DASD-only log stream 104
  - defining DASD-only type 104
  - defining log streams 104
  - defining system logger 104
  - determining retention period 100
  - disabling 112
  - enabling 109
  - handling errors 113
  - log stream parameters 105
  - parameters 105
  - security for 101
  - setting up 98
  - size of data sets 101
  - structure parameters 108
  - switching log streams 111
  - viewing messages 113

## N

- navigation
  - keyboard 489
- NetSpool
  - customizing 179
  - environment variables 202
  - exits for transparent data, writing 189
  - filters, writing 190
  - increasing allowed number of processes 180
  - job ID, fixed 428, 453
  - job ID, unaltered 428, 454
  - Language Environment runtime options 189
  - LU classes, specified in startup procedure 186
  - message logs 182
  - overview 14

- NetSpool (*continued*)
  - startup procedure for 182
- netpool-use-fixed-jobid 453
- netpool-use-unaltered-jobid 454
- network printers, customizing PSF for 337
- NETWORK statement 28
- new functions 487
- NIFCB parameter (JES2) 317, 318
- NIUCS parameter (JES2) 317, 318
- NLSPATH environment variable 51, 351, 435
- Notices 493
- NOTIFY parameter, RACF 373

## O

- operating mode
  - determining 29
  - setting 30
- operator group, for security 55
- OPERLOG messages 94
- output devices for SAP R/3, creating 161
- overview of Infoprint Server components 1
  - AFP Printer Driver for Windows 6
  - AFP Viewer plug-in for Windows 7
  - Infoprint Central 23
  - Infoprint Port Monitor for Windows 6
  - IP PrintWay 16
  - NetSpool 14
  - Print Interface 7
  - Printer Inventory 3
  - Printer Inventory Manager 3
  - Transform Interface 21
  - transforms 22
  - Windows client 6

## P

- page definition default for IP PrintWay
  - JES2 317, 319
  - JES3 317, 323
- panels, ISPF
  - defining libraries in TSO logon procedure 77
  - displaying Infoprint Server option 77
  - enabling 76
  - modifying AOPINIT EXEC 77
- parameters
  - JES2 initialization statements 318
  - JES3 initialization statements 321
  - LANGUAGE parameter 187
  - TRACE parameter 187
- password attribute 158, 410
- PATH environment variable 51, 435
- PDEFAULT parameter (JES3) 317, 323
- performance and the TRKCELL parameter (JES2) 320
- performance, improving in IP PrintWay extended mode 242
- permissions for directories and daemons 59
- PIDU, used to back up and restore Printer Inventory 114
- PJL functions in IP PrintWay
  - customizing Infoprint transforms 240
  - JOB command
    - AOP\_PJL environment variable 240
- PNAME parameter (JES3) 322
- port number
  - for IPP Server 152, 153
  - for Print Interface LPD 145, 146
  - used by LPR 147

- PORT statement in TCP/IP 128
- PORTRANGE statement in TCP/IP 128
- ports, low 128
- prefix for job ID 130
- prefix for job identifier 138, 181, 203
- PRESELECT parameter (JES2) 320
- Print Interface
  - AOPPRINT JCL procedure, customizing 141
  - common customization required for all functions 126
  - configuration attributes 145
  - customizing 125
  - environment variables 137
  - environment variables for z/OS UNIX commands 140
  - filters, writing 131
  - increasing allowed number of processes 129
  - IPP Server customization 150
  - JES customization 137
  - LPD customization 142
  - SAP customization 155
  - software requirements 126
  - subsystem customization 163
  - TCP/IP customization 127
- Print Interface, overview 7
- print queue, changing in exit 256
- PRINTDEF statement (JES2) 317, 318
- printer code page, changing default 40, 87
- PRINTER environment variable 141, 435
- Printer Inventory
  - authorizing users 52
  - backing up 114
  - FSS and FSA definitions, IP PrintWay 325
  - naming in IP PrintWay startup procedure 335
  - naming in NetSpool startup procedure 187
  - overview 3
  - restoring 114
  - sharing 27
- Printer Inventory Manager
  - backing up and restoring Printer Inventory 114
  - creating aopd.conf file 37
  - creating aopmsg.conf file 96
  - creating startup and shutdown procedures for 61
  - customizing 27
  - directories 32
  - dynamic enablement of Infoprint Server 28
  - enabling ISPF panels for 76
  - establishing security for 52
  - making runtime libraries available for 31
  - overview 3
  - saving the common message log 120
  - setting environment variables for 46
  - setting Workload Manager goals 123
  - verifying software requirements 28
- printer setup string
  - adding in a filter program 131
  - adding in exit 267
- printer sharing, customizing PSF for 337
- printer, setting a default 141, 435
- PrinterInfo data structure 463
- PrintWay, overview 16
- PROC parameter (JES2) 319
- procedure, startup
  - for Infoprint Server daemons 67
  - IP PrintWay 332
    - sample 334
    - specifying name of in JES2 319
    - specifying name of in JES3 322
  - NetSpool 182

