IBM
Version 2 Release 4

Infoprint Transforms to AFP for z/OS

G550-0443-06

Note

Before using this information and the product it supports, read the information in "Notices" on page 133.

This edition applies to Version 2 Release 4 of IBM® Infoprint Transforms to AFP for z/OS® (program number 5655–N60) and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition replaces G550–0443–05.

Contents

List of Figures

List of Tables

About this publication

This publication describes Version 2 Release 4 (V2.4) of IBM Infoprint Transforms to AFP for z/OS (program number 5655-N60).

With this product, you can transform documents to Advanced Function Presentation (AFP) format from:

- HP Printer Control Language (PCL)
- Adobe Portable Document Format (PDF)
- Adobe PostScript
- SAP R/3 System Generic Output Format (SAPGOF)

The transforms require Infoprint Server, which is a separately priced feature of z/OS. The transforms run on all supported releases of z/OS.

Who should read this publication

This publication is for anyone who needs to transform documents to AFP, system programmers who customize the transform products, administrators responsible for maintaining the Infoprint Server Printer Inventory, and diagnosticians who must diagnose transform errors.

Readers should be familiar with Infoprint Server, AFP, z/OS UNIX System Services, and z/OS job control language (JCL).

How to read syntax diagrams

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

			For example:	
This notation:	**Means:**	**You enter:**	**This publication 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	*filename ...*	`file1 file2`
Lowercase	Item the system defines	As shown, in lowercase	`lp`	`lp`
Lowercase italics	Variable item	A value for the item	`MOUNT` *devnum*	`MOUNT A30`
Parentheses	List of items	The parentheses and one or more items from the list	`(GT10,GT12)`	`(GT10,GT12)`

This notation:	Means:	You enter:	For example:	
			This publication shows:	You enter:
Special characters	Various symbols	As shown	`%filter-options`	`%filter-options`
Underline	Default	The item, or you can omit it	`K T REF`	`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 related information.

Preventive Service Planning information

Before installing Infoprint transforms, you should review the current Preventive Service Planning (PSP) information, also called the *PSP bucket*. You should also periodically review the current PSP information.

The PSP upgrade IDs and subsets are:

Table 1: PSP upgrade IDs and subsets

Transform	Upgrade ID	Subset
IBM Infoprint Transforms to AFP	5655N60	HXFR2D0

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 delivered several weeks prior to installation.

Related documentation

This section lists documentation for Infoprint Server and other related documentation.

Table 2: Documentation for Infoprint Server

Publication	Form number	Description
z/OS Infoprint Server Introduction	SA38-0692	Introduces Infoprint Server. This publication contains printing scenarios that show how you can use Infoprint Server in your installation.

Table 2: Documentation for Infoprint Server (continued)		
Publication	**Form number**	**Description**
z/OS Infoprint Server Customization	SA38-0691	Describes customization tasks for Infoprint Server. This publication 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.
z/OS Infoprint Server Operation and Administration	SA38-0693	Describes operator procedures and administrative tasks for Infoprint Server. This publication 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®.
z/OS Infoprint Server User's Guide	SA38-0695	Describes user tasks for Infoprint Server. This publication 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.
z/OS Infoprint Server Messages and Diagnosis	GA32-0927	Describes messages from Infoprint Server. This publication also describes how to use Infoprint Server tracing facilities to diagnose and report errors.

Table 3: Licensed program specification	
Licensed program specification	**Form number**
IBM Infoprint Transforms to AFP for z/OS	G550-0445

Table 4: Program directory	
Program directory	**Form number**
IBM Infoprint Transforms to AFP for z/OS	GI11-9497

How to send your comments to IBM

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

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

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

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

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

Feedback on the z/OS product documentation and content
If your comment is about the information that is provided in the z/OS product documentation library, send a detailed email to mhvrcfs@us.ibm.com. We welcome any feedback that you have, including comments on the clarity, accuracy, or completeness of the information.

To help us better process your submission, include the following information:

- Your name, company/university/institution name, and email address
- The following deliverable title and order number: IBM Infoprint Transforms to AFP for z/OS, V2.4, G550-0443-06
- The section title of the specific information to which your comment relates
- The text of your comment.

When you send comments to IBM, you grant IBM a nonexclusive 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

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

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

Summary of changes

This document describes IBM Infoprint Transforms to AFP, Version 2 Release 4. It contains information that was previously presented in *IBM Infoprint Transforms to AFP for z/OS*, G550-0443-05.

Technical changes or additions to the text and illustrations are indicated by a vertical line in the left margin.

New information

- "Examples -- pcl2afp" on page 19, "Examples -- pdf2afp" on page 28, "Examples -- ps2afp" on page 37, and "Examples -- sap2afp" on page 42 include new examples.
- The -n option has been added to the `pdf2afp` and `ps2afp` transforms. This option sets the threshold for rendering grays and colors in black. See "pdf2afp—Transform PDF data to AFP data" on page 21, "ps2afp—Transform PostScript data to AFP data" on page 30, and "Setting a halftone threshold" on page 120.
- The -r option of the `pdf2afp` and `ps2afp` transforms now supports additional resolutions of 360, 720, and 1200 pels per inch. See "pdf2afp—Transform PDF data to AFP data" on page 21, "ps2afp—Transform PostScript data to AFP data" on page 30, and "Creating AFP output for 360-pel, 720-pel, and 1200-pel printers" on page 120.
- These messages are added to "Messages" on page 109:

  AOP2026W
  AOP2028I
  AOP2029E
  AOP2507E

- You can now install Infoprint Transforms to AFP through Product ServerPac. A procedure for completing a ServerPac installation is added. To enable this function, the default installation directory for this product is changed from `/usr/lpp/Printsrv` to `/usr/lpp/IBM/XformsToAFP/V2R4`. See "Completing the installation" on page 117.
- Product-level enable and disable support using z/OS Product Registration Services is added to the release. See "Enabling V2.4 transforms in the SYS1.PARMLIB member" on page 119.

Changed information

- The PSP subset is changed to HXFR2D0. See "Preventive Service Planning information" on page xii.
- The -m option is no longer specified by default in the `pdf2afp` and `ps2afp` transform configuration file. See "Format of a PDF to AFP and PostScript to AFP transform entry" on page 61 and "Starting Infoprint Server with sufficient memory" on page 67.
- `/var/Printsrv/xfd` is added as a default of the AOPTRACEDIR environment variable. See "Environment variables for the PDF to AFP and PostScript to AFP transforms" on page 62, "Trace the transform" on page 67, and "Running traces" on page 106.
- "Adding fonts" on page 69 is updated with a new directory path.
- These messages are changed in "Messages" on page 109:

  AOP2201E
  AOP2505E

Deleted information

- References to z/OS V1.12 and V1.13 are deleted because they are no longer in service.
- References to InfoPrint Manager are deleted.

- Because WorldType fonts are included in the z/OS Font Collection, information about fonts formerly included with this product are deleted.

Chapter 1. Introduction

This information introduces Version 2 Release 4 (V2.4) of IBM Infoprint Transforms to AFP for z/OS (program number 5655-N60).

This product provides data-stream transforms that let you transform documents to Advanced Function Presentation (AFP) format from other data stream formats. These transforms let you print non-AFP data on AFP printers, also known as Intelligent Printer Data Stream (IPDS) printers. You can transform documents to AFP format from:

- Hewlett-Packard (HP) Printer Control Language (PCL)
- Adobe Portable Document Format (PDF)
- Adobe PostScript
- SAP R/3 System Generic Output Format (SAPGOF)

What is new in Version 2 Release 4

- IBM Infoprint Transforms to AFP can be installed using Product ServerPac.
- IBM Infoprint Transforms to AFP supports product registration and enablement.
- The default installation directory has changed from `/usr/lpp/Printsrv` to `/usr/lpp/IBM/XformsToAFP/V2R4`.
- New in the PDF to AFP Transform:
  - The -r *resolution* option for the `pdf2afp` command now accepts three additional values (360, 720, and 1200).
- New in the PS to AFP Transform:
  - The -r *resolution* option for the `ps2afp` command now accepts three additional values (360, 720, and 1200).

What is new in Version 2 Release 3

- The transforms can create MO:DCA Presentation Interchange Set (IS) data, which now includes the newest interchange set, MO:DCA IS/3. MO:DCA IS/3 is a subset of MO:DCA that defines an interchange format for presentation documents and includes structured fields that are not found in MO:DCA IS/1. While providing interoperability among AFP products that are MO:DCA IS/3 compliant, MO:DCA IS/3 also provides enhanced functions, including support for color and the latest fonts, images, and graphics.
- New in the PCL to AFP Transform:
  - The -r *resolution* option for the `pcl2afp` command now accepts two additional values (720 and 1200).
  - The `pcl2afp` command supports the new -C option. The -C option specifies that the transform processes the color PCL files that the PCL 5c driver created to grayscale output. If this option is not specified for transforming PCL 5c files, the resulting AFP file might not print correctly. If this option is not specified for transforming PCL 5c files, the resulting AFP file might print incorrectly.
  - The `pcl2afp` command supports the new -I option, which specifies that the transform produces MO:DCA IS/3-compliant output.
  - The PCL to AFP transform AOP_RESOLUTION environment variable accepts two additional values (720 and 1200).
- New in the PDF to AFP Transform:

  - The new `pdf2afp` command option -I specifies that the transform produces MO:DCA IS/3-compliant output.
  - The PDF to AFP transform now supports PDF 1.7 (ISO 32000) and earlier levels (except for the various interactive features of any PDF version).
- New in the PS to AFP Transform:
  - The new `ps2afp` command option -I specifies that the transform produces MO:DCA IS/3-compliant output.
- New in the SAP to AFP Transform:
  - The SAP to AFP transform supports the use of TrueType fonts (required for MO:DCA IS/3-compliant output created by using the SPF transform). Several TrueType fonts from the WorldType 8.1 library are now included in the product.
  - A new environment variable, AOP_WORLDTYPE_PATH, allows you to specify that TrueType fonts have been installed in a non-standard path.
  - Three new configuration files support TrueType fonts for the SAP to AFP transform: `fonts.tab.unicode.truetype`, `fonts.tab.unicode.embed.truetype`, and `pagedef.tab.unicode.truetype`.
  - The -r *resolution* option for the `sap2afp` command now accepts two additional values (720 and 1200).
  - The `sap2afp` command supports five new options:
    - The -a *imagetype* option specifies the IOCA function set that the transform uses when transforming Output Text Format (OTF) color image data, allowing the transform to generate Image Object Content Architecture (IOCA) FS45 output for SAP OTF data streams.
    - The -b and -k options place bar codes (-b) and boxes (-k) in AFP output by using the SAP specification instead of the AFP specification.
    - The -f option specifies the use of TrueType fonts for Unicode SAP files.
    - The -h option specifies that the OTF input file was generated using SAPGOF_ST_CHARS:X entry from the SAP Unicode system.
    - The -u option provides compatibility for customers who are using older printers that have only ASCII (and no Unicode) support, but are submitting data that contains Unicode code pages through the transform. For example, a customer might have a file generated using a SAPGOF device that uses Unicode code pages, and is printing the output on an ASCII printer.
    - The -I option specifies that the transform produces MO:DCA IS/3-compliant output.

Transform names and levels

The names of the transforms to AFP and the levels of data streams that they transform are:

This transform:	**Transforms to AFP from these levels:**
PCL to AFP	PCL 6 (XL, 5, 5e, 5c). **Note:** PCL 5c support results in grayscale output only.
PDF to AFP	PDF 1.7 and earlier levels
PostScript to AFP	PostScript Language Level 3
SAP to AFP	SAP R/3 SAPGOF Release 4.6C and most functions in SAP R/3 Release 6.10.

Using transforms with Infoprint Server

This information describes how you can use the transforms together with Infoprint Server to meet your printing needs. It describes these scenarios:

- “Printing web documents on AFP printers” on page 3
- “Printing PDF documents on AFP printers” on page 3
- “Printing SAP R/3 documents on AFP printers” on page 4

Printing web documents on AFP printers

A company wants to print web-based documents on an AFP printer. Here is how the company can use the PCL to AFP transform, Infoprint Server, and Print Services Facility™ (PSF) to meet its requirement:

1. The user installs the Infoprint Port Monitor on her Windows system and configures it to print on the AFP printer. The user associates a PCL driver with the Infoprint port.
2. The user views the web-based document using a browser, such as Microsoft Internet Explorer. The user prints the document using the standard print-submission method that the browser provides, selecting the printer configured at the Infoprint port.

   As an option, the user specifies Infoprint Server job attributes in the Infoprint Port Monitor window. For example, the user might specify distribution information for PSF to print on the printer's separator page and the name of a form definition for PSF to use when it prints the document.
3. The PCL driver associated with the Windows printer creates a document in PCL format.
4. The Infoprint Port Monitor sends the PCL document and job attributes over the TCP/IP network to Infoprint Server.
5. Infoprint Server determines that it needs to call the PCL to AFP transform. The transform converts the PCL document to AFP format.
6. Infoprint Server writes the AFP document to an output data set on the JES spool.

   Infoprint Server also specifies JES output parameters that route the output data set to the PSF-controlled AFP printer and tell PSF how to process the document. For example, Infoprint Server specifies distribution information and the name of the form definition in JES output parameters.
7. PSF selects the output data set from the JES spool. It uses the form definition to process the AFP document and sends it to the AFP printer.

Figure 1 on page 3 shows how to print a web document on an AFP printer.

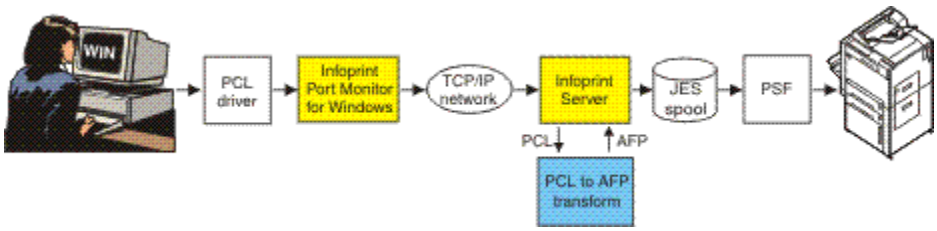

Figure 1: Printing web documents on AFP printers

Printing PDF documents on AFP printers

A company wants to print PDF documents on a high-speed AFP printer. Here is how the company can use the PDF to AFP transform, Infoprint Server, and PSF to meet its requirement:

1. The user installs the `lprafp` command on his Windows system. Using the command, the user submits the PDF document to Infoprint Server and specifies the name of the AFP printer on the command.

As an option, the user specifies Infoprint Server job attributes on the `lprafp` command. For example, the user might specify the number of copies and the name of a form definition for PSF to use when it prints the document.

2. The `lprafp` command sends the PDF document and job attributes over the TCP/IP network to Infoprint Server.
3. Infoprint Server determines that it needs to call the PDF to AFP transform. The transform converts the PDF document to AFP format.
4. Infoprint Server writes the AFP document to an output data set on the JES spool.

   Infoprint Server also specifies JES output parameters that route the output data set to the PSF-controlled AFP printer and tell PSF how to process the document. For example, Infoprint Server specifies the number of copies and the name of the form definition in JES output parameters.
5. PSF selects the output data set from the JES spool. It uses the form definition to process the AFP document and sends it to the AFP printer.

Figure 2 on page 4 shows how to print a PDF document on an AFP printer.

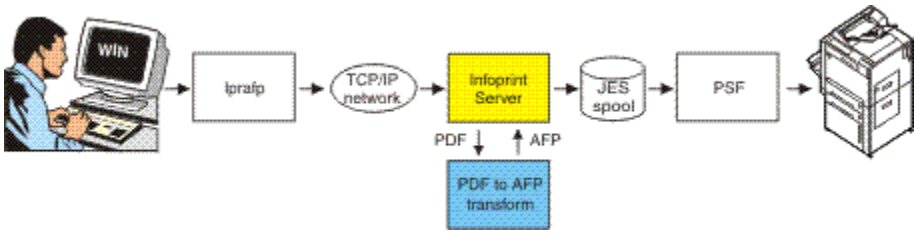

Figure 2: Printing PDF documents on AFP printers

Printing SAP R/3 documents on AFP printers

A company wants to print purchase orders created by SAP R/3 applications on a high-speed AFP printer. Here is how the company can use the SAP to AFP transform, Infoprint Server, the SAP R/3 Application Server for z/OS, and PSF to meet its requirement:

1. Using the SAP R/3 GUI, a user submits an SAP SAPGOF (ABAP or SAP OTF format) document for printing on an SAP R/3 output device that the SAP R/3 administrator has associated with a printer defined to Infoprint Server.

   The SAP R/3 application server sends the document over the TCP/IP network to the spool work process of the SAP R/3 Application Server for z/OS. (The spool work process must run on the same system as Infoprint Server.)
2. The SAP R/3 Application Server's spool work process submits the print request to the Infoprint Server SAP Output Management System (OMS).
3. Infoprint Server determines that it needs to call the SAP to AFP transform.
4. The SAP to AFP transform converts the SAP R/3 OTF data to AFP format and ABAP data to line data format. The transform selects appropriate AFP resources, such as a form definition, for PSF to use when it prints the document.
5. Infoprint Server writes the AFP document to an output data set on the JES spool.

   Infoprint Server also specifies JES output parameters that route the output data set to the AFP printer and tell PSF how to process the document. For example, the transform specifies the name of the form definition in a JES output parameter.
6. PSF selects the output data set from the JES spool. It uses the form definition to process the AFP document and sends it to the AFP printer.
7. When the data set finishes printing or is deleted from the JES spool, Infoprint Server sends notification back to the SAP application server that submitted the print request. The notification indicates whether the data set printed successfully.

Figure 3 on page 5 shows how to print an SAP R/3 document on an AFP printer.

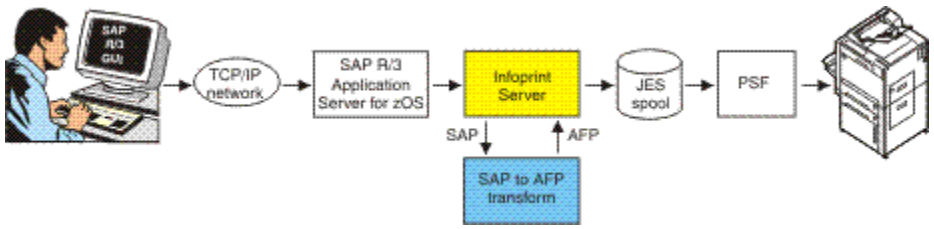

Figure 3: Printing SAP R/3 documents on AFP printers

Methods for transforming documents

You can use either of these methods to transform documents to AFP:

z/OS UNIX transform commands

z/OS UNIX System Services users can use transform commands to convert documents to AFP format from another format without printing the documents. You can run the transform commands from the z/OS UNIX command line, or you can use the Infoprint Server AOPBATCH program to run them.

If you intend to print a document many times, you might want to transform the document once and the print the output in a separate step. This is more efficient than transforming the document every time you print it.

You can use these transform commands:

- `pcl2afp` for PCL to AFP
- `pdf2afp` for PDF to AFP
- `ps2afp` for PostScript to AFP
- `sap2afp` for SAP to AFP

The input document and the output AFP document can be in an MVS™ data set or in a z/OS UNIX file.

For example, to transform the PostScript document `myfile.ps` to an AFP document named `myfile.afp`, enter:

```
ps2afp -o myfile.afp myfile.ps
```

Printer definitions

The administrator can specify transform filters in printer definitions in the Infoprint Server Printer Inventory. When you do this, Infoprint Server automatically calls the appropriate transform filter before it prints the document or sends it to an email destination. You use this method when you want to transform and print AFP documents but not save the transformed output.

For example, suppose that the administrator has configured a printer definition named "myprinter" to use the PDF to AFP transform filter:

- You can use the z/OS UNIX `lp` command to transform and print a PDF document using this printer definition. For example, to transform and print a PDF document in file `myfile.pdf`, enter this z/OS UNIX command:

  ```
  lp -d myprinter myfile.pdf
  ```

- You can transform and print the PDF output of a batch job using the Print Interface subsystem and this printer definition. For example, to transform and print a PDF document created by a batch program using the subsystem named A0P1, submit this JCL:

  ```
  //MYJOB  JOB …
  //STEP1  EXEC PGM=USERA
  //DD1    DD SUBSYS=(AOP1,'myprinter')
  ```

Specifying transform options

You can use transform options to tell the transforms how you want the data to be transformed to AFP. You can use these methods to specify transform options:

Transform configuration file

The administrator can specify some transform options in the transform configuration file `aopxfd.conf`. For example, you can tell the PCL to AFP transform the resolution of the printer in the AOP_RESOLUTION environment variable in the transform configuration file.

The administrator can create separate classes of a transform with different transform options in each class. For example, the administrator could create a transform class for AFP printers with a resolution of 300 pels, and a class for printers with a resolution of 600 pels. The administrator names and defines the transform classes in the transform configuration file. To use a transform class that the administrator has defined:

- The administrator can specify the name of the transform class in the -c transform option in the printer definition.
- The job submitter can specify the name of the transform class in the -c transform command option.

Transform command options

You can specify some transform options on the transform commands. For example, when you enter the `pdf2afp` command, you can specify option -c *transformclass* to tell the transform the name of a transform class to use.

When you transform and print documents, you can specify transform command options in the filter-options job attribute. For example, when you enter the `lp` command, you can specify the -c option in the filter-options job attribute.

Software requirements

The transforms to AFP run on z/OS V2.1 or later (Program Number 5650-ZOS).

This additional IBM software is required:

- Infoprint Server feature of z/OS with all current service applied
- SMP/E for z/OS, Version 3.4 or later (5655-G44) to install Infoprint Transforms to AFP
- PSF Version 4.3 or later for z/OS (5655–M32) to print the transform output on AFP printers

Performance considerations

Transforming documents from the document format in which they were created to another document format uses more system resources than printing the documents on printers that support the original document format.

The impact on system and network resources varies depending on items such as:

- Print volume
- Content of documents being transformed
- Current utilization of the system and network resources

The throughput of the transform varies depending on items such as:

- Document format created
- Size, density, and complexity of the documents to be transformed
- Resolution of the output documents

In addition, transformed documents are usually larger than the original documents.

PCL to AFP transform

The PCL to AFP transform converts documents in Printer Control Language (PCL) 6 (XL, 5, 5e, 5c) format to AFP format. The transform can accept PCL documents in color. However, it always creates a monochrome AFP image.

The PCL to AFP transform can create MO:DCA IS/3 AFP output. PSF V4.4 or later accepts MO:DCA IS/3 output for printing. For information about MO:DCA IS/3, see *Mixed Object Document Content Architecture Reference*.

The transform creates an AFP Image Object Content Architecture (IOCA) image for each page in the PCL document. The image can be a compressed IOCA image (recommended for faster printing) or an uncompressed image. The transform can produce a complete printable AFP document, or it can create an AFP overlay or page segment of a single page that you can print as part of other documents.

You can specify the type of IOCA image and type of document the transform creates in options on the transform command and in options in the printer definition. The administrator can specify the height and width of the output page and the resolution of the IOCA image in environment variables in the transform configuration file.

Supported fonts

The PCL to AFP transform contains a set of single-byte built-in fonts, which it uses to create a rasterized image of the data. For a list of the fonts that are built in, see "PCL to AFP transform fonts" on page 125. Double-byte character set (DBCS) fonts are not built in to the transform, but the transform can process DBCS fonts that are embedded in the PCL document.

If a PCL document specifies a font that is not built in and is not embedded in the PCL document, the transform substitutes another font that is the best match for the specified font. You cannot customize the way the transform performs font substitution, and the transform does not write any message when it substitutes fonts.

The administrator cannot add fonts to the transform. However, you can embed fonts in the PCL document.

PCL to AFP transform limitations

These limitations apply to the PCL to AFP transform:

- The transform does not produce color output. It produces monochrome output only.
- DBCS fonts are not built in to the transform, but the transform can process DBCS fonts that are embedded in the PCL document.
- PCL data can contain device commands (for example, to begin or end duplexing or to change the input bin). Because the AFP architecture defines those device functions in an AFP form definition, the transform ignores the device commands in the PCL data stream. To obtain these device functions, you must specify them in the form definition or Infoprint Server job attributes.
- Resolution conversion algorithms might produce a degraded appearance when used to reduce the resolution of a data stream. For this reason, the transform might degrade the appearance of higher-resolution data streams when used with 240-pel printers. You should verify that print fidelity is satisfactory.
- Subtle differences exist between PCL4 and PCL5e related to handling fonts. While many PCL4 files work with the transform, some might not produce the expected output.

PDF to AFP transform

The PDF to AFP transform converts documents in PDF 1.7 or earlier levels of PDF format to AFP format. The transform can accept PDF documents in color. However, it always creates a monochrome AFP image.

The PDF to AFP transform can create MO:DCA IS/3-compliant AFP output. PSF for z/OS V4.4 and later lets you print MO:DCA IS/3-compliant AFP output. For information about MO:DCA IS/3-compliant AFP, see *Mixed Object Document Content Architecture Reference*.

The transform creates an AFP Image Object Content Architecture (IOCA) image for each page in the PDF document. The image can be a compressed IOCA image (recommended for faster printing) or an uncompressed image. The transform can produce a printable AFP document, or an AFP overlay or page segment that you can print as part of other documents.

You can specify the type and resolution of the IOCA image, the height and width of the output page, and the type of document the transform creates in options on the transform command and in options in the printer definition.

Supported fonts

The PDF to AFP transform provides a set of fonts, which it uses to create a rasterized image of the data. For a list of the fonts that are provided with the transform, see Table 16 on page 125.

If a PDF document specifies a font that is not embedded in the PDF document and is not provided with the transform, the transform substitutes another font that is the best match for the specified font. You cannot customize the way the transform performs font substitution.

The administrator can add single-byte ASCII fonts to the transform. For information, see “Adding fonts” on page 69.

When the transform substitutes fonts in a document, it writes a message to the transform's `stderr` file and to the transform invoker's `stderr` file so that the administrator can see which fonts the transform has substituted.

PDF to AFP transform limitations

These limitations apply to the PDF to AFP transform:

- The transform does not produce color output. It produces monochrome output only.
- The transform cannot process DBCS fonts.
- The PDF to AFP transform might not be able to transform very large PDF documents that contain non-balanced page trees. For information about page trees, see the Adobe PDF Reference, which is available on the Adobe web site (www.adobe.com).

  Tip: Applications that create PDF documents by combining a large number of separate PDF documents sometimes create PDF documents with non-balanced tree structures.
- Resolution conversion algorithms might produce a degraded appearance when used to reduce the resolution of images imbedded in a data stream. For this reason, the transform might degrade the appearance of higher-resolution images when used with 240-pel printers. You should verify that print fidelity is satisfactory.
- This transform cannot create AFP FS45 image objects, which some color printers require.
- The transform cannot transform PDF files that are concatenated. If you use the AOPBATCH program to run the transform, you must transform each PDF file in a separate step.