- procedure, startup (*continued*)
  - sample 185
- procedure, status for Infoprint Server daemons 76
- profile, TCP/IP 127
- programming considerations for
  - filter programs 135
  - IP PrintWay exits 251
  - NetSpool exit programs 190
- programming interfaces 131
- PRTnnnnn initialization statement (JES2) 319
- PRTOPTNS parameter, changing in exit 256
- PRTQUEUE parameter, changing in exit 256
- PS, default job prefix 181
- PSF for z/OS
  - AFP printer driver for Windows for creating AFP data 6
  - AFP viewer for viewing AFP data 7
  - customizing for printer sharing 337
  - customizing to use Printer Inventory 385
  - Infoprint Central functions 23
  - Infoprint Server components to customize 25
  - migration program 6
  - Printer Inventory 5, 385
  - roadmap of components to customize 25
  - setting up security for PSF printers 356
  - transforms from AFP on z/OS 22
  - transforms to AFP on z/OS 22

## Q

- QueryToken data structure 464
- QUEUE parameter (JES2) 319

## R

- RACF
  - AOP.ADMINISTRATOR profile 53, 355
  - AOP.CONFIGURATION profile 54
  - AOP.DAEMON profile 373
  - AOP.NETSPOOL profile 364
  - AOP.PRIORITY profile 372
  - AOPADMIN group 54
  - AOPCONF group 54
  - AOPOPER group 54
  - aopsetup shell script 59
  - auditing, requesting 373
  - notification, requesting 373
  - PRINTSRV class 53, 54
  - security for
    - actions on PSF printers 361
    - changing print job priority 372
    - common message log 52
    - daemons 373
    - Infoprint Central 352
    - IP PrintWay job selection rules 365
    - IP PrintWay transmission-queue data set 331
    - NetSpool logical units (LUs) 364
    - operator commands 52
    - print jobs 367
    - printer and printer pool definitions 373
    - Printer Inventory 52
    - printers 356
- Record exit, IP PrintWay
  - examples 286
  - input 279
  - installing 312
  - modifying 313
- Record exit, IP PrintWay (*continued*)
  - output 284
  - overview 279
  - sample 312
- record format
  - input to Begin Data Set exit 271
  - input to End Data Set exit 291
  - input to Routing exit 261
- region size
  - IP PrintWay 335
  - NetSpool 186
- registers
  - for Begin Data Set exit 268
  - for Beginning of File exit 192, 195, 200
  - for Block Letter program 311
  - for End Data Set exit 288
  - for Message exit 302
  - for Record exit 279
  - for SMF exit 296
  - for the Routing exit 257
  - for Transparent Data exit 197
- ReleaseTransform function 476
- rembak command, AIX 149
- remote transform configuration file, aopxfd.conf
  - creating 175
  - format 175, 408
- remote-transform-manager-ip-address attribute 175, 408
- remote-transform-manager-port-number attribute 175, 408
- RequestTransform function 477
- requirements, software
  - Infoprint Central 340
  - IP PrintWay 213
  - NetSpool 180
  - Print Interface 126
  - Printer Inventory Manager 28
  - PSF for z/OS 385
  - Transform Interface 168
- resolve-printway-printers attribute 214
- Resource Access Control Facility (RACF)
  - AOP.ADMINISTRATOR profile 53, 355
  - AOP.CONFIGURATION profile 54
  - AOP.DAEMON profile 373
  - AOP.NETSPOOL profile 364
  - AOP.PRIORITY profile 372
  - AOPADMIN group 54
  - AOPCONF group 54
  - AOPOPER group 54
  - aopsetup shell script 59
  - auditing, requesting 373
  - notification, requesting 373
  - PRINTSRV class 53, 54
  - security for
    - actions on PSF printers 361
    - changing print job priority 372
    - common message log 52
    - daemons 373
    - Infoprint Central 352
    - IP PrintWay job selection rules 365
    - IP PrintWay transmission-queue data set 331
    - NetSpool logical units (LUs) 364
    - operator commands 52
    - print jobs 367
    - printer and printer pool definitions 373
    - Printer Inventory 52
    - printers 356
- resources, security for 178
- Response Notification exit 304



- restoring the Printer Inventory 114
- restrict ports 128
- resubmit for filtering
  - AOPRESUBMITUSER environment variable 431
  - customizing IP PrintWay to use font and page definition in printer definition 317
  - exit processing, description 250
  - specifying for transforms 168
  - suppressing SMF record 296
- RETAINF parameter, changing in exit 256
- RETAINS parameter, changing in exit 256
- RETRYL parameter, changing in exit 256
- RETRYT parameter, changing in exit 256
- REXX examples, for filters 132
- RFC 1179, LPD command codes supported 147
- roadmap of components to customize 25
- ROMS for SAP R/3, defining 159
- ROUTECD parameter (JES2) 320, 321
- routing criteria, changing in exit 256
- Routing exit, IP PrintWay
  - examples 266
  - input 257
  - installing 312
  - modifying 313
  - output 263
  - overview 256
  - programming considerations 251
  - sample 312
- runtime libraries for Language Environment 31
- runtime options for C++ Language Environment
  - changing in IP PrintWay 337
  - changing in NetSpool 189

## S

- S2STG DSECT 191
- SAF (System Authorization Facility)
  - AOP.ADMINISTRATOR profile 53, 355
  - AOP.CONFIGURATION profile 54
  - AOP.DAEMON profile 373
  - AOP.NETSPOOL profile 364
  - AOP.PRIORITY profile 372
  - AOPADMIN group 54
  - AOPCONF group 54
  - AOPOPER group 54
  - aopsetup shell script 59
  - auditing, requesting 373
  - notification, requesting 373
  - PRINTSRV class 53, 54
  - security for
    - actions on PSF printers 361
    - changing print job priority 372
    - common message log 52
    - daemons 373
    - Infoprint Central 352
    - IP PrintWay job selection rules 365
    - IP PrintWay transmission-queue data set 331
    - NetSpool logical units (LUs) 364
    - operator commands 52
    - print jobs 367
    - printer and printer pool definitions 373
    - Printer Inventory 52
    - printers 356
- sample exits 312
- SAP Callback daemon configuration file, aopsapd.conf
  - client attribute 158, 410
  - creating 157

- SAP Callback daemon configuration file, aopsapd.conf
  - (continued)
  - example 158
  - format 157, 410
  - initial-target attribute 158, 410
  - language attribute 158, 410
  - naming in environment variable 159
  - password attribute 158, 410
  - syntax rules 158
  - system attribute 158, 410
  - userid attribute 158, 410
- SAP R/3 printing
  - configuration file 157
  - configuration file for Callback daemon 157
  - creating an SAP R/3 output device 161
  - customization tasks 155
  - environment variable for configuration file 159
  - overview 11, 12
  - ROMS and LOMS for 159
  - software requirements 126
  - user for Callback daemon 156
- SCEERUN library 31
- SCEERUN2 library 31
- scheduler work block text unit (SWBTU) 253
- SCLBDLL library 31
- SCS data stream support in NetSpool 15
- SCS data streams, NetSpool exits for 189
- sdbu command 378
- search database, re-creating 378
- security, establishing
  - AOP.ADMINISTRATOR profile 53, 355
  - AOP.CONFIGURATION profile 54
  - AOP.DAEMON profile 373
  - AOP.NETSPOOL profile 364
  - AOP.PRIORITY profile 372
  - AOPADMIN group 54
  - AOPCONF group 54
  - AOPOPER group 54
  - aopsetup shell script 59
  - auditing, requesting 373
  - for actions on PSF printers 361
  - for changing print job priority 372
  - for common message log 52
  - for daemons 373
  - for Infoprint Central 352
  - for IP PrintWay job selection rules 365
  - for IP PrintWay transmission-queue data set 331
  - for NetSpool logical units (LUs) 364
  - for operator commands 52
  - for print jobs 367
  - for printer and printer pool definitions 373
  - for Printer Inventory 52
  - for printers 356
  - notification, requesting 373
  - PRINTSRV class 53, 54
- select-work-from-hold-queue attribute 232, 437
- sending comments to IBM xvii
- sendmail
  - creating an alias file 244
  - customizing 243
  - environment variables (basic mode) 336
  - environment variables (extended mode) 225
  - firewalls 244
  - specifying alternate directory 244
  - specifying options 244
- separator page, adding in exit 267, 287