PostScript to AFP transform

The PostScript to AFP transform converts documents in PostScript Language Level 3 format to AFP format. The transform can accept PostScript documents in color. However, it always creates a monochrome AFP image.

The transform creates an AFP Image Object Content Architecture (IOCA) image for each page in the PostScript document. The image can be a compressed IOCA image (recommended for faster printing) or an uncompressed image. The transform can produce a printable AFP document, or an AFP overlay or page segment that you can print as part of other documents.

The PostScript to AFP transform can create MO:DCA IS/3-compliant AFP output. PSF V4.4 or later lets you print MO:DCA IS/3-compliant output. For information about MO:DCA IS/3-compliant, see *Mixed Object Document Content Architecture Reference*.

You can specify the type, height, width, and resolution of the IOCA image, and the type of document the transform creates in options on the transform command and in options in the printer definition.

Supported fonts

The PostScript to AFP transform provides a set of fonts, which it uses to create a rasterized image of the data. For a list of the fonts that are provided with the transform, see Table 16 on page 125.

If a PostScript document specifies a font that is not embedded in the PostScript document and is not provided with the transform, the transform substitutes another font that is the best match for the specified font. You cannot customize the way the transform performs font substitution.

The administrator can add single-byte ASCII fonts to the transform. For information, see "Adding fonts" on page 69.

The transform writes a message when it substitutes fonts in a document. The transform writes the font-substitution message to the transform's `stderr file` so that the administrator can see which fonts the transform has substituted.

PostScript to AFP transform limitations

These limitations apply to the PostScript to AFP transform:

- The transform does not produce color output. It produces monochrome output only.
- The transform cannot process DBCS fonts.
- PostScript data can contain device commands (for example, to begin or end duplexing or to change the input bin). Because the AFP architecture defines those device functions in a form definition resource, the transform command ignores the device commands in the print data. To access those device functions, you must specify them in the form definition or attributes file, or on a print command, when you print the job.
- Resolution conversion algorithms might produce a degraded appearance when used to reduce the resolution of images imbedded in a data stream. For this reason, the transform might degrade the appearance of higher-resolution images when used with 240-pel printers. You should verify that print fidelity is satisfactory.
- This transform cannot create AFP FS45 image objects, which some color printers require.
- The transform cannot transform PostScript files that are concatenated. If you use the AOPBATCH program to run the transform, you must transform each PostScript file in a separate step.

SAP to AFP transform

The SAP to AFP transform converts SAP R/3 Release 4.6C Output Text Format (OTF) and Advanced Business Application Programming (ABAP) documents to AFP format:

- SAP R/3 OTF data streams are converted to MO:DCA data streams.
- SAP R/3 ABAP data streams are converted to line data.

The SAP R/3 OTF and ABAP data streams must be in ASCII or Unicode representation.

The transform selects the appropriate AFP resources (form definition, page definition, and font) to use when PSF prints the document. The transform selects the resources based on the value of the PJFORM keyword in the header of the SAP data stream.

The SAP to AFP transform can create MO:DCA IS/3-compliant AFP output. PSF V4.4 or later lets you print MO:DCA IS/3-compliant output. For information about MO:DCA IS/3-compliant, see *Mixed Object Document Content Architecture Reference*.

For information about how to customize SAP to AFP configuration files, see “Customizing SAP to AFP configuration files” on page 76.

Supported fonts

The SAP to AFP transform now supports TrueType fonts. For a list of the fonts that are provided with the transform, see Table 17 on page 126.

Supported input code pages

The SAP to AFP transform supports these input code pages:

- 0000 ASCII ISO-1 (Latin 1)
- 1100 ASCII ISO 8859/1 (Latin 1)
- 1500 Cyrillic ISO 8859/5
- 4001 OCR-A (ASCII)
- 4004 OCR-B (ASCII)
- 4102 ISO ISO/IEC DIS 10646-1.2 Unicode UTF-16BE
- 4103 ISO ISO/IEC DIS 10646-1.2 Unicode UTF-16LE
- 8000 Japanese ISO Shift-JIS
- 8300 Chinese (traditional) ISO Big5
- 8400 Chinese (simplified) ISO GB2312
- 8500 Korean ISO KSC 5601
- 8600 Thai ISO TIS620-2529

For information about the 4-byte codes (such as 1100), see the SAP R/3 4.6C specifications.

OTF BX command support

The SAP to AFP transform supports the OTF BX command. The BX command lets you print boxes that are either clear, solid, or contain four different levels of shading. The levels of shading are a percentage between 0 and 100, with 0 indicating a clear box and 100 a solid box. If the SAP to AFP transform recognizes a BX command, it creates Graphics Object Content Architecture (GOCA) output. GOCA is an AFP data architecture used to represent pictures generated by computer.

If your applications do not require GOCA support features such as box shading, you might want to improve performance by suppressing GOCA support. To suppress GOCA support, specify the -s option on the `sap2afp` command or on the transform filter in the printer definition.

SAP to AFP transform limitations

These limitations apply to the SAP to AFP transform:

- The transform cannot transform SAP R/3 EBCDIC data. It can only transform ASCII or Unicode data.
- If you use the `sap2afp` command to transform ABAP data to line data and you write to an output file, the output file must be an MVS data set, not a UNIX file.

Chapter 2. Using transforms

This information describes the z/OS UNIX transform commands that you can use. It also describes how to use the Infoprint Server subsystem and the `lprafp` command to transform and print documents.

The transform commands let you convert files to an AFP data stream without printing. The transform commands can write the output AFP data stream to a UNIX file or to an MVS data set.

For the levels of the data stream files that the transform commands transform, see "Transform names and levels" on page 2.

These commands and tasks are described:

pcl2afp—Transform PCL data to AFP data

Format

`pcl2afp` [-a *imagetype*] [-c *transformclass*] [-l *length*] [-o *outputfile*]

[-p *pagerange*] [-r *resolution*] [-t *outputtype*] [-w *width*]

[-x *xmargin*] [-y *ymargin*] [-C] [-I] [-T] [*inputfile* ...]

Description

The `pcl2afp` command converts a Printer Control Language (PCL) data file to an AFP data stream file.

If you specify the same option multiple times, the command uses only the last option.

You can specify one or more input files to be transformed. If you do not specify an input file name, or if you specify a dash (`-`) as the file name, `pcl2afp` uses standard input. The output file name is also optional. If you do not specify one, the `pcl2afp` command writes the results to standard output.

Options

Note: All options and values are case-sensitive, except as noted.

-a *imagetype*
Determines the type of AFP data stream image to generate for each page in the PCL file.

Values are:

io1-g4
Compressed Image Object Content Architecture (IOCA) image in Modified Telecommunication Standardization Sector (TSS) T.6 G4 Facsimile Coding Scheme (G4 MMR) format. This is the recommended output type because the AFP output data stream is smaller and it prints faster.

Tips:

1. Some older AFP printers do not support printing with an image type of io1-g4. For these printers, specify an image type of io1-mmr because it is the compressed image type that they support. This image type results in faster printing than uncompressed image types.
2. TSS was formerly the International Telegraph and Telephone Consultative Committee (CCITT).

im1
IM1 image. This type of image is not compressed. Specify it only if you know that your printer does not support compressed images.

Note: Do not create IM1 images at a resolution that the printer does not support because Print Services Facility (PSF) converts the IM1 images to uncompressed IOCA images at the resolution the printer requires. This conversion can cause PSF to use very large amounts of CPU time.

io1
IOCA image. This type of image is not compressed. Specify it only if you know that your printer does not support compressed images.

io1-mmr
Compressed IOCA image in Modified Modified Read (MMR) format.

Default: io1-g4

-c *transformclass*
Specifies the name of a transform class that your administrator has defined. The transform class determines these options:

- The length and width of the generated image
- The page margins
- The resolution of the output image
- The amount of memory that the transform allocates

Ask your administrator for the name of a transform class suitable for the printer and the type of job.

Tip: If the transform class specifies a resolution that the printer does not support, PSF prints the image under most conditions, but with degraded results.

-l *length*

Note:

Specifies the length of the generated image. In general, specify the length of the physical page. For more information about this option, see “Usage notes” on page 18. Specify a number followed by one of these units:

in
Inches

mm
Millimeters

pel
Pels, the default unit

Values are:

0.1334in to 53in
Inch values for 240-pel printers

0.1067in to 53in
Inch values for 300-pel printers

0.0667in to 53in
Inch values for 480-pel printers

0.0534in to 53in
Inch values for 600-pel printers

0.0445in to 53in
Inch values for 720-pel printers

0.0267in to 53in
Inch values for 1200-pel printers

3.3867mm to 1346.2mm
Millimeter values for 240-pel printers

2.7094mm to 1346.2mm
Millimeter values for 300-pel printers

1.6934mm to 1346.2mm
Millimeter values for 480-pel printers

1.3547mm to 1346.2mm
Millimeter values for 600-pel printers

1.1289mm to 1346.2mm
Millimeter values for 720-pel printers

0.6774mm to 1346.2mm
Millimeter values for 1200-pel printers

32pel to 12720pel
Pel values for 240-pel printers

32pel to 15900pel
Pel values for 300-pel printers

32pel to 25440pel
Pel values for 480-pel printers

32pel to 31800pel
Pel values for 600-pel printers

32pel to 31860pel
Pel values for 720-pel printers

32pel to 63600pel
Pel values for 1200-pel printers

Examples:

```
-l 40mm
-l 200.5mm
-l 13in
-l 5280
-l 5280pel
```

Default: The value specified in the AOP_PAGE_HEIGHT environment variable. If the environment variable is not specified, 11in is the default value.

Tips:

1. If a text margin is already set in the file, try `-l 11in` to set the length to 11 inches.
2. Inch values and millimeter values can contain a decimal point; pel values cannot.
3. You can specify the unit using lowercase or uppercase letters (for example, `in` or `IN`).

-o *outputfile*
Specifies the output path and file to which the transform output (that is, AFP data) is written. The transform overwrites any existing data in the output file. If you specify more than one output file, the last path and file name are used. If you do not specify an output file, the result is written to standard output (STDOUT).

To specify an MVS data set, such as a sequential or partitioned data set, precede the data set name with `//`. When you specify a fully qualified name, two sets of quotation marks are required. For example, `"//'hlq.PDS(MYDOC)'"` or `"//'hlq.SEQDS'"`. When you specify a partially qualified name, you only need one set of quotation marks. For example, `"//PDS(MYDOC)"` or `"//SEQDS"`.

If you specify an MVS data set, you must allocate and catalog the data set before you run this command. Allocate the output data set with these characteristics:

- Record format: VBM
- Record length: 8K (8192) bytes or larger

Allocate an MVS data set that is large enough to hold the AFP data stream. The size of the AFP data stream depends on the size and complexity of the document, the type of image compression you select in the -a option, and the resolution specified in the -r option or in the AOP_RESOLUTION environment variable. Typically, an output AFP data stream is several times as large as the input data stream. For information about the size of the AFP data stream, see "Calculating the size of the AFP data stream" on page 50.

Note: If the specified MVS output data set does not exist, the transform creates the data set; however, the data set does not have the correct record format and record length. If you attempt to print the data set, PSF for z/OS writes message APS114I.

-p *pagerange*
Specifies that the output should contain only selected pages.

The -p option counts pages by their actual sequence in the document, not by page number. For example, to write only the last page of a document whose pages are numbered `i, ii, 1, 2, 3, 4`, specify `-p 6`.

Examples of values include:

`-p 1-10`
Write the first through tenth pages.

`-p 10-`
Write pages from the tenth page until the end of the file.

-r *resolution*
Specifies the resolution used to print image data in the job. Select the correct resolution for the printer on which you intend to print the job.

Values are:

240
240 pels per inch (for example, IBM 3812, 3825, 3827, 3835, and 3900 printers)

300
300 pels per inch (for example, IBM 3112, 3116, 4019, 4028, 4029, and 4039 printers and some Hewlett-Packard printers)

480
480 pels per inch

600
600 pels per inch (for example, the InfoPrint 60 and InfoPrint 4000 printers)

720
720 pels per inch (for example, the InfoPrint 5000 printer)

1200
1200 pels per inch (for example, the InfoPrint Pro C900 printer)

Default: 600

Tips:

1. If you specify a resolution that the printer does not support, PSF prints the image under most conditions, but sometimes with degraded results.
2. A resolution of 300 pels typically produces good quality output on 300-, 600-, or 720-pel printers. 300-pel output requires one-fourth the space and transmission time compared to 600-pel output.
3. A resolution of 600 pels typically produces good quality output on 1200-pel printers.

-t *outputtype*
Determines the type of output to create.

Values are:

document
Printable document.

overlay
Graphic image that can be printed on each page of a printable document.

pagesegment
Graphic image that can be embedded in a printable document.

Default: document

Tip:

When you create overlays or page segments from multiple-page documents, use the -p option to select a single page. If you do not select a page, the output contains an overlay or page segment only for the first page because the AFP architecture does not allow multiple, concatenated overlays or page segments.

-w *width*
Specifies the maximum width of the generated image. In general, specify the width of the physical page. For more information about this option, see "Usage notes" on page 18. Specify a number followed by one of these units:

in
Inches

mm
Millimeters

pel
Pels, the default unit

Values are:

0.1334in to 25.5in
Inch values for 240-pel printers

0.1067in to 25.5in
Inch values for 300-pel printers

0.0667in to 25.5in
Inch values for 480-pel printers

0.0534in to 25.5in
Inch values for 600-pel printers

0.0445in to 25.5in
Inch values for 720-pel printers

0.0267in to 25.5in
Inch values for 1200-pel printers

3.3867mm to 647.7mm
Millimeter values for 240-pel printers

2.7094mm to 647.7mm
Millimeter values for 300-pel printers

1.6934mm to 647.7mm
Millimeter values for 480-pel printers

1.3547mm to 647.7mm
Millimeter values for 600-pel printers

1.1289mm to 647.7mm
Millimeter values for 720-pel printers

0.6774mm to 647.7mm
Millimeter values for 1200-pel printers

32pel to 6120pel
Pel values for 240-pel printers

32pel to 7650pel
Pel values for 300-pel printers

32pel to 12240pel
Pel values for 480-pel printers

32pel to 15300pel
Pel values for 600-pel printers

32pel to 18360pel
Pel values for 720-pel printers

32pel to 30600pel
Pel values for 1200-pel printers

Examples:

```
-w 40mm
-w 200.5mm
-w 13in
-w 4000
-w 4000pel
```

Default: The value specified in the AOP_PAGE_WIDTH environment variable. If the environment variable is not specified, 8.5in is the default value.

Tips:

1. If a text margin is already set in the file, try -w 8.5in to set the width to 8.5 inches
2. Inch values and millimeter values can contain a decimal point; pel values cannot.
3. You can specify the unit using lowercase or uppercase letters (for example, `in` or `IN`).

-x *xmargin*
Specifies a horizontal margin or border around the generated image to avoid the non-printable areas of some printers. For more information about this option, see "Usage notes" on page 18. Specify a number followed by one of these units:

in
Inches

mm
Millimeters

pel
Pels, the default unit

Values are:

0in to 12.75in
Inch values for all printers

0mm to 323.85mm
Millimeter values for all printers

0pel to 3060pel
Pel values for 240-pel printers

0pel to 3825pel
Pel values for 300-pel printers

0pel to 6120pel
Pel values for 480-pel printers

0pel to 7650pel
Pel values for 600-pel printers

0pel to 9180pel
Pel values for 720-pel printers

0pel to 15300pel
Pel values for 1200-pel printers

Default: 0.167in

Tips:

1. If the value is less than 1, include a leading zero. For example, specify:

```
0.5in
0.8mm
```

2. Because the X value specifies margins on *both* the left and right sides of the page, the X value can be no more than half of the width (-w) of the generated image. For example, if you specify a width of 8 inches, the X value can be no larger than 4 inches. If you specify an X value of 5 inches, a blank page is printed because the sum of the left and right margins exceeds the width of the paper.
3. The X value does not shift the image on the page. The image is cropped if it is defined to print in the left or right margin.
4. Inch values and millimeter values can contain a decimal point; pel values cannot.
5. You can specify the unit using lowercase or uppercase letters (for example, `in` or `IN`).

-y *ymargin*
Specifies a vertical margin or border around the generated image to avoid the non-printable areas of some printers. For more information about this option, see "Usage notes" on page 18. Specify a number followed by one of these units:

in
Inches

mm
Millimeters

`pel`
Pels, the default unit

Values are:

0in to 26.5in
Inch values for all printers

0mm to 673.1mm
Millimeter values for all printers

0pel to 6360pel
Pel values for 240-pel printers

0pel to 7950pel
Pel values for 300-pel printers

0pel to 12720pel
Pel values for 480-pel printers

0pel to 15900pel
Pel values for 600-pel printers

0pel to 19080pel
Pel values for 720-pel printers

0pel to 31800pel
Pel values for 1200-pel printers

Default: 0.167in

Tips:

1. If the value is less than 1, include a leading zero. For example, specify:

   ```
   0.5in
   0.8mm
   ```

2. Because the Y value specifies margins on *both* the top and bottom of the page, the Y value can be no more than half of the length (-l) of the generated image. For example, if you specify a length of 12 inches, the Y value can be no larger than 6 inches. If you specify a Y value of 7 inches, a blank page is printed because the sum of the top and bottom margins exceeds the length of the paper.
3. The Y value does not shift the image on the page. The image is cropped if it is defined to print in the top or bottom margin.
4. Inch values and millimeter values can contain a decimal point; pel values cannot.
5. You can specify the unit using lowercase or uppercase letters (for example, `in` or `IN`).

-C
Specifies that the transform processes the color PCL files that the PCL 5c driver created to grayscale output. If this option is not specified for transforming PCL 5c files, the resulting AFP file might not print correctly.

-I
Specifies that the transform produces MO:DCA IS/3-compliant output.

Default: If the -I option is not set, the default is MO:DCA IS/1 output.

This functionality is only available when a single file is sent to the transform, not several at a time.

Note: Select this option only if the level of PSF for z/OS you are using can process MO:DCA IS/3-compliant output.

-T
Requests a trace. Specify this option only if instructed by IBM service personnel. For more information, see Chapter 5, “Diagnosing errors,” on page 105.

Tip: You can use the filter-options job attribute with, for example, the `lp` command to pass any of these options except -o *outputfile* to the transform. For information about the filter-options job attribute, see *z/OS Infoprint Server User's Guide*.

Operand

inputfile
Specifies an input file to be transformed. If you specify more than one input file name, the `pcl2afp` command transforms each file to AFP format and writes the results to a single output file (if one is specified) or to standard output.

If you do not specify an input file, or if you specify a dash (`-`) as the file name, `pcl2afp` uses standard input.

Usage notes

In general, to position data on the page:

- Use -l and -w to set the physical page dimensions.
- Use -x and -y to set the amount of white space between the physical page dimensions and the image.

  These options do not shift or scale the image on the page. If the image is defined to print in the unprintable areas, it is cropped.

  For example, to create a 6.5 x 9 inch image that is centered on an 8.5 x 11 inch page, enter:

  ```
  -l 11in -w 8.5in -x 1in -y 1in
  ```

- Use a form definition that specifies zero vertical offset and zero horizontal offset, or specify X and Y offsets of 0 when you submit the print job.

Examples -- pcl2afp

Transform a file, specifying transform class

To transform the PCL file `myfile.pcl` to an AFP data stream, using the `a4_300` transform class, and write the result to a file called `myfile.afp`, enter:

```
pcl2afp -c a4_300 -o myfile.afp myfile.pcl
```

Transform and print a file, specifying image type

To transform the PCL file `myfile.pcl` to an AFP data stream as an IO1-MMR image, and send the result to the default printer with the `lp` command, enter:

```
pcl2afp -a io1-mmr myfile.pcl | lp
```

Transform a file using redirection

To transform the PCL file `input.pcl` to the AFP output file called `output.afp`, enter:

```
pcl2afp < input.pcl > output.afp
```

Note: You can use redirection operators only with UNIX files.

Transform multiple files

To transform the PCL files `input.01.pcl`, `input.02.pcl`, ... `input.xx.pcl` and write the results to one AFP output file called `output.afp`, enter:

```
pcl2afp -o output.afp input.01.pcl input.02.pcl ... input.xx.pcl
```

Transform a UNIX file to an MVS data set

To transform file `input.pcl` to an existing, cataloged MVS output data set called *hlq*.OUTPUT.AFP(*member*), where *hlq* is your user ID, enter:

```
pcl2afp -o "//'hlq.OUTPUT.AFP(member)'" input.pcl
```

Transform an MVS data set, writing the output to a UNIX file

To transform the MVS data set *hlq*.INPUT.PCL(*member*), where *hlq* is your user ID, to an output file called `output.afp`, enter:

```
pcl2afp -o output.afp "//'hlq.INPUT.PCL(member)'"
```

Transform a file and receive MVS return codes

To transform the PCL file `myfile.pcl` to an AFP data stream and write the result to a file called `myfile.afp` and receive MVS return codes (0, 4, 8), enter:

```
AOP_MVS_RETURN_CODES=YES pcl2afp -o myfile.afp myfile.pcl
```

MVS return codes indicate whether the transform was successful (0), a transform warning occurred (4), or a transform error occurred (8).

Transform a file to MO:DCA IS/3 compliant-format

To transform the PCL file `myfile.pcl` into an MO:DCA IS/3 AFP data stream and write the result to a file called `myfile.afp` enter:

```
pcl2afp -I  -o myfile.afp myfile.pcl
```

Transform a file, specifying resolution

To transform the PCL file `myfile.pcl` into a 1200 pels per inch file called `myfile.afp`, enter:

```
pcl2afp -r 1200 -o myfile.afp myfile.pcl
```

Process color files into grayscale output

To transform the PCL 5c file `myfile.pcl` into a grayscale AFP data stream, and write the result to a file called `myfile.afp`, enter:

```
pcl2afp -C -o myfile.afp myfile.pcl
```

Environment variables

The `pcl2afp` command uses these environment variables:

AOPCONF
Names the Infoprint Server configuration file. This variable takes precedence over the user-specific configuration file (`$HOME/.aopconf`) and the system default configuration file (`/etc/Printsrv/aopd.conf`). For more information about the configuration file, see *z/OS Infoprint Server Customization*.

AOP_MVS_RETURN_CODES
Specifies whether the command returns MVS return codes (0, 4, 8) or UNIX exit values (0, 1). The default is UNIX exit values.

LIBPATH
The path used to locate dynamic link libraries (DLLs).

NLSPATH
Names the directory paths that the `pcl2afp` command searches for message catalogs.

For information about setting and using environment variables, see *z/OS UNIX System Services User's Guide*.

The PCL to AFP transform uses environment variables specified in the Infoprint Server transform configuration file. For information, see "Environment variables for the PCL to AFP transform" on page 55.

Files

`$HOME/.aopconf`
Contains the user-specific Infoprint Server configuration file. This file takes precedence over `/etc/Printsrv/aopd.conf`.

/etc/Printsrv/aopd.conf
Contains the system default Infoprint Server configuration file.

For the format of the configuration files, see *z/OS Infoprint Server Customization*.

Exit values

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.

UNIX exit values (default):

0
Successful. However, a warning or data stream error might have occurred during the transform.

1
The command was not accepted, or the transform failed.

Notes:

1. The AOP_MVS_RETURN_CODES environment variable determines the type of exit values.
2. The AOP_FAIL_ON_ERROR environment variable controls whether the transform fails when it detects a data stream warning or error.

pdf2afp—Transform PDF data to AFP data

Format

pdf2afp [-a imagetype] [-c *transformclass*] [-l *length*] [-n *htvalue*]

[-o *outputfile*] [-p *pagerange*] [-r *resolution*] [-t *outputtype*]

[-w *width*] [-x *xmargin*] [-y *ymargin*] [-I] [-T] [*inputfile...*]

Description

The pdf2afp command converts a Portable Document Format (PDF) data file to an AFP data stream file.

If you specify the same option multiple times, the command uses only the last option.

You can specify one or more input files to be transformed. If you do not specify an input file name, or if you specify a dash (-) as the file name, pdf2afp uses standard input. The output file name is also optional. If you do not specify one, the pdf2afp command writes the results to standard output.

Options

Note: All options and values are case-sensitive, except as noted.

-a *imagetype*
Determines the type of AFP data stream image to generate for each page in the PDF file.

Values are:

io1-g4
Compressed Image Object Content Architecture (IOCA) image in Modified Telecommunication Standardization Sector (TSS) T.6 G4 Facsimile Coding Scheme (G4 MMR) format. The

recommended output type is iol-g4 because the AFP output data stream is smaller and it prints faster.

Tips:

1. Some older AFP printers do not support printing with an image type of io1-g4. For these printers, specify an image type of io1-mmr because it is the compressed image type that they support. This image type results in faster printing than uncompressed image types.
2. TSS was formerly the International Telegraph and Telephone Consultative Committee (CCITT).

im1
IM1 image. This type of image is not compressed. Specify it only if you know that your printer does not support compressed images.

Note: Do not create IM1 images at a resolution that the printer does not support because Print Services Facility (PSF) converts the IM1 images to uncompressed IOCA images at the resolution the printer requires. This conversion can cause PSF to use very large amounts of CPU time.

io1
IOCA image. This type of image is not compressed. Specify it only if you know that your printer does not support compressed images.

io1-mmr
Compressed IOCA image in Modified Modified Read (MMR) format.

Default: io1-g4

-c *transformclass*
Specifies the name of a transform class defined in the transform configuration file. The transform class determines these options:

- The initial transform configuration
- The fonts used in the transformed files

Ask the administrator for the name of a transform class suitable for the type of job.

-l *length*
Specifies the length of the generated image. In general, specify the length of the physical page. For more information about this option, see “Usage notes” on page 28. Specify a number followed by one of these units:

in
Inches

mm
Millimeters

pel
Pels, the default unit

Values are:

0.1334in to 53in
Inch values for 240-pel printers

0.1067in to 53in
Inch values for 300-pel printers

0.0667in to 53in
Inch values for 480-pel printers

0.0534in to 53in
Inch values for 600-pel printers

3.3867mm to 1346.2mm
Millimeter values for 240-pel printers

2.7094mm to 1346.2mm
Millimeter values for 300-pel printers

1.6934mm to 1346.2mm
Millimeter values for 480-pel printers

1.3547mm to 1346.2mm
Millimeter values for 600-pel printers

32pel to 12720pel
Pel values for 240-pel printers

32pel to 15900pel
Pel values for 300-pel printers

32pel to 25440pel
Pel values for 480-pel printers

32pel to 31800pel
Pel values for 600-pel printers

Examples:

```
-l 40mm
-l 200.5mm
-l 13in
-l 5280
-l 5280pel
```

Default: The length value set in the file is used. If none is set, 11in is the default.

Tips:

1. If a text margin is already built into the file, try `-l 11in` to set the length to 11 inches.
2. Inch values and millimeter values can contain a decimal point; pel values cannot.
3. You can specify the unit using lowercase or uppercase letters (for example, `in` or `IN`).

-n *htvalue*
Tunes the output by setting a threshold of color range from which grays and colors are rendered in black instead of halftone. The smaller the value, the more grays and colors are rendered in halftones; the larger the value, the more grays and colors are rendered in blacks.