- Server Message Block (SMB), customizing Print Interface support for 149
- Server Message Block compared to other printing protocols 381
- SetTerminationHandler function 477
- SETUP parameter (JES2) 320
- setup string, adding in exit 267
- setup strings for printers, adding in a filter program 131
- shared printers, SMB 149
- sharing IP PrintWay transmission-queue data set 329
- sharing of printers, customizing PSF for 337
- shortcut keys 489
- shutdown procedure, for Infoprint Server daemons 72
- SMB compared to other printing protocols 381
- SMB support, software requirements 126
- SMB, customizing Print Interface support for 149
- SMF exit, IP PrintWay
  - examples 301
  - input 296
  - installing 312
  - modifying 313
  - output 300
  - overview 296
  - programming considerations 251
  - sample 312
- SMF record
  - replacing in exit 296
  - suppressing in exit 296
- smf-recording attribute 215, 232, 437
- SMTP 243
- SNMP
  - read community name for Infoprint Central 349
  - write community name for Infoprint Central 349
- sockets, increasing maximum 28
- software requirements
  - Infoprint Central 340
  - IP PrintWay 213
  - NetSpool 180
  - Print Interface 126
  - Printer Inventory Manager 28
  - PSF for z/OS 385
  - Transform Interface 168
- special characters in configuration files 45, 46, 158, 406, 409, 411
- SPOOLDEF statement (JES2) 317, 318
- stacks, multiple TCP/IP 144
- START parameter (JES2) 320
- start-command attribute 170
- start-daemons attribute 37, 145, 151, 164, 168, 181, 213, 341
- starting and stopping printers from TSO 337
- starting daemons, specifying names in configuration file 37, 145, 151, 164, 168
- startup procedure
  - for Infoprint Server daemons 67
  - for IP PrintWay 332
    - sample 334
    - specifying name of in JES2 319
    - specifying name of in JES3 322
  - for NetSpool 182
    - sample 185
  - Printer Inventory Manager 67
- status procedure
  - for Infoprint Server daemons 76
  - Printer Inventory Manager 76
- STDENV data definition (DD) 68, 336, 413
- STEPLIB environment variable 31, 224, 313, 435
- STEPLIBLIST facility 31, 216, 224
- stopping daemons 72
- structure parameters 108
- subsystem, Print Interface
  - customizing 163
- suffixes for email attachments 228, 235, 419, 449
- suppress-post-unix-filter-formatting attribute 232, 437
- SWBTU control block 253
- syntax rules for SAP Callback configuration file 158
- SYS1.IMAGELIB 224, 240, 314
- SYS1.PARMLIB 28
- SYS1.PARMLIB, updating
  - for IP PrintWay basic mode 313
  - for IPP Server 151
  - for NetSpool 180
  - for Print Interface 129
  - MAXPROCUSER statement for IP PrintWay 215
  - MAXSOCKETS parameter 28
  - STEPLIBLIST facility, for SCEERUN and SCLBDLL libraries 31
- SYSLOG messages 94
- SYSOUT statement (JES3) 321
- syntax considerations
  - IP PrintWay extended mode 211
  - JES3 limitation 218
  - Printer Inventory 27
- SYSTCPD DD statement 336
- system attribute 158, 410
- System Authorization Facility (SAF)
  - AOP.ADMINISTRATOR profile 53, 355
  - AOP.CONFIGURATION profile 54
  - AOP.DAEMON profile 373
  - AOP.NETSPOOL profile 364
  - AOP.PRIORITY profile 372
  - AOPADMIN group 54
  - AOPCONF group 54
  - AOPOPER group 54
  - aopsetup shell script 59
  - auditing, requesting 373
  - notification, requesting 373
  - PRINTSRV class 53, 54
  - security for
    - actions on PSF printers 361
    - changing print job priority 372
    - common message log 52
    - daemons 373
    - Infoprint Central 352
    - IP PrintWay job selection rules 365
    - IP PrintWay transmission-queue data set 331
    - NetSpool logical units (LUs) 364
    - operator commands 52
    - print jobs 367
    - printer and printer pool definitions 373
    - Printer Inventory 52
    - printers 356
- system configuration definition
  - attributes 437
  - editing 91
  - ISPF panel for 91
  - missing 120
- system flow diagram 8
- system logger
  - deciding on naming convention 100
  - deciding to use 99
  - deciding type of log stream 99
  - defining coupling-facility log stream 106
  - defining DASD-only log stream 104
  - defining log streams 104



- system logger (*continued*)
  - determining retention period 100
  - disabling 112
  - enabling 109
  - handling errors 113
  - log stream parameters 105
  - security for 101
  - setting up 98
  - size of data sets 101
  - structure parameters 108
  - switching log streams 111
  - viewing messages 113
- SYSTEM parameter (JES3) 322