Values are 0 - 1.

Examples:

```
-n 0.2
-n 0.8
-n 1
```

Default: 0

Tip: This parameter is generally unnecessary unless you have problems with the way halftoned output is displayed.

-o *outputfile*
Specifies the output path and file into which the transform output (that is, AFP data) is written. The transform overwrites any existing data in the output file. If you specify more than one output file, the last path and file name are used. If you do not specify an output file, the result is written to standard output (STDOUT).

To specify an MVS data set, such as a sequential or partitioned data set, precede the data set name with `//`. When you specify a fully qualified name, two sets of quotation marks are required. For example, `"//'hlq.PDS(MYDOC)'"` or `"//'hlq.SEQDS'"`. When you specify a partially qualified name, you only need one set of quotation marks. For example, `"//PDS(MYDOC)"` or `"//SEQDS"`.

If you specify an MVS data set, you must allocate and catalog the data set before you run this command. Allocate the output data set with these characteristics:

- Record format: VBM
- Record length: 8K (8192) bytes or larger

Allocate an MVS data set that is large enough to hold the AFP data stream. The size of the AFP data stream depends on the size and complexity of the document, the type of image compression (-a option), and the resolution of the image (-r option). Typically, an output AFP data stream is several times as large as the input data stream. For information about the size of the AFP data stream, see "Calculating the size of the AFP data stream" on page 50.

Note: If the specified MVS output data set does not exist, the transform creates the data set; however, the data set does not have the correct record format and record length. If you attempt to print the data set, PSF for z/OS writes message APS114I.

-p *pagerange*
Specifies that the output should contain only selected pages.

Examples of values include:

`-p 1-10`
Write the first through tenth pages.

`-p 10-`
Write pages from the tenth page until the end of the job.

-r *resolution*
Specifies the resolution of the output image. Select the correct resolution for the printer on which you intend to print the image.

Values are:

240
240 pels per inch (for example, IBM 3812, 3825, 3827, 3835, and 3900 printers)

300
300 pels per inch (for example, IBM 4019, 4028, 4029, and 4039 printers and some Hewlett-Packard printers)

360
360 pels per inch

480
480 pels per inch

600
600 pels per inch (for example, InfoPrint 60 and InfoPrint 4000 printers)

720
720 pels per inch

1200
1200 pels per inch

Default: 600

Tips:

1. If you specify a resolution that the printer does not support, PSF prints the image under most conditions, but sometimes with degraded results.
2. A resolution of 300 pels typically produces good quality output on 300-, 600-, or 720-pel printers.

   300-pel output requires one-fourth the space and transmission time compared to 600-pel output.
3. A resolution of 600 pels typically produces good quality output on 1200-pel printers.

-t *outputtype*
Determines the type of output to create.

Values are:

document
Printable document.

overlay
Graphic image that can be printed on each page of a printable document.

pagesegment
Graphic image that can be embedded in a printable document.

Default: document

Tip:

When you create overlays or page segments from multiple-page documents, use the -p option to select a single page. If you do not select a page, the output contains an overlay or page segment only for the first page because the AFP architecture does not allow multiple, concatenated overlays or page segments.

-w ***width***
Specifies the maximum width of the generated image. In general, specify the width of the physical page. For more information about this option, see "Usage notes" on page 28. Specify a number followed by one of these units:

in
Inches

mm
Millimeters

pel
Pels, the default unit

Values are:

0.1334in to 25.5in
Inch values for 240-pel printers

0.1067in to 25.5in
Inch values for 300-pel printers

0.0667in to 25.5in
Inch values for 480-pel printers

0.0534in to 25.5in
Inch values for 600-pel printers

3.3867mm to 647.7mm
Millimeter values for 240-pel printers

2.7094mm to 647.7mm
Millimeter values for 300-pel printers

1.6934mm to 647.7mm
Millimeter values for 480-pel printers

1.3547mm to 647.7mm
Millimeter values for 600-pel printers

32pel to 6120pel
Pel values for 240-pel printers

32pel to 7650pel
Pel values for 300-pel printers

32pel to 12240pel
Pel values for 480-pel printers

32pel to 15300pel
Pel values for 600-pel printers

Examples:

```
-w 40mm
-w 200.5mm
-w 13in
```

```
-w 4000
-w 4000pel
```

Default: The width set in the file is used. If none is set, the default is 8.5in.

Tips:

1. If a text margin is already built into the file, try `-w 8.5in` to set the width to 8.5 inches.
2. Inch values and millimeter values can contain a decimal point; pel values cannot.
3. You can specify the unit using lowercase or uppercase letters (for example, `in` or `IN`).

-x *xmargin*

Specifies a horizontal margin or border around the generated image to avoid the non-printable areas of some printers. For more information about this option, see "Usage notes" on page 28. Specify a number followed by one of these units:

in
Inches

mm
Millimeters

pel
Pels, the default unit

Values are:

0
Zero

0in to 12.75in
Inch values for all printers

0mm to 323.85mm
Millimeter values for all printers

0pel to 3060pel
Pel values for 240-pel printers

0pel to 3825pel
Pel values for 300-pel printers

0pel to 6120pel
Pel values for 480-pel printers

0pel to 7650pel
Pel values for 600-pel printers

Default: 0

Tips:

1. If the value is less than 1, include a leading zero. For example, specify:

   ```
   0.5in
   0.8mm
   ```

2. Because the X value specifies margins on *both* the left and right sides of the page, the X value can be no more than half of the width (-w) of the generated image. For example, if you specify a width of 8 inches, the X value can be no larger than 4 inches. If you specify an X value of 5 inches, a blank page is printed because the sum of the left and right margins exceeds the width of the paper.
3. The X value does not shift the image on the page. The image is cropped if it is defined to print in the left or right margin.
4. Inch values and millimeter values can contain a decimal point; p values cannot.
5. You can specify the unit using lowercase or uppercase letters (for example, `in` or `IN`).

-y *ymargin*

Specifies a vertical margin or border around the generated image to avoid the non-printable areas of some printers. For more information about this option, see "Usage notes" on page 28. Specify a number followed by one of these units:

in
Inches

mm
Millimeters

pel
Pels, the default unit

Values are:

0
Zero

0in to 26.5in
Inch values for all printers

0mm to 673.1mm
Millimeter values for all printers

0pel to 6360pel
Pel values for 240-pel printers

0pel to 7950pel
Pel values for 300-pel printers

0pel to 12720pel
Pel values for 480-pel printers

0pel to 15900pel
Pel values for 600-pel printers

Default: 0

Tips:

1. If the value is less than 1, include a leading zero. For example, specify:

```
0.5in
0.8mm
```

2. Because the Y value specifies margins on *both* the top and bottom of the page, the Y value can be no more than half of the length (-l) of the generated image. For example, if you specify a length of 12 inches, the Y value can be no larger than 6 inches. If you specify a Y value of 7 inches, a blank page is printed because the sum of the top and bottom margins exceeds the length of the paper.
3. The Y value does not shift the image on the page. The image is cropped if it is defined to print in the top or bottom margin.
4. Inch values and millimeter values can contain a decimal point; pel values cannot.
5. You can specify the unit using lowercase or uppercase letters (for example, `in` or `IN`).

-I

Specifies that the transform produces MO:DCA IS/3-compliant data streams.

This functionality is only available when a single file is sent to the transform, not several at a time.

Note:

Select this option only if the level of PSF for z/OS you are using can process MO:DCA IS/3-compliant output.

Default: If the -I option is not specified, the default is MO:DCA IS/1 output.

-T
Requests a trace. Specify this option only if instructed by IBM service personnel. For more information, see Chapter 5, “Diagnosing errors,” on page 105.

Tip: You can use the filter-options job attribute with, for example, the `lp` command to pass any of these options except -o *outputfile* to the transform. For information about the filter-options job attribute, see *z/OS Infoprint Server User's Guide*.

Operand

inputfile
Specifies an input file to be transformed. If you specify more than one input file name, the `pdf2afp` command transforms each file to AFP format and writes the results to a single output file (if one is specified) or to standard output.

If you do not specify an input file, or if you specify a dash (`-`) as the file name, `pdf2afp` uses standard input.

Usage notes

In general, to position data on the page:

- Use -l and -w to set the physical page dimensions.
- Use -x and -y to set the amount of white space between the physical page dimensions and the image.

  These options do not shift or scale the image on the page. If the image is defined to print in the unprintable areas, it is cropped.

  For example, to create a 6.5 x 9 inch image that is centered on an 8.5 x 11 inch page, enter:

  ```
  -l 11in -w 8.5in -x 1in -y 1in
  ```

- Use a form definition that specifies zero vertical offset and zero horizontal offset, or specify X and Y offsets of 0 when you submit the print job.

Examples -- pdf2afp

Transform and print a file, specifying image type

To transform the PDF `myfile1.pdf` file to an AFP data stream as an IO1-MMR image, and send the result to the default printer with the `lp` command, enter:

```
pdf2afp -a io1-mmr myfile1.pdf | lp
```

Transform and print a file, specifying resolution

To transform the PDF file `myfile1.pdf` to an AFP data stream, and then submit it to the 4019 printer called `robin`, enter:

```
pdf2afp -r 300 myfile1.pdf | lp -d robin
```

Tip: You need to specify a resolution of 300 pels (`-r 300`) because the 4019 is a 300-pel resolution printer. The default resolution for the `pdf2afp` command is 600 pels.

Transform and print a file, specifying image type and resolution

To transform the PDF `myfile1.pdf` file to an AFP data stream in 300-pel resolution, as an IO1-MMR image, and send the result to the default printer with the `lp` command, enter:

```
pdf2afp -a io1-mmr -r 300 myfile1.pdf | lp
```

Transform a UNIX file to an MVS data set

To transform file `input.pdf` to an existing, cataloged MVS output data set called *`hlq`*`.OUTPUT.AFP(`*`member`*`)`, where *hlq* is your user ID, enter:

```
pdf2afp -o "//'hlq.OUTPUT.AFP(member)'" input.pdf
```

Transform a file and receive MVS return codes

To transform the PDF file `myfile.pdf` to an AFP data stream and write the result to a file called `myfile.afp` and receive MVS return codes (0, 4, 8), enter:

```
AOP_MVS_RETURN_CODES=YES pdf2afp -o myfile.afp myfile.pdf
```

MVS return codes indicate whether the transform was successful (0), a transform warning occurred (4), or a transform error occurred (8).

Transform a file to MO:DCA IS/3 compliant-format

To transform the PDF file `myfile.pdf` into an MO:DCA IS/3 AFP data stream and write the result to a file called `myfile.afp` enter:

```
pdf2afp -I -o myfile.afp myfile.pdf
```

Environment variables

The `pdf2afp` command uses these environment variables:

AOPCONF
Names the Infoprint Server configuration file. This variable takes precedence over the user-specific configuration file (`$HOME/.aopconf`) and the system default configuration file (`/etc/Printsrv/aopd.conf`). For more information about the configuration file, see *z/OS Infoprint Server Customization*.

AOP_MVS_RETURN_CODES
Specifies whether the command returns MVS return codes (0, 4, 8) or UNIX (0, 1) exit values. The default is UNIX exit values.

LIBPATH
The path used to locate dynamic link libraries (DLLs).

NLSPATH
Names the directory paths that the `pdf2afp` command searches for message catalogs.

For information about setting and using environment variables, see *z/OS UNIX System Services User's Guide*.

The PDF to AFP transform uses environment variables specified in the Infoprint Server transform configuration file. For information, see "Environment variables for the PDF to AFP and PostScript to AFP transforms" on page 62.

Files

$HOME/.aopconf
Contains the user-specific Infoprint Server configuration file. This file takes precedence over `/etc/Printsrv/aopd.conf`.

/etc/Printsrv/aopd.conf
Contains the system default Infoprint Server configuration file.

For the format of the configuration files, see *z/OS Infoprint Server Customization*.

Exit values

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.

UNIX exit values (default):

0
Successful. However, a warning or data stream error might have occurred during the transform.

1
The command was not accepted, or the transform failed.

Notes:

1. The AOP_MVS_RETURN_CODES environment variable determines the type of exit values.
2. The AOP_FAIL_ON_ERROR transform environment variable controls whether the transform fails when it detects a data stream warning or error.

ps2afp—Transform PostScript data to AFP data

Format

`ps2afp` [-a *imagetype*] [-c *transformclass*] [-l *length*] [-n *htvalue*]

[-o *outputfile*] [-r *resolution*] [-t *outputtype*] [-w *width*]

[-x *xmargin*] [-y *ymargin*] [-I] [-T] [*inputfile ...*]

Description

The `ps2afp` command converts a PostScript data file into an AFP data stream file.

If you specify the same option multiple times, the command uses only the last option. You can specify one or more input files to be transformed. If you do not specify an input file name, or if you specify a dash (-) as the file name, `ps2afp` uses standard input. The output file name is also optional. If you do not specify one, the `ps2afp` command writes the results to standard output.

Options

Note: All options and values are case-sensitive, except as noted.

-a *imagetype*
Determines the type of AFP data stream image to generate for each page in the PostScript file.

Values are:

io1-g4
Compressed Image Object Content Architecture (IOCA) image in Modified Telecommunication Standardization Sector (TSS) T.6 G4 Facsimile Coding Scheme (G4 MMR) format. This is the recommended output type because the AFP output data stream is smaller and it prints faster.

Tips:

1. Some older AFP printers do not support printing with an image type of io1-g4. For these printers, specify an image type of io1-mmr because it is the compressed image type that they support. This image type results in faster printing than uncompressed image types.
2. TSS was formerly the International Telegraph and Telephone Consultative Committee (CCITT).

im1
IM1 image. This type of image is not compressed. Specify it only if you know that your printer does not support compressed images.

Note: Do not create IM1 images at a resolution that the printer does not support because Print Services Facility (PSF) converts the IM1 images to uncompressed IOCA images at the resolution the printer requires. This conversion can cause PSF to use very large amounts of CPU time.

io1
IOCA image. This type of image is not compressed. Specify it only if you know that your printer does not support compressed images.

io1-mmr
Compressed IOCA image in Modified Modified Read (MMR) format.

Default: io1–g4

-c *transformclass*
Specifies the name of a transform class that your administrator has defined. The transform class determines these options:

- The initial transform configuration
- The fonts used in the transformed files

Ask your administrator for the name of a transform class suitable for the type of job.

-l *length*
Specifies the length of the generated image. In general, specify the length of the physical page. For more information about this option, see “Usage notes” on page 36. Specify a number followed by one of these units:

in
Inches

mm
Millimeters

pel
Pels, the default unit

Values are:

0.1334in to 53in
Inch values for 240-pel printers

0.1067in to 53in
Inch values for 300-pel printers

0.0667in to 53in
Inch values for 480-pel printers

0.0534in to 53in
Inch values for 600-pel printers

3.3867mm to 1346.2mm
Millimeter values for 240-pel printers

2.7094mm to 1346.2mm
Millimeter values for 300-pel printers

1.6934mm to 1346.2mm
Millimeter values for 480-pel printers

1.3547mm to 1346.2mm
Millimeter values for 600-pel printers

32pel to 12720pel
Pel values for 240-pel printers

32pel to 15900pel
Pel values for 300-pel printers

32pel to 25440pel
Pel values for 480-pel printers

32pel to 31800pel
Pel values for 600-pel printers

Examples:

```
-l 40mm
-l 200.5mm
-l 13in
-l 5280
-l 5280pel
```

Default: The length value set in the file is used. If none is set, 11in is the default.

Tips:

1. If a text margin is already built into the file, try `-l 11in` to set the length to 11 inches.
2. Inch values and millimeter values can contain a decimal point; pel values cannot.
3. You can specify the unit using lowercase or uppercase letters (for example, `in` or `IN`).

-n *htvalue*
Tunes the output by setting a threshold of color range from which grays and colors are rendered in black instead of halftone. The smaller the value, the more grays and colors are rendered in halftones; the larger the value, the more grays and colors are rendered in blacks.

Values are 0 - 1.

Examples:

```
-n 0.2
-n 0.8
-n 1
```

Default: 0

Tip: This parameter is generally unnecessary unless you have problems with the way halftoned output is displayed.

-o *outputfile*
Specifies the output path and file into which the transform output (that is, AFP data) is written. The transform overwrites any existing data in the output file. If you specify more than one output file, the last path and file name are used. If you do not specify an output file, the result is written to standard output (STDOUT).

To specify an MVS data set, such as a sequential or partitioned data set, precede the data set name with `//`. When you specify a fully qualified name, two sets of quotation marks are required. For example, `"//'hlq.PDS(MYDOC)'"` or `"//'hlq.SEQDS'"`. When you specify a partially qualified name, you only need one set of quotation marks. For example, `"//PDS(MYDOC)"` or `"//SEQDS"`.

If you specify an MVS data set, you must allocate and catalog the data set before you run this command. Allocate the output data set with these characteristics:

- Record format: VBM
- Record length: 8K (8192) bytes or larger

Allocate an MVS data set that is large enough to hold the AFP data stream. The size of the AFP data stream depends on the size and complexity of the document, the type of image compression (-a option), and the resolution of the image (-r option). Typically, an output AFP data stream is several times as large as the input data stream. For information about the size of the AFP data stream, see "Calculating the size of the AFP data stream" on page 50.

Note: If the specified MVS output data set does not exist, the transform creates the data set; however, the data set does not have the correct record format and record length. If you attempt to print the data set, PSF for z/OS writes message APS114I.

-r *resolution*

Specifies the resolution of the output image. Select the correct resolution for the printer on which you intend to print the image.

Values are:

240
240 pels per inch (for example, IBM 3812, 3825, 3827, 3835, and 3900 printers)

300
300 pels per inch (for example, IBM 4019, 4028, 4029, and 4039 printers and some Hewlett-Packard printers)

360
360 pels per inch

480
480 pels per inch

600
600 pels per inch (for example, InfoPrint 60 and InfoPrint 4000 printers)

720
720 pels per inch

1200
1200 pels per inch

Default: 600

Tips:

1. If you specify a resolution that the printer does not support, PSF prints the image under most conditions, but sometimes with degraded results.
2. A resolution of 300 pels typically produces good quality output on 300-, 600-, or 720-pel printers.

   300-pel output requires one-fourth the space and transmission time compared to 600-pel output.
3. A resolution of 600 pels typically produces good quality output on 1200-pel printers.

-t *outputtype*

Determines the type of output to create.

Values are:

document
Printable document.

overlay
Graphic image that can be printed on each page of a printable document.

pagesegment
Graphic image that can be embedded in a printable document.

Default: document

Tip: When you create an overlay or page segment from a multiple-page input file, use the -p option to select a single page. If you do not select a page, the output will contain multiple, concatenated overlays or page segments. The AFP architecture does not allow multiple, concatenated overlays or page segments.

-w *width*

Specifies the maximum width of the generated image. In general, specify the width of the physical page. For more information about this option, see "Usage notes" on page 36. Specify a number followed by one of these units:

in
Inches

mm
Millimeters

pel
Pels, the default unit

Values are:

0.1334in to 25.5in
Inch values for 240-pel printers

0.1067in to 25.5in
Inch values for 300-pel printers

0.0667in to 25.5in
Inch values for 480-pel printers

0.0534in to 25.5in
Inch values for 600-pel printers

3.3867mm to 647.7mm
Millimeter values for 240-pel printers

2.7094mm to 647.7mm
Millimeter values for 300-pel printers

1.6934mm to 647.7mm
Millimeter values for 480-pel printers

1.3547mm to 647.7mm
Millimeter values for 600-pel printers

32pel to 6120pel
Pel values for 240-pel printers

32pel to 7650pel
Pel values for 300-pel printers

32pel to 12240pel
Pel values for 480-pel printers

32pel to 15300pel
Pel values for 600-pel printers

Examples:

```
-w 40mm
-w 200.5mm
-w 13in
-w 4000
-w 4000pel
```

Default: The width set in the file is used. If none is set, the default is 8.5in.

Tips:

1. If a text margin is already built into the file, try `-w 8.5in` to set the width to 8.5 inches.
2. Inch values and millimeter values can contain a decimal point; pel values cannot.

3. You can specify the unit using lowercase or uppercase letters (for example, `in` or `IN`).

-x *xmargin*
Specifies a horizontal margin or border around the generated image to avoid the non-printable areas of some printers. For more information about this option, see "Usage notes" on page 36. Specify a number followed by one of these units:

in
Inches

mm
Millimeters

pel
Pels, the default unit

Values are:

0in to 12.75in
Inch values for all printers

0mm to 323.85mm
Millimeter values for all printers

0pel to 3060pel
Pel values for 240-pel printers

0pel to 3825pel
Pel values for 300-pel printers

0pel to 6120pel
Pel values for 480-pel printers

0pel to 7650pel
Pel values for 600-pel printers

Default: 0

Tips:

1. If the value is less than 1, include a leading zero. For example, specify:

```
0.5in
0.8mm
```

2. Because the X value specifies margins on *both* the left and right sides of the page, the X value can be no more than half of the width (-w) of the generated image. For example, if you specify a width of 8 inches, the X value can be no larger than 4 inches. If you specify an X value of 5 inches, a blank page is printed because the sum of the left and right margins exceeds the width of the paper.
3. The X value does not shift the image on the page. The image is cropped if it is defined to print in the left or right margin.
4. Inch values and millimeter values can contain a decimal point; pel values cannot.
5. You can specify the unit using lowercase or uppercase letters (for example, `in` or `IN`).

-y *ymargin*
Specifies a vertical margin or border around the generated image to avoid the non-printable areas of some printers. For more information about this option, see "Usage notes" on page 36. Specify a number followed by one of these units:

in
Inches

mm
Millimeters

pel
Pels, the default unit

Values are:

0in to 26.5in
Inch values for all printers

0mm to 673.1mm
Millimeter values for all printers

0pel to 6360pel
Pel values for 240-pel printers

0pel to 7950pel
Pel values for 300-pel printers

0pel to 12720pel
Pel values for 480-pel printers

0pel to 15900pel
Pel values for 600-pel printers

Default: 0

Tips:

1. If the value is less than 1, include a leading zero. For example, specify:

```
0.5in
0.8mm
```

2. Because the Y value specifies margins on *both* the top and bottom of the page, the Y value can be no more than half of the length (-l) of the generated image. For example, if you specify a length of 12 inches, the Y value can be no larger than 6 inches. If you specify a Y value of 7 inches, a blank page is printed because the sum of the top and bottom margins exceeds the length of the paper.
3. The Y value does not shift the image on the page. The image is cropped if it is defined to print in the top or bottom margin.
4. Inch values and millimeter values can contain a decimal point; pel values cannot.
5. You can specify the unit using lowercase or uppercase letters (for example, `in` or `IN`).

-I
Specifies that the transform produces MO:DCA IS/3-compliant data streams.

This functionality is only available when a single file is sent to the transform, not several at a time.

Default: If the -I option is not set, the default is MO:DCA IS/1 output.

-T
Requests a trace. Specify this option only if instructed by IBM service personnel. For more information, see Chapter 5, “Diagnosing errors,” on page 105.

Tip: You can use the filter-options job attribute with, for example, the `lp` command to pass any of these options except -o *outputfile* to the transform. For information about the filter-options job attribute, see *z/OS Infoprint Server User's Guide*.

Operand

inputfile
Specifies an input file to be transformed. If you specify more than one input file name, the `ps2afp` command transforms each file to AFP format and writes the results to a single output file (if one is specified) or to standard output.

If you do not specify an input file, or if you specify a dash (`-`) as the file name, `ps2afp` uses standard input.

Usage notes

In general, to position data on the page:

- Use -l and -w to set the physical page dimensions.

- Use -x and -y to set the amount of white space between the physical page dimensions and the image.

  These options do not shift or scale the image on the page. If the image is defined to print in the unprintable areas, it is cropped.

  For example, to create a 6.5 x 9 inch image that is centered on an 8.5 x 11 inch page, enter:

  ```
  -l 11in -w 8.5in -x 1in -y 1in
  ```

- Use a form definition that specifies zero vertical offset and zero horizontal offset, or specify X and Y offsets of 0 when you submit the print job.

Examples -- ps2afp

Transform a file, specifying transform class

To transform the PostScript file `myfile.ps` into an AFP data stream, using the `bigjob` transform class, and write the result to a file called `myfile.afp`, enter:

```
ps2afp -c bigjob -o myfile.afp myfile.ps
```

Transform a file, specifying image size

To transform the PostScript file `myfile2.ps` into an AFP data stream, with an image that is 8 inches high and 5 inches wide, and write the result to a file called `myfile2.afp`, enter:

```
ps2afp -l 8in -w 5in -o myfile2.afp myfile2.ps
```

Transform a file, using redirection

To transform the PostScript file `input.ps` into the AFP output file called `output.afp`, enter:

```
ps2afp < input.ps > output.afp
```

Note: You can only use redirection operators with z/OS UNIX files.

Transform multiple files

To transform the PostScript files `input.01.ps`, `input.02.ps`, ... `input.xx.ps` and write the results to one AFP output file called `output.afp`, enter:

```
ps2afp -o output.afp input.01.ps input.02.ps ... input.xx.ps
```

Transform a UNIX file to an MVS data set

To transform file `input.ps` into an existing, cataloged MVS output data set called *`hlq`*`.OUTPUT.AFP(`*`member`*`)`, where *hlq* is your user ID, enter:

```
ps2afp -o "//'hlq.OUTPUT.AFP(member)'" input.ps
```

Transform a file and receive MVS return codes

To transform the PostScript file `myfile.ps` into an AFP data stream and write the result to a file called `myfile.afp` and receive MVS return codes (0, 4, 8), enter:

```
AOP_MVS_RETURN_CODES=YES ps2afp -o myfile.afp myfile.ps
```

MVS return codes indicate whether the transform was successful (0), a transform warning occurred (4), or a transform error occurred (8).

Transform a file to MO:DCA IS/3 compliant-format

To transform the PostScript file `myfile.ps` into an MO:DCA IS/3 AFP data stream and write the result to a file called `myfile.afp` enter:

```
ps2afp -I -o myfile.afp myfile.ps
```

Environment variables

The `ps2afp` command uses these environment variables:

AOPCONF
Names the Infoprint Server configuration file. This variable takes precedence over the user-specific configuration file (`$HOME/.aopconf`) and the system default configuration file (`/etc/Printsrv/aopd.conf`). For more information about the configuration file, see *z/OS Infoprint Server Customization*.

AOP_MVS_RETURN_CODES
Specifies whether the command returns MVS return codes (0, 4, 8) or UNIX exit values (0, 1). The default is UNIX exit values.

LIBPATH
The path used to locate dynamic link libraries (DLLs).

NLSPATH
Names the directory paths that the `ps2afp` command searches for message catalogs.

For information about setting and using environment variables, see *z/OS UNIX System Services User's Guide*.

The PostScript to AFP transform uses environment variables specified in the Infoprint Server transform configuration file. For information, see "Environment variables for the PDF to AFP and PostScript to AFP transforms" on page 62.