## T

- TCP/IP
  - customizing sendmail 243
  - error when initialization is not finished 75
  - modifying the hlq.PROFILE.TCPIP data set 127
  - modifying the TCPIP.DATA data set 241
  - multiple stacks, configuring 144
  - specifying the name of 241, 325
  - standard translation table, IP PrintWay basic mode 325
  - SYSTCPD DD statement 336
- TCPCONFIG statement in TCP/IP 127
- TCPIP.DATA data set
  - DATASETPREFIX statement 242
  - for IP PrintWay basic mode 242
  - search order used by IP PrintWay basic mode 243
  - specifying name in IP PrintWay basic mode 334
  - SYSTCPD DD statement 336
  - TCPIPJOBNAME statement 242
- TCPIPJOBNAME statement 242
- TCPRCVBUFRSIZE parameter 128
- TCSENDBFRSIZE parameter 128
- temporary files written by
  - IP PrintWay basic mode 325
  - IP PrintWay extended 35
- TERM parameter (JES3) 322
- TermAPI function 478
- termination handler, writing 479
- time zone, setting 52, 437
- timeout value, increasing with sending large files to LPD 148
- TRACE parameter, in NetSpool startup procedure 187
- trace, specifying for IP PrintWay basic mode 328
- TRACEOPT DD statement 187, 188
- tracing parameters for IP PrintWay basic mode 325
- tracing, turning on in AOPSTART procedure 67
- trailer, document
  - Begin data set exit 267
  - example 278
  - XTP\_DOC\_TRAILER\_LENGTH field 275
  - XTP\_DOC\_TRAILER\_PTR field 274
  - XTP\_TRANS\_DOC\_TRAILER field 272
- transform attribute 170
- transform classes 169
- transform configuration file, aopxfd.conf
  - creating 169
  - environment attribute 170
  - example 171
  - format 169, 409
  - maximum-active attribute 171
  - maximum-idle-time attribute 171
  - minimum-active attribute 171
  - start-command attribute 170
  - start-daemons attribute 168

- transform configuration file, aopxfd.conf (*continued*)
  - transform attribute 170
- Transform data structure 464
- Transform Interface
  - customizing 167
  - overview 21
- Transform Manager configuration file, Infoprint Server 169
- Transform Manager for Linux
  - code page conversion table 178
  - configuration file 175
  - customization tasks 167
  - firewall 178
  - resource security 178
  - software requirement 168
- transforms
  - components to customize 25
  - overview 22
- translation table, standard 325
- transmission-queue data set, IP PrintWay
  - allocating and initializing 330
  - defining in startup procedure 336
  - initializing 331
  - overview 328
  - restricting access 331
  - sharing between IP PrintWay FSSs 314, 329
- Transparent Data Control exit, NetSpool
  - description 197
  - examples 199
  - input 197
  - installing 201
  - modifying 202
  - output 198
  - programming considerations 190
  - sample 201
- transparent data exits in NetSpool, writing 189
- TRKCELL parameter
  - JES2 320
  - SPOOLDEF 318
- TRUNC parameter 137
- TSO logon procedure, defining libraries in 77
- TYPE parameter (JES3) 322
- types of LU sessions supported 14
- TZ environment variable 52, 437

## U

- UCS parameter (JES2) 320
- unaltered job ID, NetSpool 428, 454
- URI for printer 153
- URL for printer 153
- user interface
  - ISPF 489
  - TSO/E 489
- userid attribute 158, 410

## V

- VARFILE data definition (DD) 70
- VTAM APPL statement for IP PrintWay, creating 245
- VTAM applications, customization required to print from 25
- VTAM protocol, IP PrintWay
  - AOP\_APPLID variable 225
  - APPL ID, specifying in basic mode 325
  - APPL statement, creating in VTAM 245
  - customization tasks specific to protocol 209
  - FSS definition, creating 325

VTAM protocol, IP PrintWay *(continued)*  
software required 213

## W

WC parameter (JES3) 324  
web browsers, customizing for Infoprint Central 376  
Windows client  
    customizing 381  
    overview 6  
Windows printing, customization required 25  
Workload Manager 123  
WritePrintFile function 478  
WS parameter  
    in DEVICE statement (JES3) 324  
    in PRTnnnnn statement (JES2) 321

## X

xcf-group-qualifier attribute 37  
XML Toolkit  
    level required for Infoprint Central 340  
    LIBPATH environment variable 351

## Z

zFS directories 32





Product Number: 5650-ZOS  
5655-N62

Printed in USA

SA38-0691-01