Files

`$HOME/.aopconf`
Contains the user-specific Infoprint Server configuration file. This file takes precedence over `/etc/Printsrv/aopd.conf`.

`/etc/Printsrv/aopd.conf`
Contains the system default Infoprint Server configuration file.

For the format of the configuration files, see *z/OS Infoprint Server Customization*.

Exit values

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.

UNIX exit values (default):

0
Successful. However, a warning or data stream error might have occurred during the transform.

1
The command was not accepted, or the transform failed.

Notes:

1. The AOP_MVS_RETURN_CODES environment variable determines the type of exit values.
2. The AOP_FAIL_ON_ERROR transform environment variable controls whether the transform fails when it detects a data stream warning or error.

sap2afp—Transform SAP OTF or ABAP data to AFP data

Format

`sap2afp` [-a *imagetype*] [-c *transformclass*] [-o *outputfile*] [-r *resolution*] [-b] [-f] [-h] [-k] [-s] [-u] [-I] [-T] [*inputfile ...*]

Description

The `sap2afp` command converts SAP Generic Output Format (SAPGOF) Output Text Format (OTF) and Advanced Business Application Programming (ABAP) data files:

- SAP OTF data files are converted into MOD:CA data streams.
- SAP ABAP data files are converted into line data streams.

The SAP OTF and ABAP data streams must be in ASCII or Unicode representation.

If you specify the same option multiple times, the command uses only the last option.

You can specify one or more input files to be transformed. If you specify more than one input file, the command concatenates the files, and the job attributes are determined by the first file transformed. If you do not specify an input file name, or if you specify a dash (`-`) as the file name, `sap2afp` uses standard input.

The output file name is also optional. If you do not specify one, the `sap2afp` command writes the results to standard output.

Options

Note: All options and values are case-sensitive.

-a *imagetype*
Specifies the IOCA function set that the transform uses when transforming OTF color image data.

Values are:

FS10
Creates black-and-white (also called bi-level) output.

FS45
Generates color IOCA FS45 output.

Default: FS10

-b

Specifies the SAP bar code orientation point as the lower left corner. This option tells the transform to override the transform's default behavior, and instead, place bar codes according to the SAP specification.

Consider using the -b option when differences between the SAP and AFP coordinates cause the bar codes to appear slightly out of place in the AFP output.

Default: The transform places bar codes according to the MO:DCA coordinate system.

-f

Specifies the use of TrueType fonts in the Unicode environment. The transform uses the configuration files `fonts.tab.unicode.truetype` (for OTF input) and `pagedef.tab.unicode.truetype` (for ABAP output) to map the TrueType fonts. For SAP OTF input files with TrueType inline fonts, the transform creates AFP data containing the referenced TrueType fonts as inline resources.

Default: The SAP to AFP transform does not use TrueType fonts to produce AFP output.

-h

Specifies that the OTF input file was generated using the SAPGOF_ST_CHARS:X entry from the SAP Unicode system.

When -h is specified, the SAP to AFP transform doubles the LENGTH parameters of ST and BC commands for Unicode pages, so that the OTF stream can be correctly processed in accordance with the SAP standard. You should only use the -h option if the SAPGOF_ST_CHARS:X entry has been set on the SAP Unicode system that generates the SAP input. This option is ignored for ABAP files.

Note: The -h option does not influence the ASCII or Unicode processing environment for the file.

Default: If the -h option is not applied, the SAP to AFP transform does not change the ST and BC commands for the OTF input.

-k

Specifies the use of the SAP specification for centering BX boxes; tells the transform to override AFP placement for SAP boxes defined by the BX command, and instead place boxes according to the SAP specification.

The -k option is useful if boxes in AFP output appear to be positioned slightly out of place, because of differences between the SAP and AFP coordinate systems.

Default: The transform prints boxes centered according to the MO:DCA coordinate system, not the SAP coordinate system.

-o *outputfile*

Specifies the output path and file into which the transform output (that is, AFP or line data) is written. The transform overwrites any existing data in the output file. If you specify more than one output file, the last specified path and file name are used. If you do not specify an output file, the result is written to standard output (STDOUT).

For SAP R/3 OTF data, you can specify either an MVS data set (such as a sequential or partitioned data set) or a UNIX file. However, for SAP ABAP data, you must specify an MVS data set.

If you specify an MVS data set, precede the data set name with `//`. When you specify a fully qualified name, two sets of quotation marks are required. For example, `"//'hlq.PDS(MYDOC)'"` or `"//'hlq.SEQDS'"`. When you specify a partially qualified name, you only need one set of quotation marks. For example, `"//PDS(MYDOC)"` or `"//SEQDS"`.

If you specify an MVS data set, you must allocate and catalog the data set before you run this command. Allocate the output data set with these characteristics:

- Record format: VBM
- Record length: 8K (8192) bytes or larger

Allocate an MVS data set that is large enough to hold the AFP data stream. Typically, the output AFP data stream is the same size as the input data stream.

Note: If the specified MVS output data set does not exist, the transform creates the data set; however, the data set does not have the correct record format and record length. If you attempt to print the data set, PSF for z/OS writes message APS114I.

-r *resolution*

Specifies the resolution used to print image data in the job. Select the correct resolution for the printer on which you intend to print the job.

Values are:

240
240 pels per inch (for example, IBM 3812, 3825, 3827, 3835, and 3900 printers)

300
300 pels per inch (for example, IBM 3112, 3116, 4019, 4028, 4029, and 4039 printers and Hewlett-Packard printers)

480
480 pels per inch

600
600 pels per inch (for example, the InfoPrint 60 and InfoPrint 4000 printers)

720
720 pels per inch (for example, the InfoPrint 60 and InfoPrint 4000 printers)

1200
1200 pels per inch (for example, the InfoPrint 60 and InfoPrint 4000 printers)

Default: The resolution specified in the `image.tab` configuration file.

Tips:

1. If you specify a resolution that the printer does not support, PSF prints the image under most conditions, but sometimes with degraded results.
2. A resolution of 300 pels typically produces good quality output on 300-, 600-, or 720-pel printers.

   300-pel output requires one-fourth the space and transmission time compared to 600-pel output.
3. A resolution of 600 pels typically produces good quality output on 1200-pel printers.

-s

Suppresses Graphic Object Content Architecture (GOCA) boxes. Some older printers do not print these boxes.

-u

Causes the transform to treat data as ASCII in SAPGOF OTF with ST commands that use Unicode code pages.

The -u option lets you transform SAP input files that were created in a Unicode environment and print them on older printers that do not support Unicode. This option produces AFP that does not use Unicode pages, so the data can print on ASCII-only printers. This option is ignored for ABAP files and for SAPGOFU data.

Note: If Unicode characters in the input file are not covered by code page ISO-8859-1, the input characters are replaced by spaces.

-I

Specifies that the transform produces MO:DCA IS/3 data streams. When this option is specified, TrueType fonts are used for creating the AFP output.

This functionality is only available when a single file is sent to the transform, not several at a time.

The -I option overrides other conflicting options:

- Because it enables Unicode, the -u option is ignored, if set.
- Because TrueType fonts are used when this option is set, the -I option disables the use of FOCA fonts and implicitly sets the -f option to use TrueType fonts.

Default: If the -I option is not specified, the default is MO:DCA IS/1 output.

-T
Requests a trace. Specify this option only if instructed by IBM service personnel. For more information, see Chapter 5, “Diagnosing errors,” on page 105.

Tip : You can use the filter-options job attribute with, for example, the `lp` command to pass any of these options except -o *outputfile* to the transform. For information about the filter-options job attribute, see *z/OS Infoprint Server User's Guide*.

Operand

inputfile
Specifies an input file to be transformed. If you specify more than one input file name, the `sap2afp` command concatenates the files. The results are written to a single output file (if one is specified) or to standard output.

The data in the file must be in stream mode. That is, it must not contain any record data.

If you do not specify an input file, or if you specify a dash (`-`) as the file name, `sap2afp` uses standard input.

Examples -- sap2afp

Transform a file, specifying resolution

To transform the SAP OTF file `myfile.otf` for printing on a 600-pel AFP printer, and write the result to a UNIX file called `myfile.afp`, enter:

```
sap2afp -r 600 -o myfile.afp myfile.otf
```

Transform and print a file

To transform the SAP OTF file `myfile.otf` into an AFP data stream, and send the result to the default printer with the `lp` command, enter:

```
sap2afp myfile.otf | lp
```

Transform a file using redirection

To transform the SAP OTF file `input.otf` into a UNIX output file called `output.afp`, enter:

```
sap2afp < input.otf > output.afp
```

Note: You can use redirection operators only with UNIX files.

Transform multiple files and concatenate the output

To transform the SAP OTF files `input.01.otf`, `input.02.otf`, ... `input.xx.otf` into one UNIX output file called `output.afp`, enter:

```
sap2afp -o output.afp input.01.otf input.02.otf ... input.xx.otf
```

Transform a UNIX file to an MVS data set

To transform file `input.sap` into an existing, cataloged MVS output data set called *`hlq`*`.OUTPUT.AFP(`*`member`*`)`, where `hlq` is your user ID, enter:

```
sap2afp -o "//'hlq.OUTPUT.AFP(member)'" input.sap
```

Transform an MVS data set, writing the output to a UNIX file

To transform the MVS data set *`hlq`*`.INPUT.OTF(`*`member`*`)`, where *hlq* is your user ID, into an output file called `output.afp`, enter:

```
sap2afp -o output.afp "//'hlq.INPUT.OTF(member)'"
```

Transform a file and receive MVS return codes

To transform the SAP file `myfile.sap` into an AFP data stream and write the result to an existing, cataloged MVS output data set called *`hlq`*`.OUTPUT.AFP(`*`member`*`)` and receive MVS return codes (0, 4, 8), enter on the same command line:

```
AOP_MVS_RETURN_CODES=YES sap2afp -o "//'hlq.OUTPUT.AFP(member)'" myfile.sap
```

MVS return codes indicate whether the transform was successful (0), a transform warning occurred (4), or a transform error occurred (8).

Transform a file to MO:DCA IS/3-compliant output

To transform the SAP file `file.sap` into an MO:DCA IS/3-compliant AFP data stream, and write the result to a UNIX file called `file.afp`, enter:

```
sap2afp -I -o file.afp file.sap
```

Transform a file, specifying TrueType fonts

To transform the SAP OTF file `myfile.otf` using TrueType fonts, and write the result to a UNIX file called `myfile.afp`, enter:

```
sap2afp -f -o myfile.afp myfile.otf
```

Improve alignment of boxes and bar codes

To transform the SAP file called `file.sap` into an AFP data stream with bar codes and boxes placed according to the SAP specification, and write the result to a UNIX file called `file.afp`, enter:

```
sap2afp -b -k -o file.afp file.sap
```

Generate color IOCA FS45 output

To transform the color SAP file called `file.sap` into a color IOCA FS45 AFP data stream and write the result to a UNIX file called `file.afp`, enter:

```
sap2afp -a fs45 -o file.afp file.sap
```

Generate color IOCA FS45 output

To transform the color SAP file called `file.sap` into a color IOCA FS45 AFP data stream and write the result to a UNIX file called `file.afp`, enter:

```
sap2afp -a fs45 -o file.afp file.sap
```

Print SAPGOF data on printers that do not support Unicode

To transform the SAP file called `file.sap` that was generated as SAPGOF and contains Unicode characters, write the result to a UNIX file called `file.afp`, that can then be printed on a printer that does not support Unicode, enter:

```
sap2afp -u -o file.afp file.sap
```

Transform files created with SAPGOF_ST_CHARS: X

To transform SAP data called `file.sap` that was generated on a SAP Unicode system with the SAPGOF_ST_CHARS:X entry set, and write the result to a UNIX file called `file.afp`, enter:

```
sap2afp -h -o file.afp file.sap
```

Environment variables

The `sap2afp` command uses these environment variables:

AOPCONF
Names the Infoprint Server configuration file. This variable takes precedence over the user-specific configuration file (`$HOME/.aopconf`) and the system default configuration file (`/etc/Printsrv/aopd.conf`). For more information about the configuration file, see *z/OS Infoprint Server Customization*.

AOP_MVS_RETURN_CODES
Specifies whether the command returns MVS return codes (0, 4, 8) or UNIX exit values (0, 1). The default is UNIX exit values.

LIBPATH
The path used to locate dynamic link libraries (DLLs).

NLSPATH
Names the directory paths that the `sap2afp` command searches for message catalogs.

For information about setting and using environment variables, see *z/OS UNIX System Services User's Guide*.

The SAP to AFP transform uses environment variables specified in the Infoprint Server transform configuration file. For information, see "Environment variables for the SAP to AFP transform" on page 72.

Files

`$HOME/.aopconf`
Contains the user-specific Infoprint Server configuration file. This file takes precedence over `/etc/Printsrv/aopd.conf`.

`/etc/Printsrv/aopd.conf`
Contains the system default Infoprint Server configuration file.

Exit values

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.

UNIX exit values (default):

0
Successful. However, a warning or data stream error might have occurred during the transform.

1
The command was not accepted, or the transform failed.

Notes:

1. The AOP_MVS_RETURN_CODES environment variable determines the type of exit values.
2. The AOP_FAIL_ON_ERROR transform environment variable controls whether the transform fails when it detects a data stream warning or error.

Transforming data with the AOPBATCH program

This information describes how to use the Infoprint Server AOPBATCH program to run the transform commands. The AOPBATCH program lets you submit a batch job to transform data to AFP. Infoprint Server provides the AOPBATCH program in SYS1.LINKLIB.

AOPBATCH parameters

The AOPBATCH parameters are the name of the transform command, followed by transform options and arguments, in this format:

```
EXEC PGM=AOPBATCH,PARM='/transform_name transform_options'
```

/
The optional slash indicates that the PARM data that follows is input to AOPBATCH. If you omit the initial slash, your PARM data might be interpreted as C++ run-time options. You *must* include the initial slash if any of the PARM data itself includes a slash. For example, if the transform name is `/mylib/ps2afp`, specify: `PARM='//mylib/ps2afp ...'`.

transform_name
The name of an executable transform program that resides in a z/OS UNIX file. The name of the transform program is case-sensitive. You can specify one of these command names: `pcl2afp, pdf2afp, ps2afp, sap2afp`.

If the transform program does not reside in one of the directories that are specified in the PATH environment variable, also specify the pathname. You can use the STDENV DD statement to set the PATH environment variable if the default value set by AOPBATCH is not suitable. For information about the defaults set for environment variables, see “AOPBATCH DD statements” on page 46.

transform_options
Options and arguments accepted by the transform. For a description of each, see “pcl2afp—Transform PCL data to AFP data” on page 11, “pdf2afp—Transform PDF data to AFP data” on page 21, “ps2afp—Transform PostScript data to AFP data” on page 30, or “sap2afp—Transform SAP OTF or ABAP data to AFP data” on page 39.

You must specify the transform input data set or file as a transform argument, and you must specify the -o transform option to identify where you want the transform to write its output. (This is because the transform cannot write its output to standard output and cannot read input from standard input.) To identify the transform input and output data set or file, you can specify either a DD statement name or a data set or file name. You must specify a DD statement name if you want to write the transform output to an MVS data set that does not already exist.

Specify the names of DD statements to the transform in this format:

```
//DD:DDname
```

The name of the DD statement is *DDname*.

When you specify an MVS data set name in the -o option, 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, a high-level qualifier is added to the name you specify:

- If you are running under TSO (batch or interactive), the TSO user prefix is appended.
- If you are running under MVS batch or IMS™ (batch or online), the RACF® user ID is appended.
- If your system does not use RACF, a high-level qualifier is not added.

For examples of different ways to specify transform input and output data sets and files, see "AOPBATCH examples" on page 47.

Note: A maximum region size of 6 MB should be sufficient.

AOPBATCH DD statements

The AOPBATCH JCL procedure accepts these standard DD statements:

STDENV
Specifies environment variables that the transform command uses. You can specify the environment variables in-stream in the JCL, in an MVS data set, or in a UNIX file. Specify the environment variables in the format *variable*=*value*, with one environment variable per line or record. Sequence numbers in columns 73 - 80 in data that is specified with the STDENV DD statement are ignored and not treated as part of the data.

If you omit the STDENV DD statement or do not specify one of 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/%L/%N:/usr/lpp/Printsrv/En_US/%N:/usr/lib/nls/msg/%L/%N
```

Notes:

1. To obtain MVS return codes (0, 4, 8) instead of the default UNIX exit values (0, 1), set the AOP_MVS_RETURN_CODES=yes environment variable. For information, see "AOP_MVS_RETURN_CODES environment variable" on page 51.
2. AOPBATCH sets the HOME environment variable to the user's home directory and sets the LOGIN variable to the user ID.
3. Do not specify the _BPX_SHAREAS environment variable. AOPBATCH sets it appropriately.

STDERR
Specifies the system output data set where error messages are to be written. The data set can be an MVS data set or a UNIX file. The transforms do not write messages related to errors in the input data stream in this data set. Instead, the transforms write these messages at the end of the output document.

STDOUT
Specifies the system output data set where informational messages are to be written. The data set can be an MVS data set or a UNIX file.

You can also include DD statements to specify MVS data sets that contain input data to be transformed, the transformed output, or job attributes that are input to the transform.

Rules:

1. Do not use DD names STDIN, STDOUT, or STDERR to specify the transform input and output data sets. Instead, use other DD names, such as INPUT and OUTPUT, which are used in the examples.
2. If you have not added the Language Environment® run-time library (CEE.SCEERUN) or the C++ run-time library (CBC.SCLBDLL) to the system LNKLST, specify these data sets in a STEPLIB DD statement.
3. The PDF to AFP and the PostScript to AFP transforms cannot transform concatenated input files. If you want to transform more than one PDF or PostScript file in the same job, create a separate step to transform each file. (The PCL to AFP transform can transform concatenated input files.) See "AOPBATCH examples" on page 47 for an example of how to transform and print multiple PDF files in the same job.
4. If you want to write the AFP data stream to an MVS data set, you must allocate and catalog the MVS data set before you run AOPBATCH or include a DD statement in the AOPBATCH job to allocate the data set. Allocate a data set with these characteristics:
   - Record format: VBM.
   - Record length: 8192 (8K) or larger.
   - Disposition: SHR or OLD overwrites any existing data in the data set; MOD appends the output to any existing data. MOD is the default.

   Allocate an MVS data set that is large enough to hold the AFP data stream. The size of the AFP data stream depends on the size and complexity of the document, the type of image compression, and the resolution of the image. Typically, the PCL to AFP, PDF to AFP, and PostScript to AFP transforms create an output AFP data stream that is several times as large as the input data stream. For more information about the size of the AFP data stream, see "Calculating the size of the AFP data stream" on page 50.

   Note: If you specify an MVS output data set that does not exist, the transform creates the data set; however, the data set does not have the correct record format and record length. If you attempt to print the data set, PSF for z/OS writes message APS114I.

AOPBATCH examples

These examples show how to use the AOPBATCH procedure to transform data. For additional AOPBATCH examples, see *z/OS Infoprint Server User's Guide*.

Specify transform input and output in MVS data sets, and specify environment variables

This example shows how to transform data when the transform input is in an MVS data set and transform output is written to an MVS data set. The PostScript to AFP transform reads PostScript input from data set HLQ.INPUT.PS and writes AFP output to data set HLQ.OUTPUT.AFP. HLQ represents the high-level qualifier.

This example also shows how to specify environment variables in-stream in the STDENV DD statement. If you installed Infoprint Server and created Infoprint Server configuration files in default directories, you do not need to specify the PATH, LIBPATH, NLSPATH, and AOPCONF environment variables.

```
//AOPBATCH JOB  ...
//TRANSFRM EXEC PGM=AOPBATCH,PARM='/ps2afp -c letter -r 300 -o //DD:OUT
//              PUT DD:INPUT'
//INPUT    DD   DSN=HLQ.INPUT.PS,DISP=SHR
//OUTPUT   DD   DSN=HLQ.OUTPUT.AFP,DISP=(NEW,CATLG,DELETE),
// DCB=(RECFM=VBM,LRECL=32756,BLKSIZE=32760),SPACE=(CYL,(1,1))
//STDOUT   DD   SYSOUT=*
//STDERR   DD   SYSOUT=*
//STDENV   DD   *
AOP_MVS_RETURN_CODES=YES
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/En_US/%N
AOPCONF=/etc/Printsrv/aopd.conf
/*
```

Specify transform input in a UNIX file and output in an MVS data set

This example shows how to transform data when the transform input is in a UNIX file and transform output is written to an MVS data set. The PDF to AFP transform reads the PDF input from file `/tmp/input.pdf` and writes AFP output to `HLQ.OUTPUT.AFP`.

```
//AOPBATCH JOB  ...
//TRANSFRM EXEC PGM=AOPBATCH,PARM='/pdf2afp -c letter_300 -o //DD:OUTPU
//             T /tmp/input.pdf'
//OUTPUT   DD   DSN=HLQ.OUTPUT.AFP,DISP=(NEW,CATLG,DELETE),
// DCB=(RECFM=VBM,LRECL=32756,BLKSIZE=32760),SPACE=(CYL,(1,1))
//STDOUT   DD   SYSOUT=*
//STDERR   DD   SYSOUT=*
```

Transform and print data sets

This example shows how to transform data and print the output from the transform in a subsequent step.

In the first step, the PostScript to AFP transform reads PostScript input from data set `HLQ.INPUT.PS` and writes AFP output to data set `HLQ.OUTPUT.AFP`, where HLQ represents the high-level qualifier.

In the second step, the AOPPRINT JCL procedure prints the AFP output to the printer named `mypsfprinter`, which is defined in the Printer Inventory.

For information about AOPPRINT, see *z/OS Infoprint Server User's Guide*.

```
//AOPBATCH JOB  ...
//TRANSFRM EXEC PGM=AOPBATCH,PARM='/ps2afp -o //DD:OUTPUT //DD:INPUT'
//INPUT    DD   DSN=HLQ.INPUT.PS,DISP=SHR
//OUTPUT   DD   DSN=HLQ.OUTPUT.AFP,DISP=(NEW,CATLG,DELETE),
// DCB=(RECFM=VBM,LRECL=32756,BLKSIZE=32760),SPACE=(CYL,(1,1))
//STDOUT   DD   SYSOUT=*
//STDERR   DD   SYSOUT=*
//PRINT    EXEC AOPPRINT,PRINTER='mypsfprinter'
//SYSIN    DD   DSN=HLQ.OUTPUT.AFP,DISP=SHR
```

Transform and print data sets only if the transform is successful

This example shows how to transform data and print the output from the transform in a subsequent step only if no warnings or errors occur during the transform.

In the first step, the PostScript to AFP transform reads PostScript input from data set `HLQ.INPUT.PS` and writes AFP output to data set `HLQ.OUTPUT.AFP`, where HLQ represents the high-level qualifier.

The STDENV data set specifies the AOP_MVS_RETURN_CODES environment variable. Because this variable is set to YES, the transform returns MVS return codes: 0 (successful), 4 (warning), 8 (failure).

In the second step, the AOPPRINT JCL procedure prints the AFP output to the printer named `mypsfprinter`, which is defined in the Printer Inventory. This step runs only if the return code from the transform step is less than 4.

For information about AOPPRINT, see *z/OS Infoprint Server User's Guide*.

```
//AOPBATCH JOB  ...
//TRANSFRM EXEC PGM=AOPBATCH,PARM='/ps2afp -o //DD:OUTPUT //DD:INPUT'
//INPUT    DD   DSN=HLQ.INPUT.PS,DISP=SHR
//OUTPUT   DD   DSN=HLQ.OUTPUT.AFP,DISP=(NEW,CATLG,DELETE),
// DCB=(RECFM=VBM,LRECL=32756,BLKSIZE=32760),SPACE=(CYL,(1,1))
//STDOUT   DD   SYSOUT=*
//STDERR   DD   SYSOUT=*
//STDENV   DD   *
AOP_MVS_RETURN_CODES=YES
/*
//PRINT    EXEC AOPPRINT,PRINTER='mypsfprinter',
//  COND=(4,LT,TRANSFRM)
//SYSIN    DD   DSN=HLQ.OUTPUT.AFP,DISP=SHR
```

Continuing parameter fields in JCL

To continue a parameter field:

1. Interrupt the field after a complete parameter or subparameter, including the comma that follows it, at or before column 71.
2. Code // in columns 1 and 2 of the following statement.
3. Code a blank character in column 3 of the following statement.
4. Continue the interrupted parameter or field beginning in any column from 4 through 16.

To continue a parameter that is enclosed in apostrophes:

1. Extend the parameter to column 71. Do not code an apostrophe in column 71.
2. Code // in columns 1 and 2 of the following statement.
3. Continue the parameter in column 16 of the following statement even if doing this splits the parameter.

AOPBATCH exit values

AOPBATCH returns one of these values:

1
An error occurred while AOPBATCH was trying to write to the STDOUT DD.

2
An error occurred while AOPBATCH was trying to write to the STDERR DD.

4
AOPBATCH could not spawn the process.

nn
The exit value from the transform command.

The transforms 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.

Using the Print Interface subsystem

You can use the Print Interface subsystem (provided with Infoprint Server) to transform data created by a batch application to AFP format so that it can be printed on an AFP printer. The Print Interface subsystem writes the AFP output from the transform to an output data set on the JES spool. PSF can then print the output data set on an AFP printer.

For information about the Print Interface subsystem, see *z/OS Infoprint Server User's Guide*.

Example: This example shows how to print a PDF document created by a batch application on an AFP printer using form definition F1MYDEF, which is in library USERA.MYLIB, to format the AFP data. It assumes that:

- The Print Interface subsystem named AOP1 has been started.
- Printer definition `myafpprinter` exists in the Infoprint Server Printer Inventory with these characteristics:
  - The PDF to AFP transform is specified.
  - The class and destination name for the PSF-controlled AFP printer are specified. (CLASS and DEST are JES work-selection parameters that can direct output from the JES spool to a PSF-controlled printer.)

```
//SUBSYS JOB …
//STEP1  EXEC PGM=USERA
```

```
//OUTDS1 OUTPUT FORMDEF=MYDEF,USERLIB=USERA.MYLIB
//DD1    DD SUBSYS=(AOP1,'myafpprinter'),OUTPUT=*.OUTDS1
```

Using the lprafp command

The `lprafp` command lets you print documents from Windows systems and UNIX systems (such as HP-UX and SunOS) and specify Infoprint Server job attributes. For example, when you use the `lprafp` command, you can specify the name of a form definition in the `form-definition` attribute and the number of copies in the copies attribute.

You might want to use the `lprafp` command to print PDF or PostScript documents that are on your Windows or UNIX systems.

You can download the `lprafp` command and find out how to use it at LPRAFP with Infoprint Server for z/OS web page (www.infoprintsolutionscompany.com/internet/dcfdata.nsf/vwWeb/P4000243).

Example: This example shows how to print two copies of a PDF document named `myfile.pdf` on an AFP printer named `myafpprinter` using form definition F1F0RM to format the AFP data. It assumes that:

- The host name of the z/OS system where Infoprint Server is running is `systema`.
- Printer definition `myafpprinter` exists in the Infoprint Server Printer Inventory with these characteristics:
  - The PDF to AFP transform is specified.
  - The class and destination name for the PSF-controlled AFP printer are specified. (CLASS and DEST are JES work-selection parameters that can direct output from the JES spool to a PSF-controlled printer.)

```
lprafp -o"XAOP form-definition=f1form copies = 2" -p myafpprinter -s systema
myfile.pdf
```

Calculating the size of the AFP data stream

If a transform command writes the AFP data stream to an MVS data set, you must allocate an MVS data set that is large enough to hold the AFP data stream. Typically, the AFP data stream is several times as large as the input data stream.

The size of the AFP data stream depends on the size and complexity of the document, whether the AFP images are compressed, and the resolution of the images. You specify the type of image compression in the -a option of the transform command. You can specify the resolution in the -r option of the transform command, in the AOP_RESOLUTION environment variable (PCL to AFP transform), and in the `image.tab` configuration file (SAP to AFP transform).

Steps for calculating the size of the AFP data stream:

1. Use these formulas to calculate the number of bytes per page:

   page width * resolution * page height * resolution = bits per page
   bits per page / 8 = bytes per page

   The page width, page height, and resolution must be in the same units.

   For example, for an 8.5 x 11 inch page at 600 pels (bits) per inch resolution:

   ```
   (8.5 * 600 * 11.0 * 600) / 8  = 4,207,500 bytes per page
   ```

   Note: If the pages contain blank areas, which is typical of most text pages, the number of bytes per page can be substantially smaller.

2. If the transform creates a compressed IOCA image (G4 MMR or MMR format), multiply the bytes per page by 50%. For example:

```
4,207,500 * 0.5 = 2,103,750 bytes per page
```

3. Multiply the number of bytes per page by the number of pages in the document. For example, for 100 pages:

```
2,103,750 * 100 = 210,375,000 bytes in AFP data stream
```

Note: The SAP to AFP transform does not produce full-page AFP images. This transform creates an AFP data stream that is approximately the same size as the input data stream.

Tips: To reduce the size of the AFP data stream:

1. Specify the lowest resolution that produces good quality output. A resolution of 300 pels typically produces good quality output on 300- or 600-pel printers. A resolution of 300 pels typically produces good quality output on 300-, 600-, or 720-pel printers. 300-pel output requires one-fourth the space and transmission time compared to 600 pel output. (The default resolution for the PDF to AFP, PostScript to AFP, and SAP to AFP transforms is 600 pels. The default resolution for the PCL to AFP transform is 240 pels.)
2. Specify compressed IOCA images (-a io1-g4 or -a io1-mmr). Specify uncompressed IOCA or IM1 images only if you know that your printer does not support compressed images. Most newer printers support compressed IOCA images.
3. Do not specify IM1 images at a resolution that the printer does not support because Print Services Facility (PSF) converts the IM1 images to uncompressed IOCA images at the resolution the printer requires. This conversion can cause PSF to use very large amounts of CPU time.

Note: You should only embed TrueType fonts if the product you are using does not use a RAT. If you embed TrueType fonts, the output file will be much larger, which might decrease performance or cause the transform to run out of space when writing the output

AOP_MVS_RETURN_CODES environment variable

The AOP_MVS_RETURN_CODES environment variable determines whether the transform command returns MVS return codes (0, 4, 8) instead of UNIX exit values (0, 1). MVS return codes let you determine whether a warning or error occurred during the transform.

MVS return codes are useful when you use the AOPBATCH JCL procedure to run a transform and want a subsequent step in the job to run only if the transform step was successful. For an example of using MVS return codes with AOPBATCH, see “AOPBATCH examples” on page 47.

AOP_MVS_RETURN_CODES
Specifies whether the `lp`, `pcl2afp`, `pdf2afp`, `ps2afp`, and `sap2afp` commands return MVS return codes or UNIX exit values:

YES
MVS return codes:

Value
Meaning

0
Successful.

4
A warning occurred during the transform.

8
The command was not accepted, a data stream error occurred during the transform, or the transform failed.

NO

UNIX exit values:

Value
Meaning

0
Successful. However, a warning or data stream error might have occurred during the transform.

1
The command was not accepted, or the transform failed.

Notes:

1. Specify this environment variable before the command on the z/OS UNIX command line. 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).
2. If you use the AOPPRINT JCL procedure, specify this environment variable in the STDENV data set of the procedure. AOPPRINT runs the `lp` command and returns the code from the `lp` command.
3. If you use the AOPBATCH JCL procedure to run a transform command or the `lp` command, specify this environment variable in the STDENV data set of the procedure. AOPBATCH returns the code from the transform command or the `lp` command.
4. Do not specify this environment variable in the transform entry in the Infoprint Server transform configuration file, `aopxfd.conf`.
5. You can specify the value using lowercase or uppercase letters (for example, `no` or `NO`).

Default: NO

Example: AOP_MVS_RETURN_CODES=YES

Table 5 on page 52 shows the transform exit values when the AOP_FAIL_ON_ERROR and AOP_MVS_RETURN_CODES environment variables are set and the transform is successful, a warning occurs, or a data stream error occurs. (The **Fail on error** field in the printer definition and the fail-on-transform-error job attribute provide equivalent function to the AOP_FAIL_ON_ERROR environment variable.)

Table 5: Transform exit values with AOP_FAIL_ON_ERROR and AOP_MVS_RETURN_CODES environment variables

AOP_FAIL_ON_ERROR	AOP_MVS_RETURN_CODES=YES	AOP_MVS_RETURN_CODES=NO
NO	Successful = 0 Warning = 4 Error = 8	Successful = 0 Warning = 0 Error = 0
ERROR	Successful = 0 Warning = 4 Error = 8	Successful = 0 Warning = 0 Error = 1
WARNING	Successful = 0 Warning = 8 Error = 8	Successful = 0 Warning = 1 Error = 1

Chapter 3. Customizing transforms

This information contains tasks that describe how to customize the transforms:

- “Customizing the PCL to AFP transform” on page 53
- “Customizing the PDF to AFP and PostScript to AFP transforms” on page 60
- “Customizing the SAP to AFP transform” on page 70

Customizing the PCL to AFP transform

To customize the PCL to AFP transform, do the tasks listed in this table. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies to your installation.

Task	Condition
“Creating an entry in the Infoprint Server transform configuration file” on page 53	Required

Creating an entry in the Infoprint Server transform configuration file

You must create at least one transform entry in the Infoprint Server transform configuration file (`aopxfd.conf`) for the PCL to AFP transform. In the transform entry, you specify:

- Environment variables that control the transform
- Attributes that control how the Infoprint Server Transform Manager manages the transform

For information about how to create and edit the Infoprint Server transform configuration file, see *z/OS Infoprint Server Customization*. After you update the transform configuration file, you must restart the Infoprint Server Transform Manager.

Tip: After you restart the Infoprint Server Transform Manager, check for error messages in the transform's `standard error`. If you find any error messages, fix the errors and restart the Infoprint Server Transform Manager. For more information about how to find the transform message logs, see “Finding the transform stderr file” on page 105.

Transform classes

If you want to specify different transform options for different printers, you can create different classes of the transform. For example, you might create classes for printers that print on different paper sizes.

For each transform class, you must create a separate transform entry. You select a name for the transform class in the transform entry (see the transform attribute).

To use a transform class, job submitters must specify the class name in the -c option on the `pcl2afp` command or in the filter-options job attribute, and administrators must specify the class name in the -c filter option in the printer definition. For example, these z/OS UNIX commands use the `letter_300` transform class:

```
pcl2afp -c letter_300 -o myfile.afp myfile.pcl
lp -d myprinter -o "filter-options='-c letter_300'" myfile.pcl
```

Tip: The sample transform configuration file, `/usr/lpp/Printsrv/samples/aopxfd.conf`, shows examples of transform entries with different transform classes. Also, see “Examples -- Transform configuration file entries for the PCL to AFP transform” on page 58.

Format of a PCL to AFP transform entry

The format of a PCL to AFP transform entry in the transform configuration file (`aopxfd.conf`) is:

```
transform pcl2afp[_transformclass]
    start-command = pcl2afpd
    [ environment  = {name -> value [ name -> value]… } ]
    [ maximum-active = number ]
    [ maximum-idle-time = seconds ]
    [ minimum-active = number ]
    ;
```

transform pcl2afp[_*transformclass*]
This statement indicates the beginning of a transform entry.

pcl2afp
The name of the transform.

transformclass
The name of an optional transform class. Specify from 1 to 63 characters, including letters, numbers, or special characters. The name of the transform class is case-sensitive.

Default: No transform class.

Example: transform pcl2afp_letter_300

start-command = pcl2afpd
This attribute names the transform daemon and option. Enclose the value in single or double quotation marks if you specify the option.

pcl2afpd
The name of the transform daemon. If the transform daemon is not in a directory that is identified in the PATH environment variable in the `aopstart` EXEC, specify the full directory path name of the daemon. (The `pcl2afpd` daemon is installed in `/usr/lpp/Printsrv/bin`.) This attribute is required.

environment = {*name* –> *value* [*name* –> *value*]... }
Environment variables that specify transform options. Enclose the entire set of environment variables in braces. The values in these environment variables override environment variables with the same name that are set in the `aopstart` EXEC. If a value contains special characters or spaces, enclose the value in single or double quotation marks.

For information about the environment variables you can specify, see “Environment variables for the PCL to AFP transform” on page 55.

Rule: Environment variable names must be uppercase. Environment variable values are case-insensitive.

Default: None.

Example:

```
environment = {AOP_RESOLUTION -> 240}
```

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 the minimum-active attribute. For more information about the format of a transform entry, see *z/OS Infoprint Server Customization*.

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. For more information about the format of a transform entry, see *z/OS Infoprint Server Customization*.

Default: Idle transform daemons are not shut down.

minimum-active = *number*
The minimum number of transform daemons that the Infoprint Server Transform Manager activates concurrently. Specify a number less than or equal to the number specified in the maximum-active attribute. For more information about the format of a transform entry, see *z/OS Infoprint Server Customization*.

Default: `minimum-active = 0`

Environment variables for the PCL to AFP transform

Use environment variables to specify:

- Height and width of the page
- Margins of the page
- Resolution of the printer
- Record length of AFP images
- Tracing options
- Whether the transform fails when warnings or data stream errors occur
- Whether the transform writes messages to a trailer error page

You can specify these environment variables. All environment variables are optional.

_BPX_JOBNAME
The job name for this transform. When you assign a different job name to each class of transform, the operator can manage the transform daemons more effectively. Specify a job name of 1 to 8 alphanumeric characters. Incorrect job names are ignored. For more information about the _BPX_JOBNAME variable, see *z/OS UNIX System Services Planning*.

Default: The job name is AOPXFD.

Example: `environment = {_BPX_JOBNAME -> PCL2AFPD}`

AOP_FAIL_ON_ERROR
Specifies whether the transform stops processing when a warning or data stream error occurs during the transform. Warnings can indicate degraded output.

Valid values are:

error
The transform fails when a data stream error occurs and does not create an output document. However, the transform continues if warnings occur.

no
The transform continues processing when a warning or data stream error occurs and creates an output document. This value is the default.

warning
The transform fails when a warning or data stream error occurs and does not create an output document.

Tips:

1. For backward compatibility, value yes is allowed and is equivalent to error.
2. The fail-on-transform-error job attribute and the Fail on error field in the printer definition override this value.
3. You can specify the value by using lowercase or uppercase letters (for example, no or NO).

Default: AOP_FAIL_ON_ERROR –> no

Example: `environment = {AOP_FAIL_ON_ERROR -> error}`

AOP_HORIZONTAL_MARGINS
The left and right margins of the page. Specify a number followed by one of these units:

in
Inches

mm
Millimeters

pel
Pels, the default unit

Tips:

1. Inch values and millimeter values can contain a decimal point; pel values cannot.
2. You can specify the unit by using lowercase or uppercase letters (for example, `in` or `IN`).
3. The -x option of the `pcl2afp` command and transform filter overrides this value.

Default: `AOP_HORIZONTAL_MARGINS –> 0.167in`

AOP_PAGE_HEIGHT
The height of the page. Specify a number followed by one of these units:

in
Inches

mm
Millimeters

pel
Pels, the default unit

Tips:

1. Inch values and millimeter values can contain a decimal point; pel values cannot.
2. You can specify the unit by using lowercase or uppercase letters (for example, `in` or `IN`).
3. The -l option of the `pcl2afp` command and transform filter overrides this value.

Default: AOP_PAGE_HEIGHT –> 11in

AOP_PAGE_WIDTH
The width of the page. Specify a number followed by one of these units:

in
Inches

mm
Millimeters

pel
Pels, the default unit

Tips:

1. Inch values and millimeter values can contain a decimal point; pel values cannot.
2. You can specify the unit by using lowercase or uppercase letters (for example, `in` or `IN`).
3. The -w option of the `pcl2afp` command and transform filter overrides this value.

Default: `AOP_PAGE_WIDTH –> 8.5in`

AOP_RECLEN
The length of the AFP images the transform creates. Each output record contains an AFP image and an additional 17 bytes of information (for example, header information). For example, if you specify a value of 8 (8,000 bytes) for the AFP image in this environment variable, the length of each output record is 8,017 bytes.

Valid values are from 8 to 32, in units of 1,000 bytes. For example, 8 = 8,000.

Tip: The default value is suitable for most installations. However, a larger record length might improve performance.

Default: AOP_RECLEN -> 8

Example: `environment = {AOP_RECLEN -> 16}`

AOP_RESOLUTION
The resolution of the printer in pels per inch. The transform creates the AFP images with this resolution. Valid values: 240, 300, 480, 600, 720, and 1200.

Default: AOP_RESOLUTION –> 240

Example: `environment = {AOP_RESOLUTION –> 300}`

Tips:

1. If you specify a resolution that the printer does not support, PSF prints the image under most conditions, but sometimes with degraded results.
2. A resolution of 300 pels typically produces good quality output on 300-, 600-, or 720-pel printers. 300-pel output requires one-fourth the space and transmission time compared to 600-pel output.
3. A resolution of 600 pels typically produces good quality output on 1200-pel printers.
4. The -r option of the `pcl2afp` command and transform filter overrides this value.

AOP_TRAILER_ERROR_PAGE
Specifies whether the transform writes messages for transform warnings and data stream errors to a trailer error page. The trailer error page is the last page in the output document. Warnings can indicate degraded output.

Valid values are:

error
The transform writes error messages, but not warning messages, to a trailer error page. This option does not apply when AOP_FAIL_ON_ERROR–> `error` or AOP_FAIL_ON_ERROR–>`warning` because no output document is created when an error occurs.

no
The transform does not write messages to a trailer error page.

`warning`
The transform writes warning and error messages to a trailer error page. This option does not apply when AOP_FAIL_ON_ERROR–>`warning` because no output document is created when an error or warning occurs. This value is the default.

Tips:

1. The trailer-transform-error-page job attribute and the **Trailer error page** field in the printer definition override this value.
2. You can specify the value by using lowercase or uppercase letters (for example, no or NO).

Default: `AOP_TRAILER_ERROR_PAGE –> warning`

Example: `environment = {AOP_TRAILER_ERROR_PAGE –> error}`

AOP_VERTICAL_MARGINS
The top and bottom margins of the page. Specify a number followed by one of these units:

in
Inches

mm
Millimeters

`pel`
Pels, the default unit

Tips:

1. Inch values and millimeter values can contain a decimal point; pel values cannot.
2. You can specify the unit by using lowercase or uppercase letters (for example, `in` or `IN`).

3. The -y option of the pcl2afp command and transform filter overrides this value.

Default: AOP_VERTICAL_MARGINS –> 0.167in

AOPTRACEDIR
The full path name of the directory where the transform writes trace information. You can specify the same directory for different transform classes. The name of the trace file identifies the transform and transform class, and contains a time stamp.

This directory must already exist. If the directory does not exist, the transform writes trace information to the transform's standard error. For information about how to find the standard error, see "Finding the transform stderr file" on page 105.

Default: AOPTRACEDIR -> /var/Printsrv/trace

Examples:

```
environment = {AOPTRACEDIR -> /var/Printsrv/xfd}
environment = {AOPTRACEDIR -> .}
```

AOPTRACEON
Turns on tracing. The transform traces all transform requests that use this transform class. Any value turns on tracing. To turn tracing off, do not specify this environment variable. Specify this environment variable only if IBM directs you to do so. Tracing can adversely affect performance.

Default: Tracing is turned off.

Example: environment = {AOPTRACEON -> 1}

AOP_TRIM
Indicates whether the transform creates trimmed AFP images or full-page AFP images. Trimmed AFP images are smaller than full-page AFP images. Therefore, AFP output files that contain trimmed AFP images are smaller and can be transmitted more quickly over a network.

In contrast to full-page AFP images, trimmed AFP images do not include any white space that surrounds the data on the page. The transform positions trimmed AFP images on the page so that the data prints in the correct position on the page. For example, if a page is 8 1/2 by 11 inches and has a 1-inch margin on all sides:

- A trimmed AFP image is 6 1/2 x 9 inches. It is positioned at offset 1 inch (x), 1 inch (y) on the page.
- A full-page AFP image is 8 1/2 by 11 inches. It is positioned at offset 0 (x), 0 (y) on the page.

AFP documents print the same whether you specify AOP_TRIM = yes or AOP_TRIM = no. However, AFP overlays and AFP page segments typically print differently because trimmed AFP images are smaller than full-page AFP images. A page segment that contains a trimmed AFP image typically prints higher and to the left on a page. (The -t transform option determines the type of output. For example, -t pagesegment creates a page segment.)

Values are:

yes
The transform creates trimmed AFP images (default).

no
The transform creates full-page AFP images.

The V1 transforms always create full-page AFP images. To create full-page AFP images in V2, you must specify the AOP_TRIM=no environment variable.

Tip: In addition to the environment variables listed in this information, you can specify other environment variables. For example, you can specify the _CEE_RUNOPTS environment variable. For information, see "Changing Language Environment run-time options" on page 59.

Examples -- Transform configuration file entries for the PCL to AFP transform

This information shows sample transform entries in the Infoprint Server transform configuration file (aopxfd.conf) for the PCL to AFP transform.

Print on letter size paper on 300-pel printers

This transform entry can be used for printers that print on letter size paper (8.5 inches wide, 11 inches high) and that have a resolution of 300 pels per inch. This transform entry creates transform class "letter_300". To use this transform class, specify the class in the -c transform option as shown in these two examples:

```
pcl2afp -c letter_300 -o myfile.afp myfile.pcl
lp -o "filter-options='-c letter_300'" -d myprinter myfile.pcl
```

```
 transform pcl2afp_letter_300
      start-command = pcl2afpd
      maximum-active = 2
      maximum-idle-time = 300   # 5 minutes
      minimum-active = 1
      environment = {
          _BPX_JOBNAME -> PCL2AFPD
          AOP_PAGE_HEIGHT -> 11in
          AOP_PAGE_WIDTH -> 8.5in
          AOP_HORIZONTAL_MARGINS -> 0.167in
          AOP_VERTICAL_MARGINS -> 0.167in
          AOP_RESOLUTION -> 300
       }
;
```

Print on A4 paper on 300-pel printers

This transform entry can be used for printers that print on A4 paper (210 millimeters wide, 297 millimeters high). This transform entry creates transform class "a4_300". To use this transform class, specify the class in the -c transform option as shown in these two examples:

```
pcl2afp -c a4_300 -o myfile.afp myfile.pcl
lp -o "filter-options='-c a4_300'" -d myprinter myfile.pcl
```

```
transform pcl2afp_a4_300
      start-command = pcl2afpd
      maximum-active = 2
      maximum-idle-time = 300  # 5 minutes
      minimum-active = 1
      environment = {
          _BPX_JOBNAME -> PCL2AFPD
          AOP_PAGE_HEIGHT -> 297mm
          AOP_PAGE_WIDTH -> 210mm
          AOP_HORIZONTAL_MARGINS -> 4.242mm
          AOP_VERTICAL_MARGINS -> 4.242mm
          AOP_RESOLUTION -> 300
       }
;
```

Changing Language Environment run-time options

The transform sets these run-time options for the Language Environment(LE):

- ALL31(ON)
- ANYHEAP(16K,8K,ANYWHERE,FREE)
- BELOWHEAP(8K,4K,FREE)
- DEPTHCONDLMT(10)
- ERRCOUNT(0)
- HEAP(512K,512K,ANYWHERE,FREE)
- HEAPPOOLS=(ON)
- LIBSTACK(8K,4K,FREE)

- STACK(128K,128K,ANYWHERE,KEEP,512K,128K)
- STORAGE(NONE,NONE,NONE,8K)
- TERMTHDACT(DUMP)
- THREADHEAP(8K,8K,ANYWHERE,KEEP)
- THREADSTACK(OFF)
- TRAP(ON,SPIE)

If you need to change any of these options, you can specify a new value by setting the _CEE_RUNOPTS environment variable in the transform configuration file, `aopxfd.conf`.

For example, to change the ERRCOUNT value, specify this environment variable: `_CEE_RUNOPTS -> 'ERRCOUNT(4)'`

Guideline: Do not specify smaller initial allocation sizes in the ANYHEAP, HEAP, STACK, and THREADSTACK options. Setting smaller values can cause severe performance degradation.

For more information, see *z/OS XL C/C++ Programming Guide*

Customizing the PDF to AFP and PostScript to AFP transforms

To customize the PDF to AFP and PostScript to AFP transform, do the tasks listed in this table. Required tasks are required by all installations. The customization tasks are the same for both transforms. Optional tasks are required only if the listed condition applies to your installation.

Tip: The same transform daemon (`ps2afpd`) transforms both PDF and PostScript data streams. Therefore, the customization tasks for the PDF to AFP transform and the PostScript to AFP transform are the same.

Task	Condition
"Creating an entry in the transform configuration file" on page 60	Required
"Starting Infoprint Server with sufficient memory" on page 67	Required
"Setting up security" on page 68	Required
"Adding fonts" on page 69	Optional: To add custom fonts

Creating an entry in the transform configuration file

You must create at least one transform entry in the Infoprint Server transform configuration file (`aopxfd.conf`) for the PDF to AFP and PostScript to AFP transform. In the transform entry, you specify:

- Environment variables that control the transform
- Attributes that control how the Infoprint Server Transform Manager manages the transform

For information about how to create and edit the Infoprint Server transform configuration file, see *z/OS Infoprint Server Customization*. After you update the transform configuration file, you must restart the Infoprint Server Transform Manager.

Tip: After you restart the Infoprint Server Transform Manager, check for error messages in the transform's `standard error`. If you find any error messages, fix the errors and restart the Infoprint Server Transform Manager. For more information about how to find the transform message logs, see "Finding the transform stderr file" on page 105.

Transform classes

If you want to specify different transform options for different transform situations, you can create different classes of the transform. For example, you could create a separate class that turns tracing on. You could use this transform class if IBM directs you to trace the transform.

For each transform class, you must create a separate transform entry. You select a name for the transform class in the transform entry (see the transform attribute).

To use a transform class, job submitters must specify the class name in the -c option on the pdf2afp or ps2afp transform command or in the filter-options job attribute, and administrators must specify the class name in the -c filter option in the printer definition. For example, these z/OS UNIX commands use the "trace" transform class:

```
pdf2afp -c trace -o myfile.afp myfile.pdf
lp -d myprinter -o "filter-options='-c trace'" myfile.pdf
```

Tip: The sample transform configuration file, /usr/lpp/Printsrv/samples/aopxfd.conf, shows examples of transform entries with different transform classes. Also, see "Examples -- Transform configuration file entries for the PDF to AFP and PostScript to AFP transforms" on page 66.

Format of a PDF to AFP and PostScript to AFP transform entry

The format of a PDF to AFP and PostScript to AFP transform entry in the transform configuration file (aopxfd.conf) is:

```
transform ps2afp[_transformclass]
    start-command = "ps2afpd [-m nnnnnn[K|M]]"
    [ environment = {name -> value [ name -> value]… } ]
    [ maximum-active = number ]
    [ maximum-idle-time = seconds ]
    [ minimum-active = number ]
   ;
```

transform ps2afp[_*transformclass*]
This statement indicates the beginning of a transform entry.

ps2afp
The name of the PDF and PostScript to AFP transform.

transformclass
The name of an optional transform class. Specify from 1 to 63 characters, including letters, numbers, or special characters. The name of the transform class is case-sensitive.

Default: No transform class.

Example: transform ps2afp_trace

start-command = "ps2afpd [-m *nnnnnn*[K|M]]"
This attribute names the transform daemon and option. Enclose the value in single or double quotation marks if you specify the option.

ps2afpd
The name of the transform daemon. If the transform daemon is not in a directory identified in the PATH environment variable in the aopstart EXEC, specify the full directory path name of the daemon. (The ps2afpd daemon is installed in /usr/lpp/IBM/XformsToAFP/V2R4/bin, with a symbolic link from /usr/lpp/Printsrv/bin.) This attribute is required.

-m *nnnnnn*[K|M]
The maximum number of bytes of memory the transform daemon uses to do transforms. Specify the number of bytes in either kilobytes (K) or megabytes (M).

Suggested values are:

15M to 2047M
15360K to 1048576K

The amount of memory the transform requires depends on the complexity of the documents to be transformed. If you specify too low a value, the transform of complex PDF or PostScript documents might fail with undefined PostScript errors. If this occurs, try not specifying this option.

You must specify a maximum region size that is at least 10 MB greater than the value you specify in this option. For example, if you specify a value of 246M, set the region size to at least 256 MB

(246 MB + 10 MB). For information about how to specify the region size, see "Starting Infoprint Server with sufficient memory" on page 67.

Notes:

- Use of this option is not recommended.
- If you do use this option, specify a region size of at least 256 MB unless you routinely transform large or complex files. In that case, consider specifying a larger region size, such as 512 MB or 1024 MB, to improve performance and avoid errors. Otherwise, if the region size is too small to transform a file, the transform fails.

Default: None.

Example: `start-command = "ps2afpd -m 512M"`

environment = {*name* –> *value* [*name* –> *value*]... }
Environment variables that specify transform options. Enclose the entire set of environment variables in braces. The values in these environment variables override environment variables with the same name that are set in the aopstart EXEC. If a value contains special characters or spaces, enclose the value in single or double quotation marks.

For information about the environment variables you can specify, see "Environment variables for the PDF to AFP and PostScript to AFP transforms" on page 62.

Default: None

Example:

```
environment = {AOP_RESOURCE_PATH –> /etc/Printsrv/ps2afpv2.4/
fonts}
```

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 the minimum-active attribute. For more information about the format of a transform entry, see *z/OS Infoprint Server Customization*.

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. For more information about the format of a transform entry, see *z/OS Infoprint Server Customization*.

Default: Idle transform daemons are not shut down.

minimum-active = *number*
The minimum number of transform daemons that the Infoprint Server Transform Manager activates concurrently. Specify a number less than or equal to the number specified in the maximum-active attribute. For more information about the format of a transform entry, see *z/OS Infoprint Server Customization*.

Default: minimum-active = 0

Environment variables for the PDF to AFP and PostScript to AFP transforms

Environment variables let you specify:

- Directory that contains transform resources
- Record length of AFP images
- Tracing options
- Whether the transform fails when warnings or data stream errors occur
- Whether the transform writes messages to a trailer error page

You can specify these environment variables. All environment variables are optional.

_BPX_JOBNAME
The job name for this transform. When you assign a different job name to each class of transform, the operator can manage the transform daemons more effectively. Specify a job name of 1 to 8 alphanumeric characters. Incorrect job names are ignored. For more information about the _BPX_JOBNAME variable, see *z/OS UNIX System Services Planning*.

Default: The job name is AOPXFD.

Example: `environment = {_BPX_JOBNAME -> PS2AFPD}`

_CEE_DMPTARG
The directory where Language Environment (LE) writes a CEEDUMP. You can specify a period (.) to write the CEEDUMP to the transform's current working directory. IBM customer service expects to find CEEDUMPs in the transform's current working directory.

The transform's current working directory is: *base-directory*/xfd/ps2afp.#.d. *base-directory* is the base directory specified in the Infoprint Server configuration file, `aopd.conf`. An example of the transform's current working directory is:

```
/var/Printsrv/xfd/ps2afp.0.d
```

If you specify a directory other than the transform's current working directory, make sure that the user ID that owns the ps2afpd file has write access to the directory. For example, if user PS2AFPD owns transform file ps2afpd, user PS2AFPD must have write access to the directory.

Default: LE writes the dump to the directory specified in the TMPDIR environment variable. If the TMPDIR environment variable is not specified, LE writes the dump to the `/tmp` directory.

Example: `environment = {_CEE_DMPTARG -> .}`

Note: For other transforms, such as the PCL to AFP transform, LE writes the CEEDUMP to the transform's current working directory by default. This is because the owners of the other transforms have write access to the Infoprint Server base directory. Therefore, you do not need to specify the _CEE_DMPTARG environment variable for other transforms. For more information about how LE determines where to write the CEEDUMP, see *z/OS Language Environment Debugging Guide*.

AOP_FAIL_ON_ERROR
Specifies whether the transform stops processing when a warning or data stream error occurs during the transform. Warnings can indicate degraded output.

Valid values are:

error
The transform fails when a data stream error occurs and does not create an output document. However, the transform continues if warnings occur.

no
The transform continues processing when a warning or data stream error occurs and creates an output document. This is the default.

warning
The transform fails when a warning or data stream error occurs and does not create an output document.

Tips:

1. For backward compatibility, value yes is allowed and is equivalent to error.
2. The fail-on-transform-error job attribute and the **Fail on error** field in the printer definition override this value.
3. You can specify the value using lowercase or uppercase letters (for example, no or NO).

Default: AOP_FAIL_ON_ERROR –> no

Example: `environment = {AOP_FAIL_ON_ERROR -> error}`

errorhandling

AOP_FONT_SUBSTITUTION_MESSAGES
Indicates whether the transform writes a message (AOP2500W) when it substitutes fonts in a document. However, if another warning or an error occurs during the transform, the transform writes the font-substitution message regardless of the setting of this environment variable because the font-substitution message might help you diagnose problems.

Font-substitution messages let you determine if the transform has used substitute fonts. If a substitute font is not acceptable, you might be able to add the original font to the transform. For information, see "Adding fonts" on page 69.

Valid values are:

yes
The transform writes messages about font substitution.

no
The transform does not write messages about font substitution.

Tip: If the `AOP_TRAILER_ERROR_PAGE -> error` or `AOP_TRAILER_ERROR_PAGE -> no` environment variable is set, the transform does not write the font-substitution message to a trailer error page regardless of the setting of the AOP_FONT_SUBSTITUTION_MESSAGES environment variable.

Default: AOP_FONT_SUBSTITUTION_MESSAGES -> yes

Example: environment = {AOP_FONT_SUBSTITUTION_MESSAGES -> no}

AOP_RECLEN
The length of the AFP images the transform creates. Each output record contains an AFP image and an additional 17 bytes of information (for example, header information). For example, if you specify a value of 8 (8,000 bytes) for the AFP image in this environment variable, the length of each output record is 8,017 bytes.

Valid values are from 8 to 32, in units of 1,000 bytes. For example, 8 = 8,000.

Tip: The default value is suitable for most installations. However, a larger record length might improve performance.

Default: `AOP_RECLEN -> 8`

Example: `environment = {AOP_RECLEN -> 16}`

AOP_RESOURCE_PATH
The directories that contains fonts and other transform resources. If the fonts and other resources are in more than one directory, list all directories separated by a colon. The directories are searched in the order that they are listed. Do not list the default resource directories because the transform always searches these directories last. For more information, see "Adding fonts" on page 69.

Default:

```
AOP_RESOURCE_PATH -> /usr/lpp/Printsrv/ps2afpv2.4/lib/:/usr/lpp/Printsrv/
ps2afpv2.4/Resource/Init:/usr/lpp/Printsrv/ps2afpv2.4/Resource/
Font:/usr/lpp/Printsrv/ps2afpv2.4/Resource/
```

Example:

```
environment = {AOP_RESOURCE_PATH -> /etc/Printsrv/ps2afpv2.4/fonts}
```

AOP_TRAILER_ERROR_PAGE
Specifies whether the transform writes messages for transform warnings and data stream errors to a trailer error page. The trailer error page is the last page in the output document. Warnings can indicate degraded output.

Valid values are:

error
The transform writes error messages, but not warning messages, to a trailer error page. This option does not apply when AOP_FAIL_ON_ERROR–> error or AOP_FAIL_ON_ERROR–>warning because no output document is created when an error occurs.

no
The transform does not write messages to a trailer error page.

warning
The transform writes warning and error messages to a trailer error page. This option does not apply when AOP_FAIL_ON_ERROR–>warning because no output document is created when an error or warning occurs. This is the default.

Tips:

1. The trailer-transform-error-page job attribute and the **Trailer error page** field in the printer definition override this value.
2. You can specify the value using lowercase or uppercase letters (for example, no or NO).

Default: AOP_TRAILER_ERROR_PAGE –> warning

Example: environment = {AOP_TRAILER_ERROR_PAGE –> error}

AOPTRACEDIR
The full path name of the directory where the transform writes trace information. You can specify the same directory for different transform classes. The name of the trace file identifies the transform and transform class, and contains a time stamp.

Defaults:

- When AOPTRACEON is turned on or when the permissions of /var/Printsrv/trace are drwxrwxrwx or drwxrwx-wx, AOPTRACEDIR -> /var/Printsrv/trace
- Otherwise, AOPTRACEDIR -> /var/Printsrv/xfd

When trace information is written to /var/Printsrv/xfd:

- The trace information is appended to the ps2afp.#.stderr file for each instance of the transform. The stderr file is automatically cleaned up when you restart Infoprint Server Transform Manager.
- RACF message ICH408I is written to the log. You can ignore it.

Examples:

```
environment = {AOPTRACEDIR -> /var/Printsrv/xfd}
environment = {AOPTRACEDIR -> .}
```

AOPTRACEON
Turns tracing on. The transform traces all transform requests that use this transform class. Any value turns tracing on. To turn tracing off, do not specify this environment variable. Specify this environment variable only if IBM directs you to do so. Tracing can adversely affect performance.

Default: Tracing is turned off.

Example: environment = {AOPTRACEON -> 1}

AOP_TRIM
Indicates whether the transform creates trimmed AFP images or full-page AFP images. Trimmed AFP images are smaller than full-page AFP images. Therefore, AFP output files that contain trimmed AFP images are smaller and can be transmitted more quickly over a network.

In contrast to full-page AFP images, trimmed AFP images do not include any white space that surrounds the data on the page. The transform positions trimmed AFP images on the page so that the data prints in the correct position on the page. For example, if a page is 8 1/2 by 11 inches and has a 1 inch margin on all sides:

- A trimmed AFP image is 6 1/2 x 9 inches. It is positioned at offset 1 inch (x), 1 inch (y) on the page.
- A full-page AFP image is 8 1/2 by 11 inches. It is positioned at offset 0 (x), 0 (y) on the page.

AFP documents print the same whether you specify AOP_TRIM = yes or AOP_TRIM = no. However, AFP overlays and AFP page segments typically print differently because trimmed AFP images are smaller than full-page AFP images. A page segment that contains a trimmed AFP image typically prints higher and to the left on a page. (The -t transform option determines the type of output. For example, `-t pagesegment` creates a page segment.)

Values are:

yes
The transform creates trimmed AFP images (default).

no
The transform creates full-page AFP images.

The V1 transforms always create full-page AFP images. To create full-page AFP images in V2, you must specify the AOP_TRIM=no environment variable.

Tip: In addition to the environment variables listed in this information, you can specify other environment variables. For example, you can specify the _CEE_RUNOPTS environment variable. For information, see "Changing Language Environment run-time options" on page 66.

Changing Language Environment run-time options

The transform sets these run-time options for the Language Environment (LE):

- ALL31(ON)
- ANYHEAP(16K,8K,ANYWHERE,FREE)
- BELOWHEAP(8K,4K,FREE)
- DEPTHCONDLMT(10)
- ERRCOUNT(0)
- HEAP(512K,512K,ANYWHERE,FREE)
- LIBSTACK(8K,4K,FREE)
- STACK(128K,128K,ANYWHERE,KEEP,512K,128K)
- STORAGE(NONE,NONE,NONE,8K)
- TERMTHDACT(DUMP)
- THREADHEAP(8K,8K,ANYWHERE,KEEP)
- THREADSTACK(OFF)
- TRAP(ON,SPIE)

If you need to change any of these options, you can specify a new value by setting the _CEE_RUNOPTS environment variable in the transform configuration file, `aopxfd.conf`.

For example, to change the ERRCOUNT value, specify this environment variable: `_CEE_RUNOPTS -> 'ERRCOUNT(4)'`

Guideline: Do not specify smaller initial allocation sizes in the ANYHEAP, HEAP, STACK, and THREADSTACK options. Setting smaller values can cause severe performance degradation.

For more information, see *z/OS XL C/C++ Programming Guide*.

Examples -- Transform configuration file entries for the PDF to AFP and PostScript to AFP transforms

This information shows sample transform entries in the Infoprint Server transform configuration file (`aopxfd.conf`) for the PDF to AFP and PostScript to AFP transforms.

Specify a transform resource path

This transform entry specifies a resource directory that your installation created to contain custom fonts. This transform does not create a transform class. Therefore, to use this transform, do not specify a transform class in the -c transform option.

```
transform ps2afp
      start-command = ps2afpd
      maximum-active = 2
      maximum-idle-time = 300   # 5 minutes
      minimum-active = 1
      environment = {
        _BPX_JOBNAME -> PS2AFPD
        _CEE_DMPTARG -> . # CEEDUMPs in current working directory
        AOP_RESOURCE_PATH -> /etc/Printsrv/ps2afpv2.4/fonts
      }
;
```

Trace the transform

This transform entry creates transform class trace, traces transform jobs, writes trace information to a file in the transform's working directory, `ps2afp_trace.#.d`, and appends a message pointing to the trace information file to the `ps2afp.#.stderr` file. Both the working directory and the `stderr` file are in the default directory, `/var/Printsrv/xfd`:

```
transform ps2afp_trace
      start-command = ps2afpd
      maximum-active = 2
      maximum-idle-time = 300   # 5 minutes
      minimum-active = 1
      environment = {
        _BPX_JOBNAME -> PS2AFPD
        _CEE_DMPTARG –> . # CEEDUMPs in current working directory
        AOPTRACEON ->1
        AOPTRACEDIR -> .
      }
;
```

To use transform class trace, specify the class in the -c transform option as shown in these two examples:

```
pdf2afp -c trace -o myfile.afp myfile.pdf
lp -o "filter-options='-c trace'" -d myprinter myfile.pdf
```

Starting Infoprint Server with sufficient memory

The region size available for Infoprint Server should be 256 MB or more so that you can transform large or complex data streams. If you specify the -m option in the transform configuration file (not recommended), the region size must be at least 10 MB larger than the number of bytes specified in the -m option. For information about the -m option, see “Format of a PDF to AFP and PostScript to AFP transform entry” on page 61.

To make sure that sufficient memory is available when you start the transform:

- In the AOPSTART procedure, specify a region size of at least 256 MB in the REGION parameter on the EXEC statement. If the REGION parameter is not specified, the default region size defined for your installation is used.
- Make sure the maximum address space size for Infoprint Server is at least 256 MB.

  You can set the system-wide maximum address space size in the BPXPRMxx member or with the SETOMVS command.

  Examples:

  1. This statement in the BPXPRMxx member sets the maximum region size to 256 MB:

     ```
     MAXASSIZE(268435456)  /* 256*1024*1024 = 256MB    */
     ```

2. This command sets the maximum region size to 256 MB:

   ```
   SETOMVS MAXASSIZE=268435456
   ```

3. This command checks the MAXASSIZE value from the operator console:

   ```
   d omvs,o
   ```

You can set the maximum address space for Infoprint Server higher than the system-wide maximum address in the ASSIZEMAX parameter of the RACF ADDUSER and ALTUSER command for the user that starts Infoprint Server. The ASSIZEMAX value overrides the MAXASSIZE value.

If you enter the `aopstart` command from the z/OS UNIX command line during a TSO session, the SIZE parameter on the TSO/E LOGON panel determines the maximum region size for an address space. Specify a SIZE parameter of at least 256 MB.

- 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*.

Setting up security

Security checking done in the transform requires that the user identifier (UID) of the executable file for the transform, `ps2afpd`, not be 0 (zero). When it is installed, file `ps2afpd` has a UID of 0. Therefore, you must change the owner of the file. The new owner must have a UID that is not 0 and not the default UID. Also, the set-user-ID flag for the file must be turned on.

First use the Resource Access Control Facility (RACF), or another program that follows system authorization facility (SAF) protocol, to create a user and group profile for the owner of `ps2afpd`. Then, change the owner of `ps2afpd` and turn on the set-user-ID flag.

The `ps2afpd` is installed with the extrattr +p program-control bit set on, because of a UNIX System Services LE-related security fix.

Steps for establishing security:

1. Define a group to RACF.

   The group profile must have an OMVS segment and a group identifier (GID). You can use any group name. You should not give this group any authority to the z/OS file system.

   This group must not have any additional authority, such as authority to BPX.SUPERUSER or BPX.DAEMON.

   For example, this RACF command defines group PS2AFPDG. For *ps2afpdg-gid*, specify an integer that is different from other GIDs in your installation:

   ```
   ADDGROUP (PS2AFPDG) OMVS(GID(ps2afpdg-gid))
   ```

2. Define a user to RACF as a z/OS UNIX user.

   This user will be the owner of `ps2afpd`. The transform, as well as PostScript jobs being transformed, run with the UID of this user. For complete information about how to define z/OS UNIX users to RACF, see *z/OS UNIX System Services Planning*.

   The user profile must have an OMVS segment. Its UID must not be 0 and not be the default UID, which is defined in the BPX.DEFAULT.USER profile in the RACF FACILITY class. You can use any user name. For example, you can use PS2AFPD.

   You should not give this user authority to the z/OS file system. However, if any PostScript jobs to be transformed require access to certain files, you can give this user access to the required files or connect this user to another group (or groups) that has access to the required files.

   This user must not have any additional authority, such as authority to BPX.SUPERUSER or BPX.DAEMON.

For example, this RACF command defines user PS2AFPD. For *ps2afpd-uid*, specify an integer that is different from other UIDs in your installation:

```
ADDUSER (PS2AFPD) OMVS(UID(ps2afpd-uid))
```

3. Connect the user to the group.

   Connect the user defined in step 2 to the group defined in step 1. For example, this RACF command connects user PS2AFPD to the PS2AFPDG group:

   ```
   CONNECT (PS2AFPD) GROUP(PS2AFPDG)
   ```

4. Switch to an effective UID of 0:

   ```
   su
   ```

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

5. Assign the user as the owner of file `ps2afpd`.

   Use the `chown` command to assign the user defined in step 2 as the owner of `ps2afpd`. For example, type this command on the z/OS UNIX command line to assign user PS2AFPD as the owner:

   ```
   chown PS2AFPD /usr/lpp/Printsrv/bin/ps2afpd
   ```

6. Turn the set-user-ID flag on for file `ps2afpd`.

   The `chown` command turns off the set-user-ID flag. Therefore, use the `chmod` command to turn this flag on again. For example, type this command on the z/OS UNIX command line:

   ```
   chmod u+s /usr/lpp/Printsrv/bin/ps2afpd
   ```

7. If you switched to an effective UID of 0 in step 4, switch back to your own UID:

   ```
   exit
   ```

After doing these steps, you can use the `ls` command to list the owner of `ps2afpd` and to verify that the set-user-ID flag is on. For example, type on the z/OS UNIX command line:

```
ls -l /usr/lpp/Printsrv/bin/ps2afpd
```

Assuming that you assigned user PS2AFPD as the owner of `ps2afpd`, output from the `ls` command should look like this:

```
-rws------ 1 PS2AFPD PS2AFPDG ... /usr/lpp/Printsrv/bin/ps2afpd
```

The lowercase letter s in the owner permissions section indicates that the set-user-ID flag is on and that the owner has permission to execute the file.

For more information about RACF, see *z/OS Security Server RACF Security Administrator's Guide*. For more information about the `chown`, `chmod`, and `ls` commands, see *z/OS UNIX System Services Command Reference*.

Adding fonts

The PDF to AFP and PostScript to AFP transforms provide fonts in the `/usr/lpp/IBM/XformsToAFP/V2R4/ps2afpv2.4/Resource/Font` directory. If the PDF and PostScript files you transform use other fonts, you can add the fonts to the transforms. You can add these types of fonts:

- Type 0, 1, 3, 4, and 42 fonts
- MultiMaster fonts

Rules:

1. Fonts must be in ASCII representation. Postscript font files in both PFA and PFB format are supported.
2. The font file name can have a suffix. However, a suffix is not required.
3. If the font file name is not exactly the same as the font name, you must specify the font file name in the `Fontmap.GS` file. (If the font file name is the same as the font name, the transform finds the font even if the font name is not in the `Fontmap.GS` file.)
4. The data in the `Fontmap.GS` file must be in ASCII representation.

Steps for adding fonts:

1. Create a directory for the fonts. Set the z/OS UNIX permissions so that the owner can read, write, and access the directories, and everyone can read and access them. For example:

```
mkdir /etc/Printsrv/ps2afpv2.4
chmod 755 /etc/Printsrv/ps2afpv2.4
mkdir /etc/Printsrv/ps2afpv2.4/fonts
chmod 755 /etc/Printsrv/ps2afpv2.4/fonts
```

2. Add fonts to the fonts directory created in step 1. Set the z/OS UNIX permissions so that the owner can read and write the font files, and everyone can read them. For example:

```
chmod 644 /etc/Printsrv/ps2afpv2.4/fonts/*
```

3. (Optional) If the names of the font files are not the same as the font names, edit the `Fontmap.GS` file. First, copy the file from directory `/usr/lpp/IBM/XformsToAFP/V2R4/ps2afpv2.4/Resource/Init` to the fonts directory created in step 1. For example:

```
cp /usr/lpp/IBM/XformsToAFP/V2R4/ps2afpv2.4/Resource/Init/Fontmap.GS /etc/Printsrv/ps2afpv2.4/fonts/
Fontmap.GS
```

   Then follow the instructions in the file to specify the names and file names of the fonts you added.

   Tip: The data in the `Fontmap.GS` file must be in ASCII representation. To edit this file in EBCDIC on the z/OS system, you can use the `iconv` command to convert between ASCII and EBCDIC. These steps show how to convert data between EBCDIC and ASCII code pages, then convert data in the `Fontmap.GS` file:

   a. Convert data in `Fontmap.GS` from ASCII to EBCDIC and create file `Fontmap.e`:

```
iconv -f iso8859-1 -t ibm-1047 Fontmap.GS > Fontmap.e
```

   b. Edit file `Fontmap.e`.

   c. Convert data in `Fontmap.e` from EBCDIC to ASCII and replace data in file `Fontmap.GS`:

```
iconv -f ibm-1047 -t iso8859-1 Fontmap.e > Fontmap.GS
```

4. Specify the directory that contains the fonts you added and the modified `Fontmap.GS` file in the AOP_RESOURCE_PATH environment variable in the transform configuration file, `aopxfd.conf`. For example:

```
environment = {AOP_RESOURCE_PATH -> /etc/Printsrv/ps2afpv2.4/fonts}
```

5. Restart the Infoprint Server Transform Manager to pick up the changes to the `Fontmap.GS` and `aopxfd.conf` files.

Customizing the SAP to AFP transform

To customize the SAP to AFP transform, do the tasks listed in this table. Required tasks are required by all installations. Optional tasks are required only if the listed condition applies to your installation.

Task	Condition
“Creating an entry in the transform configuration file” on page 71	Required
“Customizing SAP to AFP configuration files” on page 76	Optional: To customize the SAP to AFP transform
“Customizing multibyte conversion tables” on page 91	Optional: To customize multibyte conversion tables

Creating an entry in the transform configuration file

You must create at least one transform entry in the Infoprint Server transform configuration file (`aopxfd.conf`) for the SAP to AFP transform. In the transform entry, you specify:

- Environment variables that control the transform
- Attributes that control how the Infoprint Server Transform Manager manages the transform

For information about how to create or edit the Infoprint Server transform configuration file, see *z/OS Infoprint Server Customization*. After you update the transform configuration file, you must restart the Infoprint Server Transform Manager.

Tip: After you restart the Infoprint Server Transform Manager, check for error messages in the transform's `standard error`. If you find any error messages, fix the errors and restart the Infoprint Server Transform Manager. For more information about how to find the transform message logs, see “Finding the transform stderr file” on page 105.

Transform classes

If you want to specify different transform options or environment variable for different print applications, you can create different classes of the transform. For each transform class, you must create a separate transform entry. You select a name for the transform class in the transform entry (see the transform attribute).

To use a transform class, job submitters must specify the class name in the -c option on the `sap2afp` command or in the filter-options job attribute, and administrators must specify the class name in the -c filter option in the printer definition. For example, these z/OS UNIX commands use the "myfonts" transform class:

```
sap2afp -c myfonts -o myfile.afp myfile.sap
lp -d myprinter -o "filter-options='-c myfonts'" myfile.sap
```

Tip: The sample transform configuration file, `/usr/lpp/Printsrv/samples/aopxfd.conf`, shows examples of transform entries with different transform classes. Also, see “Examples -- Transform configuration file entries for the SAP to AFP transform” on page 75.

Format of an SAP to AFP transform entry

The format of an SAP to AFP transform entry in the transform configuration file (`aopxfd.conf`) is:

```
transform sap2afp[_transformclass]
    start-command = "sap2afpd "
    [ environment  = {name -> value [ name -> value]… } ]
    [ maximum-active = number ]
    [ maximum-idle-time = seconds ]
    [ minimum-active = number ]
    ;
```

transform sap2afp[_*transformclass*]
This statement indicates the beginning of a transform entry.

sap2afp
The name of the transform.

transformclass
The name of an optional transform class. Specify from 1 to 63 characters, including letters, numbers, or special characters. The name of the transform class is case-sensitive.

Default: No transform class.

Examples:

```
transform sap2afp
```

```
transform sap2afp_myfonts
```

start-command = sap2afpd
This attribute names the transform daemon.

sap2afpd
The name of the transform daemon. If the transform daemon is not in a directory identified in the PATH environment variable in the `aopstart` EXEC, specify the full directory path name of the daemon. (The `sap2afpd` daemon is installed in `/usr/lpp/Printsrv/bin`.) This attribute is required.

environment = {*name* –> *value* [*name* –> *value*]... }
Environment variables that specify transform options. Enclose the entire set of environment variables in braces. The values in these environment variables override environment variables with the same name that are set in the `aopstart` EXEC. If a value contains special characters or spaces, enclose the value in single or double quotation marks.

For information about the environment variables you can specify, see "Environment variables for the SAP to AFP transform" on page 72.

Default: None.

Example:

```
environment = {AOP_SAP2AFP_RESOURCES -> /etc/Printsrv/sap2afpv2.4/
res}
```

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 the minimum-active attribute. For more information about the format of a transform entry, see *z/OS Infoprint Server Customization*.

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. For more information about the format of a transform entry, see *z/OS Infoprint Server Customization*.

Default: Idle transform daemons are not shut down.

minimum-active = *number*
The minimum number of transform daemons that the Infoprint Server Transform Manager activates concurrently. Specify a number less than or equal to the number specified in the maximum-active attribute. For more information about the format of a transform entry, see *z/OS Infoprint Server Customization*.

Default: `minimum-active = 0`

Environment variables for the SAP to AFP transform

Environment variables let you specify:

- Transform job name

- Error handling option
- Resource path
- Tracing options
- Whether the transform fails when warnings or data stream errors occur
- Whether the transform writes messages to a trailer error page

You can specify these environment variables. All environment variables are optional.

_BPX_JOBNAME
The job name for this transform. When you assign a different job name to each class of transform, the operator can manage the transform daemons more effectively. Specify a job name of 1 to 8 alphanumeric characters. Incorrect job names are ignored. For more information about the _BPX_JOBNAME variable, see *z/OS UNIX System Services Planning*.

Default: The job name is AOPXFD.

Example: `environment = {_BPX_JOBNAME -> SAP2AFPD}`

AOP_FAIL_ON_ERROR
Specifies whether the transform stops processing when a warning or data stream error occurs during the transform. Warnings can indicate degraded output.

Valid values are:

error
The transform fails when a data stream error occurs and does not create an output document. However, the transform continues if warnings occur.

no
The transform continues processing when a warning or data stream error occurs and creates an output document. This is the default.

warning
The transform fails when a warning or data stream error occurs and does not create an output document.

Tips:

1. For backward compatibility, value yes is allowed and is equivalent to error.
2. The fail-on-transform-error job attribute and the **Fail on error** field in the printer definition override this value.
3. You can specify the value using lowercase or uppercase letters (for example, no or NO).

Default: `AOP_FAIL_ON_ERROR -> no`

Example: `environment = {AOP_FAIL_ON_ERROR -> error}`

AOP_RECLEN
The length of the output records that the transform creates.

Valid values are from 8 to 32, in units of 1,000 bytes. For example, 8 = 8,000.

Tip: The default value is suitable for most installations. However, a larger record length might improve performance.

Default: AOP_RECLEN -> 8

Example: `environment = {AOP_RECLEN -> 16}`

AOP_SAP2AFP_ICU
The directory that contains all of the SAP to AFP transform double-byte conversion tables. You can specify only one directory name. If the conversion tables are in the default directory, `/usr/lpp/Printsrv/sap2afpv2.4/icu`, you do not need to set this environment variable.

Default: AOP_SAP2AFP_ICU -> /usr/lpp/Printsrv/sap2afpv2.4/icu

Example:

```
environment = {AOP_SAP2AFP_ICU -> /etc/Printsrv/sap2afpv2.4/icu}
```

AOP_SAP2AFP_RESOURCES

The directory that contains all of the SAP to AFP transform resources, such as configuration files. You can specify only one directory name. If the Infoprint Server Transforms resources are in the default directory, `/usr/lpp/Printsrv/sap2afpv2.4/res`, you do not need to set this environment variable.

Default: `AOP_SAP2AFP_RESOURCES -> /usr/lpp/Printsrv/sap2afpv2.4/res`

Example:

```
environment = {AOP_SAP2AFP_RESOURCES -> /etc/Printsrv/
sap2afpv2.4/res}
```

AOP_TRAILER_ERROR_PAGE

Specifies whether the transform writes messages for transform warnings and data stream errors to a trailer error page. The trailer error page is the last page in the output document. Warnings can indicate degraded output.

Valid values are:

error
The transform writes error messages, but not warning messages, to a trailer error page. This option does not apply when `AOP_FAIL_ON_ERROR–> error` or `AOP_FAIL_ON_ERROR–>warning` because no output document is created when an error occurs.

no
The transform does not write messages to a trailer error page.

warning
The transform writes warning and error messages to a trailer error page. This option does not apply when `AOP_FAIL_ON_ERROR–>warning` because no output document is created when an error or warning occurs. This is the default.

Tips:

1. The trailer-transform-error-page job attribute and the **Trailer error page** field in the printer definition override this value.
2. You can specify the value using lowercase or uppercase letters (for example, no or NO).

Default: `AOP_TRAILER_ERROR_PAGE –> warning`

Example: `environment = {AOP_TRAILER_ERROR_PAGE –> error}`

AOPTRACEDIR

The full path name of the directory where the transform writes trace information. You can specify the same directory for different transform classes. The name of the trace file identifies the transform and transform class, and contains a time stamp.

This directory must already exist. If the directory does not exist, the transform writes trace information to the transform's `stderr` file. For information about how to find the `stderr` file, see “Finding the transform stderr file” on page 105.

Default: `AOPTRACEDIR -> /var/Printsrv/trace`

Examples:

```
environment = {AOPTRACEDIR -> /var/Printsrv/xfd}
environment = {AOPTRACEDIR -> .}
```

AOPTRACEON
Turns tracing on. The transform traces all transform requests that use this transform class. Any value turns tracing on. To turn tracing off, do not specify this environment variable. Specify this environment variable only if IBM directs you to do so. Tracing can adversely affect performance.

Default: Tracing is turned off.

Example: `environment = {AOPTRACEON -> 1}`

AOP_WORLDTYPE_PATH
The directory that contains WorldType TrueType fonts. You can specify only one directory name. If the WorldType fonts are installed in the default directory, `/usr/lpp/fonts/worldtype`, you do not need to set this environment variable.

Default: `/usr/lpp/fonts/worldtype`

Example: `AOP_WORLDTYPE_PATH -> /u/Printsrv/worldtypefonts`

Tip: In addition to the environment variables listed in this information, you can specify other environment variables. For example, you can specify the _CEE_RUNOPTS environment variable. For information, see "Changing Language Environment run-time options" on page 75.

Changing Language Environment run-time options

The transform sets these run-time options for the Language Environment (LE):

- ALL31(ON)
- ANYHEAP(16K,8K,ANYWHERE,FREE)
- BELOWHEAP(8K,4K,FREE)
- DEPTHCONDLMT(10)
- ERRCOUNT(0)
- HEAP(512K,512K,ANYWHERE,FREE)
- LIBSTACK(8K,4K,FREE)
- STACK(128K,128K,ANYWHERE,KEEP,512K,128K)
- STORAGE(NONE,NONE,NONE,8K)
- TERMTHDACT(DUMP)
- THREADHEAP(8K,8K,ANYWHERE,KEEP)
- THREADSTACK(OFF)
- TRAP(ON,SPIE)

If you need to change any of these options, you can specify a new value by setting the _CEE_RUNOPTS environment variable in the transform configuration file, `aopxfd.conf`.

For example, to change the ERRCOUNT value, specify this environment variable: `_CEE_RUNOPTS -> 'ERRCOUNT(4)'`

Guideline: Do not specify smaller initial allocation sizes in the ANYHEAP, HEAP, STACK, and THREADSTACK options. Setting smaller values can cause severe performance degradation.

For more information, see *z/OS XL C/C++ Programming Guide*

Examples -- Transform configuration file entries for the SAP to AFP transform

This example shows a transform entry in the Infoprint Server transform configuration file (`aopxfd.conf`) for the SAP to AFP transform. This entry does not create a transform class.

```
transform sap2afp
     start-command = sap2afpd
     maximum-active = 2
     maximum-idle-time = 300   # 5 minutes
     minimum-active = 1
     environment = {
```

```
        _BPX_JOBNAME -> SAP2AFPD
    }
;
```

This example shows how to specify the AOP_SAP2AFP_RESOURCES environment variable in the transform entry. This example assumes that the transform configuration files are in directory `/etc/Printsrv/sap2afpv2.4/res`. This entry does not create a transform class.

```
transform sap2afp
     start-command = sap2afpd
     maximum-active = 2
     maximum-idle-time = 300   # 5 minutes
     minimum-active = 1
     environment = {
         _BPX_JOBNAME -> SAP2AFPD
         AOP_SAP2AFP_RESOURCES -> /etc/Printsrv/sap2afpv2.4/res
     }
;
```

Customizing SAP to AFP configuration files

Table 6 on page 76 lists the SAP to AFP transform configuration files that you can customize. For instructions about customizing SAP configuration files, see “Steps for customizing SAP to AFP transform configuration files” on page 91.

Table 6: SAP to AFP transform configuration files

Configuration file	Purpose	SAP data stream
`barcode.tab`	Maps SAP OTF bar code names to Bar Code Object Content Architecture (BCOCA) bar codes.	OTF data only
`defcp.tab`	Maps single-byte ASCII code points to EBCDIC code points.	ABAP data only
`defcp.tab.cyr`	Maps single-byte ASCII code points to EBCDIC code points, and also contains some Cyrllic conversion directives.	ABAP data only
`defcp.tab.japan`	Maps single-byte ASCII code points to EBCDIC code points, and also contains some Japanese conversion directives.	ABAP data only
`defcp.tab.korea`	Maps single-byte ASCII code points to EBCDIC code points, and also contains some Korean conversion directives.	ABAP data only
`defcp.tab.spchinese`	Maps single-byte ASCII code points to EBCDIC code points, and also contains some Simplified Chinese conversion directives.	ABAP data only
`dcfcp.tab.tdchinese`	Maps single-byte ASCII code points to EBCDIC code points, and also contains some Traditional Chinese conversion directives.	ABAP data only
`defcp.tab.unicode`	Maps single-byte ASCII code points to EBCDIC code points, and also contains some Unicode conversion directives.	ABAP data only
`fonts.tab`	Maps the fonts used in the OTF data stream to AFP fonts.	OTF data only
`fonts.tab.unicode`	Maps the AFP outline fonts to be used for OTF files in a Unicode environment.	OTF data only

Table 6: SAP to AFP transform configuration files (continued)		
Configuration file	**Purpose**	**SAP data stream**
`fonts.tab.unicode.truetype`	Maps the TrueType fonts to be used for OTF files in a Unicode environment.	OTF data only
`fonts.tab.unicode.embed.truetype`	Maps the TrueType fonts to be used for OTF files in a Unicode environment, embedding the fonts in the document. This sample file must be copied and customized to be used by the transform, as described in Chapter 3, "Customizing transforms," on page 53.	OTF Only
`image.tab`	Defines values used to print image data.	OTF data only
`pagedef.tab`	For SAP R/3 format names, specifies the page definition, form definition, and ABAP coded fonts.	ABAP and OTF data
`pagedef.tab.cyr`	Used by ASCII-encoded ABAP files containing Cyrillic characters in a non-Unicode environment. The structure of the file is the same as the `pagedef.tab` file.	ABAP and OTF data
`pagedef.tab.japan`	Used by ASCII-encoded files containing Japanese characters. The structure of the file is the same as the `pagedef.tab` file.	ABAP and OTF data
`pagedef.tab.korea`	Used by ASCII-encoded files containing Korean characters. The structure of the file is the same as the `pagedef.tab` file.	ABAP and OTF data
`pagedef.tab.spchinese`	Used by ASCII-encoded files containing Simplified Chinese characters. The structure of the file is the same as the `pagedef.tab` file.	ABAP and OTF data
`pagedef.tab.tdchinese`	Used by ASCII-encoded files containing Tradional Chinese characters. The structure of the file is the same as the `pagedef.tab` file.	ABAP and OTF data
`pagedef.tab.unicode`	Specifies the form definitions to be used for ABAP and OTF input files, and page definitions to be used for ABAP files, in a Unicode environment.	ABAP and OTF files
`pagedef.tab.unicode.truetype`	Specifies the form definitions to be used with TrueType for ABAP and OTF input files, and page definitions to be used with TrueType for ABAP files, in a Unicode environment.	ABAP and OTF data
xxxx`0000.tab`	Maps SAP code pages to AFP code pages.	OTF data only
`6400fonts.tab`	Lists raster fonts that apply exclusively to the InfoPrint 6400 printer with the appropriate character set and code page.	OTF data only
`userenv`	Specifies the input character set.	ABAP and OTF

Rules:

1. All configuration files must reside in the same directory.
2. If the configuration files are not in the default directory, `/usr/lpp/Printsrv/sap2afpv2.4/res`, environment variable AOP_SAP2AFP_RESOURCES must specify the directory name.
3. TrueType fonts include a resource access table (RAT) for processing the fonts, but RAT processing of TrueType fonts occurs in the product that processes the AFP, for example, PSF. If you have your own RAT, you can configure PSF or another processor to use your RAT to process TrueType fonts. For more information, see the product documentation for the processor.

barcode.tab configuration file

The `barcode.tab` configuration file maps SAP R/3 Output Text Format (OTF) bar code names to AFP Bar Code Object Content Architecture (BCOCA) bar codes.

You need to edit this file in these situations:

- If you define a new OTF user bar code, you must add the bar code name to this file and map it to a BCOCA bar code.
- If you want to print an OTF bar code with a different width or wide-to-narrow ratio, you can specify the ModWidth and Ratio keywords for the SAP bar code.

You can specify these keywords and values in file `barcode.tab` file:

`BarCode = name`
This required keyword specifies the OTF bar code name (SAPBARCODE parameter of the BC OTF command). The maximum length of this field is 8 bytes. It can contain any value.

`Type = nnn`
This required keyword specifies the AFP bar code type as defined in *Data Stream and Object Architectures: Bar Code Object Content Architecture Reference* for the Barcode Type parameter of the **Barcode Data Descriptor** structured field. This value can be any decimal value between 0 and 999. The transform does not verify that this value maps to a valid hexadecimal barcode type.

`Mode=nnn`
This required keyword specifies the modifier byte in the Barcode Modifier parameter of the **Barcode Data Descriptor** structured field. Any numeric value between 0 and 999 is accepted.

`Flag = [0 | 128]`
This required keyword controls the printing of the Human Readable Interface (HRI) character. Specify one of these values:

0
Prints the HRI characters

128
Does not print the HRI characters

`FlagPdf=nnn`
This optional keyword is used only for PDF417 bar codes (Type = 030). The value is byte 5 of the Bar Code Symbol Data (BSA) from PDF417 Special-Function parameters. For information, see *Bar Code Object Content Architecture Reference*.

`ModWidth = nnn`
This optional keyword specifies the width of the narrowest defined bar code element (bar or space) in mils (that is, thousandths of an inch – 1/1000). Allowed values are 1– 254.

For bar code types that explicitly specify the module width, such as POSTNET and RM4SCC, this field is ignored.

Default: The printer default ModWidth

`Ratio = nnn`
This optional keyword specifies the ratio of the width of wide bar code elements to narrow bar code elements, when only two different size elements exist, that is, for a two-level bar code type. Express the ratio as a percentage. Allowed values are 100 – 500. However, the value should usually be

between 200 and 300. For example: 200 represents a ratio of 2 to 1, and 250 represents a ratio of 2.5 to 1.

Specify a value that is appropriate for the bar code type and for your printer. Many bar codes do not support ratio values, and other bar codes require that the ratio value fall within a specific range. If you specify a ratio value that is not appropriate for the printer, error messages are sent at printing time.

For bar code types that explicitly specify the module width, such as POSTNET and RM4SCC, this field is ignored.

Default: The default ratio for the bar code type.

Note: Verify any values you enter in this table. The transform does not verify values, and values that are not valid might result in errors.

Figure 4 on page 79 shows the SAP bar codes defined in the default `barcode.tab` configuration file.

```
BarCode = ARTNR    Type = 017 Mode = 002 Flag = 000
BarCode = AUFNR    Type = 017 Mode = 002 Flag = 000
BarCode = BARCLVS  Type = 001 Mode = 001 Flag = 000

BarCode = BC_C128B Type = 017 Mode = 002 Flag = 128
BarCode = BC_CD39  Type = 001 Mode = 001 Flag = 128
BarCode = BC_CD39C Type = 001 Mode = 002 Flag = 128
BarCode = BC_EAN13 Type = 009 Mode = 000 Flag = 128
BarCode = BC_EAN8  Type = 008 Mode = 000 Flag = 128
BarCode = BC_EANH  Type = 017 Mode = 002 Flag = 128
BarCode = BC_I25   Type = 012 Mode = 001 Flag = 128
BarCode = BC_I25C  Type = 012 Mode = 002 Flag = 128
BarCode = BC_MSI   Type = 002 Mode = 001 Flag = 128
BarCode = BC_MSIC  Type = 002 Mode = 002 Flag = 128
BarCode = BC_MSIC1 Type = 002 Mode = 003 Flag = 128
BarCode = BC_MSIC2 Type = 002 Mode = 005 Flag = 128
BarCode = BC_PSN5  Type = 024 Mode = 000 Flag = 128
BarCode = BC_PSN9  Type = 024 Mode = 001 Flag = 128
BarCode = BC_PSN11 Type = 024 Mode = 002 Flag = 128

BarCode = C128A_00 Type = 017 Mode = 002 Flag = 000
BarCode = C128A_01 Type = 017 Mode = 002 Flag = 000
BarCode = C128B_00 Type = 017 Mode = 002 Flag = 000
BarCode = C128B_01 Type = 017 Mode = 002 Flag = 000
BarCode = CD39C_00 Type = 001 Mode = 002 Flag = 000
BarCode = CD39C_01 Type = 001 Mode = 002 Flag = 000
BarCode = CD39_00  Type = 001 Mode = 001 Flag = 000
BarCode = CD39_01  Type = 001 Mode = 001 Flag = 000

BarCode = KUNAUNR  Type = 017 Mode = 002 Flag = 000
BarCode = KUNAUPS  Type = 017 Mode = 002 Flag = 000
BarCode = MBBARC   Type = 017 Mode = 002 Flag = 000
BarCode = MBBARC1  Type = 008 Mode = 000 Flag = 000
BarCode = RSNUM    Type = 017 Mode = 002 Flag = 000
BarCode = RSPOS    Type = 017 Mode = 002 Flag = 000
BarCode = RUECKNR  Type = 017 Mode = 002 Flag = 000

BarCode = ZCODE39  Type = 001 Mode = 001 Flag = 000
BarCode = BC_93    Type = 033 Mode = 000 Flag = 000


BarCode = ZBC39    Type = 001 Mode = 001 Flag = 000
BarCode = ZBC39N   Type = 001 Mode = 001 Flag = 000
BarCode = ZINT25NC Type = 012 Mode = 001 Flag = 000
BarCode = ZINT25   Type = 012 Mode = 001 Flag = 000
BarCode = ZBC128N  Type = 017 Mode = 002 Flag = 000
BarCode = ZBC128   Type = 017 Mode = 002 Flag = 000
BarCode = ZBC128U  Type = 017 Mode = 002 Flag = 000
BarCode = ZPDF2    Type = 030 Mode = 000 Flag = 000 FlagPdf = 128
BarCode = ZPDF3    Type = 030 Mode = 000 Flag = 000 FlagPdf = 128
BarCode = ZPDF4    Type = 030 Mode = 000 Flag = 000 FlagPdf = 128
BarCode = ZPDF1    Type = 030 Mode = 000 Flag = 000 FlagPdf = 128
BarCode = ZPDF1    Type = 030 Mode = 000 Flag = 000 FlagPdf = 000
BarCode = ZBC93    Type = 033 Mode = 000 Flag = 000
BarCode = ZBC93N   Type = 033 Mode = 000 Flag = 000
```

Figure 4: Default `barcode.tab` configuration file

Examples:

1. To shorten an existing bar code named BC_CD29, specify this line in the `barcode.tab` file:

```
BarCode=BC_CD39 Type=001 Mode=001 Flag=128 Ratio=200 ModWidth=10
```

2. To map a new user OTF bar code named ZDOBAR to a BCOCA bar code, add this line in the `barcode.tab` file:

```
BarCode=ZDOBAR Type=017 Mode=002 Flag=128
```

For instructions about customizing SAP configuration files, see "Steps for customizing SAP to AFP transform configuration files" on page 91.

defcp.tab configuration file

The `defcp.tab` configuration file maps single-byte ASCII code points to EBCDIC code points. The SAP to AFP transform uses this file for ABAP data conversion.

The format of the file is: ASCII code = EBCDIC code. The transform checks the values for a decimal numeric value between 0 and 255.

Figure 5 on page 80 shows the default `defcp.tab` file.

```
// Linedata CodePage
//-------------------------------------------------------------------------
// This file is used internally by sap2afp, as well as to convert ABAP
// data to linedata.

// Format : Ascii=Ebcdic

// T1000819 + box characters -> T1DABASE

000 = 064
001 = 064
002 = 064
003 = 064
004 = 055
⋮
250 = 222
251 = 219
252 = 208
253 = 064
254 = 062
255 = 223
```

Figure 5: Default `defcp.tab` configuration file

For instructions about customizing SAP configuration files, see "Steps for customizing SAP to AFP transform configuration files" on page 91.

fonts.tab configuration file

The `fonts.tab` configuration file maps the fonts used in the SAP OTF data stream to AFP fonts. If you add an OTF user font, you must add a new entry in the `fonts.tab` configuration file for the new OTF font.

These font families are predefined with SAP R/3:

CNHEI
Simplified Chinese double-byte font (People's Republic of China)

CNKAI
Simplified Chinese double-byte font (People's Republic of China)

CNSONG
Chinese double-bye standard font (People's Republic of China)

COURCYR
COURCYR Cyrillic font, non-proportional (Russia), ISO 8859/5

COURIER
Courier

DBGOTHIC
Heisei Gothic

DBMINCHO
Heisei Mincho

HELVCYR
Cyrillic font, proportional (Russia), ISO 8859/5

HELVE
Helvetica

JPMINCHO
Heisei Mincho

KPSAMMUL
Double-byte standard font (South Korea)

LETGOTH
Letter Gothic

LNPRINT
Line Print

OCRA
Optical Character Recognition A

OCRB
Optical Character Recognition B

TIMECYR
Cyrillic font, proportional (Russia), ISO 8859/5

TIMES
Times New Roman

TWSONG
Traditional Chinese double-byte standard font (Taiwan)

Tip: JPMINCHO and DBMINCHO are different names for the same font.

A font named BARCODE must be defined for the HRI character of a bar code.

These attributes in the `fonts.tab` configuration file set the format of the fonts you use to print with SAP R/3:

DefCodePage
Specifies the default SBCS code page used if no FC OTF command is given or if the requested font is not found in the `fonts.tab` table.

DefCharSet
Specifies the default SBCS character set used if no FC OTF command is given or if the requested font is not found in the `fonts.tab` table.

DBDefCodePage
Specifies the default DBCS code page used if no FC OTF command is given or if the requested font is not found in the `fonts.tab` table.

DBDefCharSet
Specifies the default DBCS character set used if no FC OTF command is given or if the requested font is not found in the `fonts.tab` table.

SBDefCodePage
Specifies the default SBCS code page used for half-width characters in DBCS fonts if no FC OTF command is given or if the requested font is not found in the `fonts.tab` table.

SBDefCharSet
Specifies the default SBCS character set used for half-width characters in DBCS fonts if no FC OTF command is given or if the requested font is not found in the `fonts.tab` table.

Font
Describes the font family (FONTFAMILY parameter of the FC OTF command). The maximum size is 8 bytes and content is not verified.

Size
Specifies the font size in 1/10 of a point (FONT SIZE parameter of the FC OTF command). The value must be numeric and is not verified.

Type
Defines the font type (BOLD and ITALIC parameter of the FC OTF command). `Type=0` is normal, `Type=1` is italic, `Type=2` is bold, and `Type=3` is italic bold. Any other value is not valid.

CodePage
Specifies the code page and requires a valid AFP code page name (8 bytes). The value is not verified. A name that is not valid can result in an error message.

CharSet
Specifies the AFP font character set. The content is not verified.

CodedFont
Specifies the AFP coded font. The content is not verified.

DB
Specifies whether the font is an SBCS font (`DB=0`) or a DBCS font (`DB=1`).

SBCodePage
Specifies the single-byte code page used for half-width characters in DBCS fonts. The value is not verified. A name that is not valid can result in an error message.

SBCharSet
Specifies the AFP single-byte font character set used for half-width characters in DBCS fonts. The content is not verified.

SBCodedFont
Specifies the AFP single-byte coded font used for half-width characters in DBCS fonts. The content is not verified.

If an SBCS font that matches the Font, Size, and Type values is not found, the code page and font character set from the DefCodePage and DefCharSet attributes are used. A warning message is sent.

If a DBCS font that matches the Font, Size, and Type values is not found, the code pages and font character sets from the DBDefCodePage, SBDefCodePage, DBDefCharSet, and SBDefCharSet attributes are used. A warning message is sent.

The SAP device types IBMAFP and IBMAFP3 support the predefined font families. These font families are also supported as IBM Expanded Core Fonts or IBM Japanese Fonts.

For ABAP listings, use the Letter Gothic Latin1 font provided with the AFP Font Collection V2 (program number 5648-B33). If you choose not to install the AFP Font Collection and plan to use the Gothic Text fonts provided with the IBM Compatibility Fonts, adjust the font names in the `pagedef.tab` configuration file.

Figure 6 on page 83 and Figure 7 on page 83 show sections of the `fonts.tab` configuration file.

For z/OS V2R1 and higher, the Letter Gothic Latin1 font is included in the z/OS Font Collection.

```
DefCodePage = T1V10273
DefCharSet  = C0420000
DBDefCodePage = T10300
DBDefCharSet  = CZJHMN
SBDefCodePage = T1H01027
SBDefCharSet  = CZJHMN

// Courier
Font=COURIER  Size=070 Type=0 CodePage=T1V10273 CharSet=C0420070 DB=0
Font=COURIER  Size=070 Type=1 CodePage=T1V10273 CharSet=C0430070 DB=0
Font=COURIER  Size=070 Type=2 CodePage=T1V10273 CharSet=C0440070 DB=0
Font=COURIER  Size=070 Type=3 CodePage=T1V10273 CharSet=C0450070 DB=0
Font=COURIER  Size=080 Type=0 CodePage=T1V10273 CharSet=C0420080 DB=0
Font=COURIER  Size=080 Type=1 CodePage=T1V10273 CharSet=C0430080 DB=0
Font=COURIER  Size=080 Type=2 CodePage=T1V10273 CharSet=C0440080 DB=0
Font=COURIER  Size=080 Type=3 CodePage=T1V10273 CharSet=C0450080 DB=0
Font=COURIER  Size=100 Type=0 CodePage=T1V10273 CharSet=C0420000 DB=0
Font=COURIER  Size=100 Type=1 CodePage=T1V10273 CharSet=C0430000 DB=0
Font=COURIER  Size=100 Type=2 CodePage=T1V10273 CharSet=C0440000 DB=0
Font=COURIER  Size=100 Type=3 CodePage=T1V10273 CharSet=C0450000 DB=0
Font=COURIER  Size=120 Type=0 CodePage=T1V10273 CharSet=C04200B0 DB=0
Font=COURIER  Size=120 Type=1 CodePage=T1V10273 CharSet=C04300B0 DB=0
Font=COURIER  Size=120 Type=2 CodePage=T1V10273 CharSet=C04400B0 DB=0
Font=COURIER  Size=120 Type=3 CodePage=T1V10273 CharSet=C04500B0 DB=0
Font=COURIER  Size=140 Type=0 CodePage=T1V10273 CharSet=C04200D0 DB=0
Font=COURIER  Size=140 Type=1 CodePage=T1V10273 CharSet=C04300D0 DB=0
Font=COURIER  Size=140 Type=2 CodePage=T1V10273 CharSet=C04400D0 DB=0
Font=COURIER  Size=140 Type=3 CodePage=T1V10273 CharSet=C04500D0 DB=0
Font=COURIER  Size=200 Type=0 CodePage=T1V10273 CharSet=C04200J0 DB=0
Font=COURIER  Size=200 Type=1 CodePage=T1V10273 CharSet=C04300J0 DB=0
Font=COURIER  Size=200 Type=2 CodePage=T1V10273 CharSet=C04400J0 DB=0
Font=COURIER  Size=200 Type=3 CodePage=T1V10273 CharSet=C04500J0 DB=0
```

Figure 6: Courier portion of default `fonts.tab` configuration file

```
// Japanese Gothic
Font=DBGOTHIC Size=060 Type=0 CodePage=T10300 CharSet=CZJHKG DB=1 SBCodePage=T1H01027
SBCharSet=CZJHKG
Font=DBGOTHIC Size=080 Type=0 CodePage=T10300 CharSet=CZJHKG DB=1 SBCodePage=T1H01027
SBCharSet=CZJHKG
Font=DBGOTHIC Size=100 Type=0 CodePage=T10300 CharSet=CZJHKG DB=1 SBCodePage=T1H01027
SBCharSet=CZJHKG
Font=DBGOTHIC Size=120 Type=0 CodePage=T10300 CharSet=CZJHKG DB=1 SBCodePage=T1H01027
SBCharSet=CZJHKG
Font=DBGOTHIC Size=140 Type=0 CodePage=T10300 CharSet=CZJHKG DB=1 SBCodePage=T1H01027
SBCharSet=CZJHKG
Font=DBGOTHIC Size=160 Type=0 CodePage=T10300 CharSet=CZJHKG DB=1 SBCodePage=T1H01027
SBCharSet=CZJHKG
Font=DBGOTHIC Size=180 Type=0 CodePage=T10300 CharSet=CZJHKG DB=1 SBCodePage=T1H01027
SBCharSet=CZJHKG
```

Figure 7: Heisei Gothic portion of default `fonts.tab` configuration file

For instructions about customizing SAP configuration files, see "Steps for customizing SAP to AFP transform configuration files" on page 91.

fonts.tab.unicode.truetype configuration file

The `fonts.tab.unicode.truetype` file specifies TrueType fonts to use when transforming OTF files in a Unicode environment. The additional parameters set the names and paths of the fonts you use to print.

You can map the TrueType fonts in `fonts.tab.unicode.truetype` in one of these ways.

- Use FullFontName:

```
Font=COURIER  Size=080 Type=0 FullFontName="WT SerifDuo"
```

A RAT must be present if you use this mapping method.

- Use TTFileName:

```
Font=COURIER  Size=080 Type=0 TTFileName="wt_dw___.ttf"
```

  No RAT is required, and the transform embeds the specified TrueType file in the AFP output.

- Use TTFileName with linked fonts:

```
Font=COURIER  Size=080 Type=0 TTFileName="wt_dw___.ttf"
TTLinkFileNames="wts_w___.ttf:wt__w___.ttf"
```

  No RAT is required. The transform embeds the specified TrueType file and the specified TrueType linked font files in the AFP output.

The `fonts.tab.unicode.truetype` file maps the TrueType fonts using FullFontName. The `fonts.tab.unicode.embed.truetype` file embeds the TrueType fonts using TTFileName; it doesn't use TTLinkFileNames, but you can use TTLinkFileNames when you use TTFileName with linked fonts.

These are the attributes in the `fonts.tab.unicode.truetype` configuration file:

DefTTFileName
Specifies the name of the default TrueType font file used when no matching entry for an OTF font is found.

The TrueType font file is searched in the AOP_WORLDTYPE_PATH directory.

TTFFileName
Specifies the name of the TrueType font file used for the corresponding OTF font entry.

The TrueType font file is searched in the AOP_WORLDTYPE_PATH directory.

TTLinkFileNames
Specifies the linked fonts for the corresponding base font that was specified using TTFileName.

If the base font does not contain a certain character, the transform searches for that character in the fonts specified as linked fonts. You can enter multiple names separated by a colon. The linking order is the order you entered the linked font names.

This parameter is optional and can be entered only after the TTFileName attribute.

DefFullFontName
Specifies the default font full name used when no matching entry for an OTF font is found.

FullFontName
Specifies the default font full name used for the corresponding OTF font entry.

FullFontName indicates that a RAT is present for printing the AFP file, and the TrueType fonts are not embedded in the AFP file. When the name of the TrueType font file is specified, the transform embeds the TrueType fonts in the AFP file.

fonts.tab.unicode.embed.truetype configuration file

The `fonts.tab.unicode.embed.truetype` configuration file is provided as a sample of a `fonts.tab.unicode.truetype` configuration file that tells the transform to embed the TrueType fonts into the file. The transform does not automatically read this file. For Unicode files, when -f or -I is specified, the transform automatically reads `fonts.tab.unicode.truetype`, which uses full font names for TrueType fonts. To use the contents of `fonts.tab.unicode.embed.truetype` to embed fonts, replace the contents of the `fonts.tab.unicode.truetype` file with the contents of `fonts.tab.unicode.embed.truetype`.

You should only use this file to embed TrueType fonts if the product you are using does not use a RAT. When you embed TrueType fonts, the output file is much larger, which might lower performance or cause the transform to run out of space while writing the output.

To use the `fonts.tab.unicode.embed.truetype` file:

1. Create a copy of the original `fonts.tab.unicode.truetype` file.

   ```
   cp fonts.tab.unicode.trueype fonts.tab.unicode.truetype.save
   ```

2. Copy the `fonts.tab.unicode.embed.truetype` configuration file to the `fonts.tab.unicode.truetype` configuration file.

   ```
   cp fonts.tab.unicode.embed.truetype fonts.tab.unicode.truetype
   ```

3. Specify the -f or -I option with the `sap2afp` command. The TrueType fonts are embedded in your AFP output

For instructions about customizing SAP configuration files, see "Steps for customizing SAP to AFP transform configuration files" on page 91.

image.tab configuration file

The `image.tab` configuration file defines values used to print image data. It contains these parameters:

DEFRES
Specifies the default resolution used for printing image data if you do not specify a resolution in the printer definition and the job submitter does not specify a resolution in the -r option on the `lp` or `sap2afp` command. Valid values are 240, 300, 480, 600, 720, and 1200.

Width
Specifies the width of the dither matrix.

Height
Specifies the height of the dither matrix.

Cell
Specifies the values for the dither matrix.

Transform
Specifies 256 grayscale correction values.

These are the `image.tab` files. To use one of these files, rename it to `image.tab`:

- `image.tab` - same as `image.tab.85lr`
- `image.tab.141ap` - 141 lines/inch AppleWriter grayscale emulation
- `image.tab.141dt` - 141 lines/inch Xerox DocuTech grayscale emulation
- `image.tab.141lr` - 141 lines/inch linear gamma correction
- `image.tab.85ap` - 85 lines/inch AppleWriter grayscale emulation
- `image.tab.85dt` - 85 lines/inch Xerox DocuTech grayscale emulation
- `image.tab.85lr` - 85 lines/inch linear gamma correction

Figure 8 on page 86 shows the default `image.tab` configuration file.

```
// image.tab
//
// This file is used to determine the dither matrix and the
// grayscale correction values.
// Define the default output resolution
DEFRES = 600
// Define the dither matrix
Width = 78
Height = 78
Cell =
 15  63 184 219 249 240 158 109  98  54  82 118 133 113  75  64
 87 104 166 215 234 229 179  62  38   4  15  59 186 220 246 240
166 123  87  61  65 108 133 118  78  64 100 105 167 219 234 230
⋮
 28  73 193 223 253 247 199  76  31   8  20  93 148 156 203 178
136 126 144 186 201 155 151  90  43  24  28  75 199 222 254 251
196  79  31   7
//
// Grayscale correction values
//
Transform =
0
1
2
⋮
248
252
255
```

Figure 8: Default `image.tab` configuration file

For instructions about customizing SAP configuration files, see "Steps for customizing SAP to AFP transform configuration files" on page 91.

pagedef.tab configuration file

The `pagedef.tab` configuration file maps SAP R/3 format names to the names of AFP form definitions, AFP page definitions, and fonts. The SAP to AFP transform uses the form definition name when it transforms either SAP OTF or ABAP data. However, the transform uses the page definition and font names only for SAP ABAP data.

Every SAP R/3 document to be printed is associated with an SAP R/3 format; for example, the X_65_80 format for ABAP data or the LETTER format for OTF data. If you define a user format to SAP R/3, you must add the format to the `pagedef.tab` file.

A form definition lets you specify electronic overlays, portrait or landscape presentation mode, the paper source, and simplex or duplex printing. By default, the transform uses the F1A10111 form definition, which specifies no electronic overlays, portrait presentation, the primary paper source, and duplex printing. You can change the form definition in the `pagedef.tab` file to another form definition that PSF provides. For information about form definitions that PSF provides, see *PSF for z/OS: User's Guide*

For example, to print in landscape orientation, you can define new user formats to SAP R/3 (for example ZLAND for ABAP data and Z_65_255 for OTF data), and then map the new user formats to form definition F1C10110 in the `pagedef.tab` file. F1C10110 is a form definition provided by PSF, with landscape presentation mode specified.

File `pagedef.tab` contains these parameters:

Paper
Specifies the name of the SAP R/3 format. This is the value of the OTF print option parameter PJFORM or PJPAPER.

FormDef
Specifies the name of the form definition that is used for printing both OTF and ABAP reports. An error results if PSF cannot find the form definition during printing.

PageDef
Specifies the name of the page definition that is used for printing ABAP reports. An error results if PSF cannot find the page definition during printing.

FontNorm
Specifies the normal coded font that is used for printing ABAP data.

FontBold
Specifies the bold coded font that is used for printing ABAP data.

Figure 9 on page 87 shows the default `pagedef.tab` configuration file.

```
// ABAP Formats
Paper=X_65_132 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=X_44_120 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=X_58_170 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT8A FontBold=GT8A
Paper=X_65_255 FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2A FontBold=GT2A
Paper=X_65_80  FormDef=F1A10111 PageDef=P1C09182 FontNorm=GT2A FontBold=GB2A
Paper=X_90_120 FormDef=F1A10111 PageDef=P1C09182 FontNorm=GT5A FontBold=GT5A
Paper=X_PAPER  FormDef=F1A10111 PageDef=P1C09182 FontNorm=GT2A FontBold=GB2A

// OTF Formats
Paper=DINA3    FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=DINA4    FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=DINA5    FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=EXECUTIV FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=INCH4    FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=INCH6    FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=INCH7    FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=INCH8    FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=INCH11   FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=INCH12   FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=LEGAL    FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=LETTER   FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=LINE_21  FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=LINE_22  FormDef=F1A10111 PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A

//Sample of a added user-defined pagedef entry:
//Paper=Z_INCH12 FormDef=F1A10111    PageDef=P1SAPPD  FontNorm=GT2A FontBold=GB2A
```

Figure 9: Default `pagedef.tab` configuration file

For instructions about customizing SAP configuration files, see "Steps for customizing SAP to AFP transform configuration files" on page 91.

pagedef.tab.unicode configuration file

The `pagedef.tab.unicode` configuration file maps SAP R/3 format names to the names of AFP form definitions, AFP page definitions, and fonts in a Unicode environment. The SAP to AFP transform uses the form definition name when it transforms either SAP OTF or ABAP data. The structure is the same as the `pagedef.tab` file.

For ABAP format entries, the page definitions that map the Courier Unicode fonts are used by default. To process ABAP files containing CJK characters, specify the names of the corresponding Unicode page definitions for Japanese, Simplified Chinese, Traditional Chinese, or Korean in the `pagedef.tab.unicode` configuration file.

If you have your own custom page definitions to use for Unicode processing, modify your sources to include mappings and associations for the required Unicode fonts and recompile them. Specify the name of the custom page definition in the `pagedef.tab.unicode` configuration file.

For instructions about customizing SAP configuration files, see "Steps for customizing SAP to AFP transform configuration files" on page 91.

pagedef.tab.unicode.truetype configuration file

The `pagedef.tab.unicode.truetype` configuration file specifies the form definitions to use with TrueType for ABAP and OTF input files and page definitions to use with TrueType for ABAP files in a Unicode environment. The structure is identical to the `pagedef.tab` file.

Figure 10 on page 88 shows a sample `pagedef.tab.unicode.truetype` configuration file.

```
// ABAP Formats; FontNorm and FontBold are ignored in this table,
// because they are set in the corresponding unicode pagedef.
Paper=X_65_132 FormDef=F1SAP PageDef=P1UTT683 FontNorm=xxxx FontBold=yyyy
Paper=X_44_120 FormDef=F1SAP PageDef=P1UTT683 FontNorm=xxxx FontBold=yyyy
Paper=X_58_170 FormDef=F1SAP PageDef=P1UTT683 FontNorm=xxxx FontBold=yyyy
Paper=X_65_255 FormDef=F1SAP PageDef=P1UTT683 FontNorm=xxxx FontBold=yyyy
Paper=X_65_80  FormDef=F1SAP PageDef=P1UTT182 FontNorm=xxxx FontBold=yyyy
Paper=X_90_120 FormDef=F1SAP PageDef=P1UTT182 FontNorm=xxxx FontBold=yyyy
Paper=X_PAPER  FormDef=F1SAP PageDef=P1UTT182 FontNorm=xxxx FontBold=yyyy

// OTF Formats
Paper=DINA3    FormDef=F1SAP PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=DINA4    FormDef=F1SAP PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=DINA5    FormDef=F1SAP PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=EXECUTIV FormDef=F1SAP PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=INCH4    FormDef=F1SAP PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=INCH6    FormDef=F1SAP PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=INCH7    FormDef=F1SAP PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=INCH8    FormDef=F1SAP PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=INCH11   FormDef=F1SAP PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=INCH12   FormDef=F1SAP PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=LEGAL    FormDef=F1SAP PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=LETTER   FormDef=F1SAP PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=LINE_21  FormDef=F1SAP PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A
Paper=LINE_22  FormDef=F1SAP PageDef=P1V06683 FontNorm=GT2A FontBold=GB2A

//Sample of a added user-defined pagedef entry:
//Paper=Z_INCH12 FormDef=F1SAP    PageDef=P1SAPPD  FontNorm=GT2A FontBold=GB2A
```

Figure 10: Sample `pagedef.tab.unicode.truetype` configuration file

For instructions about customizing SAP configuration files, see "Steps for customizing SAP to AFP transform configuration files" on page 91.

*xxxx*0000.tab configuration file

These configuration files contain tables that map characters of an individual SAP code page to an AFP code page:

Table 7: SAP code page configuration files

File name	SAP R/3 input code page	Description of input code page	AFP output code page	Description of output code page
00000000.tab	T1000819	Latin-1 ISO ANSI 8-bit (ASCII)	T1V10273	Germany F.R./ Austria-CECP
11000000.tab	T1000819	Latin-1 ISO ANSI 8-bit (ASCII)	T1V10273	Germany F.R./ Austria-CECP
15000000.tab	T1000915	Cyrillic ISO 8859/5	T1001172	Cyrillic Multilingual with Box Draw
40010000.tab	T1000876[1]	OCR-A (ASCII)	T1000892	OCR-A
40040000.tab	T1000877[1]	OCR-B (ASCII), MICR_C, MICR_E	T1000893	OCR-B
41020000.tab	4102	Unicode UTF-16 Big Endian	T11200	Unicode
41030000.tab	4103	Unicode UTF-16 Little Endian	T11200	Unicode

Table 7: SAP code page configuration files (continued)

File name	SAP R/3 input code page	Description of input code page	AFP output code page	Description of output code page
80000000.tab	IBM-943C	Japanese ISO Shift-JIS	IBM-1399-SAP2AFP	Japanese
83000000.tab	8300	Traditional Chinese ISO Big5	IBM-937-SAP2AFP	Traditional Chinese
84000000.tab	8400	Simplified Chinese ISO GB2312	IBM-1388-SAP2AFP	Simplified Chinese
85000000.tab	8500	Korean ISO KSC 5601	IBM-933-SAP2AFP	Korean
86000000.tab	8600	Thai ISO TIS620–2529	T1000874	Thai

Note:

1. The input OCR-A and the OCR-B code pages are equivalent to ISO-8859-1 with the special characters *hook*, *fork*, and *chair*.

Asian versions of SAP R/3 use special coding to represent double-byte special characters. Therefore, the `80000000.tab` file must indicate the SAP R/3 system code page and the output code page. These parameters in the table specify the input and output code pages:

- CONVERT FOR=*xxxx*

  xxxx is the SAP R/3 system code page.
- CONVERT TO=*yyyyyyyyyy*

  yyyyyyyy is the output code page conversion table.

Figure 11 on page 89 shows the default `80000000.tab` configuration file used for the SAP R/3 system code page 8000 (Japanese).

```
CONVERT FOR=8000
CONVERT TO=ibm-1399-sap2afp
```

Figure 11: Default `80000000.tab` configuration file

For instructions about customizing SAP configuration files, see “Steps for customizing SAP to AFP transform configuration files” on page 91.

6400fonts.tab configuration file

The `6400fonts.tab` configuration file maps Output Text Format (OTF) fonts to fonts in an InfoPrint 6400 printer. The printer must support the graphics object content architecture (GOCA) feature.

You can edit the `6400fonts.tab` file to resize an OTF font. To do this, you specify the SIZE and CPI parameters.

Parameters in the configuration file are:

CharSet
Specifies the AFP font character set.

CodedFont
Specifies the AFP coded font. The content is not verified.

CodePage
Specifies the AFP code page name (8 bytes).

CPI
Specifies the horizontal width (in characters per inch) to which the 6400 printer-resident fonts should be scaled. The smaller the number you specify, the wider the font you receive.

DB
Specifies whether the font is an SBCS font (DB=0) or a DBCS font (DB=1).

Font
Describes the font family (FONTFAMILY parameter of the FC OTF command). The maximum size is 8 bytes.

SBCharSet
Specifies the AFP single-byte font character set used for half-width characters in DBCS fonts.

SBCodedFont
Specifies the AFP single-byte coded font used for half-width characters in DBCS fonts.

SBCodePage
Specifies the single-byte code page used for half-width characters in DBCS fonts.

Size
Specifies the height (in points) to which the 6400 printer-resident font should be scaled. Because there are 72 points in an inch, 720 indicates a height of an inch.

Type
Defines the font type. Valid values are: Type=0 is normal, Type=1 is italic, Type=2 is bold, and Type=3 is italic bold.

Examples:

1. For fonts that print one inch high, specify this line in the 6400fonts.tab configuration file:

```
Font=TIMES Size=720 Type=3 CocePage=T1V10273 CharSet=C0S0ESTR DB=0
```

2. For fonts that print one inch wide, specify this line in the 6400fonts.tab configuration file:

```
Font=TIMES Size=360 Type=3 CodePage=T1V10273 CharSet=C0S0ESTR DB=0 CPI=1
```

3. For fonts that print one-half inch wide, specify this line in the 6400fonts.tabconfiguration file:

```
Font=TIMES Size=360 Type=3 CodePage=T1V10273 CharSet=C0S0ESTR DB=0 CPI=2
```

Note: After you edit the 6400fonts.tab file, access the Print Screen List window on the SAP R/3 graphical user interface and specify 6400 in the **Title** field of the Spool request pane.

For instructions about customizing SAP configuration files, see "Steps for customizing SAP to AFP transform configuration files" on page 91.

userenv configuration file

The userenv configuration file, provided in /usr/lpp/Printsrv/sap2afpv2.4/res, specifies the processing environment, such as the character-set encoding used for data in ASCII SAP files. To enable the SAP to AFP transform for non-Latin character sets, you must edit the userenv configuration file to specify the character set. Valid values in userenv are:

Value	Character Set
CYRILLIC	Cyrillic
JAPANESE	Japanese (Shift-JIS encoding)
KOREAN	Korean
NORMAL	Latin
SPCHINESE	Simplified Chinese
TDCHINESE	Traditional Chinese

Value	Character Set
UNICODE	Unicode, or a mixture of ASCII and Unicode

The SAP to AFP transform can automatically detect Unicode encoding of SAP OTF and ABAP files. Therefore, to transform SAPGOFU files, the `userenv` file does not need to specify UNICODE. However, to transform SAP ASCII files that contain Unicode data, the `userenv` file must specify UNICODE. If the `userenv` file specifies UNICODE, all SAP files must contain Unicode data or ASCII with Unicode.

For Unicode, if the -f or -I option is specified, `pagedef.tab.unicode.truetype` is selected as the page definition table file.

For instructions about customizing SAP configuration files, see "Steps for customizing SAP to AFP transform configuration files" on page 91.

Steps for customizing SAP to AFP transform configuration files

1. Create a new directory.

   Do not change files that reside in the `/usr/lpp/Printsrv/sap2afpv2.4/res` directory; you should create a new resource directory and point to it by using the AOP_SAP2AFP_RESOURCES environment variable in the `aopxfd.conf` configuration file. For example, you can create directory `/etc/Printsrv/sap2afpv2.4/res`.
2. Copy all configuration files to the new directory or use symbolic links.

   All configuration files must reside in the same directory. However, if you do not want to customize some files, you can use symbolic links to the `/usr/lpp/Printsrv/sap2afpv2.4/res` directory for those files.
3. Change to the new directory.
4. Convert the files that you want to customize from ASCII to EBCDIC.
5. Edit the files and save them.
6. Convert the edited files from EBCDIC to ASCII.
7. Specify the directory that contains all the configuration files in the AOP_SAP2AFP_RESOURCES environment variable in the Infoprint Server transform configuration file, `aopxfd.conf`.

   Tip: You can also specify the AOP_SAP2AFP_RESOURCES environment variable in the AOPSTART EXEC. The `aopxfd.conf` file takes precedence over the AOPSTART EXEC.
8. Restart the Infoprint Server Transform Manager.

   Tip: After you restart the Infoprint Server Transform Manager, check for error messages in the transform's `stderr` file. If you find any error messages, fix the errors and restart the Infoprint Server Transform Manager. For information about how to find the transform message logs, see "Finding the transform stderr file" on page 105.

Customizing multibyte conversion tables

To transform and print SAP R/3 data streams that contain multibyte special characters, you might need to customize the conversion tables for the code pages that the SAP to AFP transform uses.

You can edit the source files that the transform provides in the `/usr/lpp/Printsrv/sap2afpv2.4/icu` directory, and then use the z/OS UNIX `makeconv` command to compile them. These source files define the mapping between UCS-2 and multibyte code sets:

- Japanese: `ibm-939-sap2afp.ucm`
- Japanese: `ibm-1399-sap2afp.ucm`
- Korean: `ibm-933-sap2afp.ucm`
- Simplified Chinese: `ibm-1388-sap2afp.ucm`
- Traditional Chinese: `ibm-937-sap2afp.ucm`

Steps for customizing conversion tables:

1. Create a new directory for the conversion tables. For example, you can create directory `/etc/Printsrv/sap2afpv2.4/icu`.

   You should *not* change files that reside in the `/usr/lpp/Printsrv/sap2afpv2.4/icu` directory.
2. Copy all files from directory `/usr/lpp/Printsrv/sap2afpv2.4/icu` to the new directory or use symbolic links.

   All `.cnv` conversion tables must in the same directory. However, if you do not want to customize some files, you can use symbolic links to the `/usr/lpp/Printsrv/sap2afpv2.4/icu` directory for those files.
3. Change to the new directory. For example:

   ```
   cd /etc/Printsrv/sap2afpv2.4/icu
   ```
4. Edit the `.ucm` file and save it.
5. Use the z/OS UNIX `makeconv` command to compile the `.ucm` file and create a `.cnv` file. For example:

   ```
   makeconv -p ICUDATA ibm-1388-sap2afp.ucm
   ```
6. Change the access permissions of the .cnv file to give everyone permission to read the file. For example:

   ```
   chmod 644 ibm-1388-sap2afp.cnv
   ```
7. Specify the new directory that contains the conversion tables in the AOP_SAP2AFP_ICU environment variable in the SAP to AFP transform entry of the Infoprint Server transform configuration file.

   Tip: You can also specify the AOP_SAP2AFP_RESOURCES environment variable in the AOPSTART EXEC. The `aopxfd.conf` file takes precedence over the AOPSTART EXEC.
8. Restart the Infoprint Server Transform Manager.

   Tip: After you restart the Infoprint Server Transform Manager, check for error messages in the transform's `stderr` file. If you find any error messages, fix the errors and restart the Infoprint Server Transform Manager. For more information about how to find the transform message logs, see "Finding the transform stderr file" on page 105.

Chapter 4. Administering transforms

This information contains tasks that describe how Infoprint Server administrators can set up printer definitions in the Infoprint Server Printer Inventory so that Infoprint Server automatically transforms documents to AFP format before it writes them to the JES spool or sends them to other print servers or email destinations.

Requesting the PCL to AFP transform

This information describes how you can set up your printer definitions so that Infoprint Server automatically calls the PCL to AFP transform when it processes PCL documents.

PCL to AFP transform filter

In the printer definitions for AFP printers, you can specify the PCL to AFP transform filter and associate it with the PCL data format. A filter is a program that modifies the input data before it is sent to the printer. When you associate the transform filter with the PCL data format, Infoprint Server automatically calls the PCL to AFP transform to process a document with the PCL data format.

Table 8 on page 93 shows the filter name and filter options that you can specify in the **Filter** field of a printer definition.

Table 8: PCL to AFP filter name and filter options

Field name (attribute name)	**Filter name**	**Filter options**
Filter (filters)	pcl2afp.dll	[%filter-options] [-a *imagetype*] [-c *transformclass*] [-p *pagerange*] [-r *resolution*] [-t *outputtype*] [-w *width*] [-x *xmargin*] [-y *ymargin*] [-C] [-I] [-T]

The filter options are:

%filter-options
Passes options that are specified in the filter-options job attribute to the transform.

You can type the %filter-options option in any position relative to the other filter options. If you specify filter options to the right of %filter-options, those options override the same options that are specified in the filter-options job attribute.

-c *transformclass*
Specifies the name of a transform class that is defined in the transform configuration file, `aopxfd.conf`. The name is case-sensitive. The job submitter can also specify this option in the filter-options job attribute.

For a description of the other PCL to AFP filter options, see "pcl2afp—Transform PCL data to AFP data" on page 11.

Steps for editing printer definitions for the PCL to AFP transform

To edit printer definitions, use either Infoprint Server ISPF panels or the Printer Inventory Definition Utility (PIDU) program. This information describes how to use the ISPF panels. For information about the PIDU program, see *z/OS Infoprint Server Operation and Administration*.

Before you begin: You must be authorized to edit the Printer Inventory. You must have UPDATE access to the AOP.ADMINISTRATOR profile in the RACF PRINTSRV class.

Steps for editing printer definitions:

1. (Optional) Create a Processing component. If you need to specify the PCL to AFP transform in a large number of printer definitions, you can create a component to simplify administration. Then, to change your transform filter in the future, you can make the change in your Processing component.
2. On the Processing panel of either the printer definition or the component, select the PCL data format.

   Tip: In PSF printer definitions, also select the `Line data`, `MO:DCA-P`, and `Text` data formats. PSF can print line data and MO:DCA-P data on AFP printers, and Infoprint Server automatically converts text data to line data.
3. Next to the **PCL** data format, specify the `pcl2afp.dll` filter and filter options in the **Filter** field. Type the absolute path name if the filter is not in a directory that is named in the LIBPATH environment variable. For filter options, see "PCL to AFP transform filter" on page 93.
4. In PSF printer definitions, do not select the **Resubmit for filtering** field.
5. (Optional) In the **Fail on error** field, select whether you want the transform to fail when a transform warning or data stream error occurs during the transform. When a transform fails, the transform does not produce an output document.

   The Infoprint Server fail-on-transform-error job attribute overrides this field.

   If you do not select an option in this field and fail-on-transform-error is not specified, the AOP_FAIL_ON_ERROR environment variable determines when the transform fails.
6. (Optional) In the **Trailer error page** field, select the type of messages you want the transform to write on a page at the end of the AFP output (called the *trailer error page*).

   The Infoprint Server trailer-transform-error-page job attribute overrides this value.

   If you do not select an option in this field, the AOP_TRAILER_ERROR_PAGE environment variable determines the type of messages the transform writes.
7. (Optional) If you created a Processing component, specify the name of the Processing component in the **Component name** field on the PSF printer definition panel. Do this step in all printer definitions to which the transform applies.

   After you include a Processing component, check the Processing panel in the printer definition to ensure that the transform is specified correctly. You might need to remove (space over) any filters that are specified on the Processing panel of the printer definition itself. If any filters are specified in the printer definition, the filters that are specified in the component are not used.
8. (Optional) On the Allocation panel, you can specify the name of the form definition that PSF uses when it prints the AFP image data that the transform creates.

Example -- ISPF Processing panel for the PCL to AFP transform

This ISPF panel shows how to specify the PCL to AFP transform in a printer definition for a PSF printer. Only a portion of the Processing panel is shown.

```
                                Processing

Printer definition name . afp-printer
⋮
Supported data formats and associated filters:
Data format:  Filter:

/ Line data   ____________________________________________________ (extend)
/ MO:DCA-P    ____________________________________________________ (extend)
_ PostScript  ____________________________________________________ (extend)
/ Text        ____________________________________________________ (extend)
/ PCL         pcl2afp.dll -c letter_300 %filter-options            (extend)
_ PDF         ____________________________________________________ (extend)
_ SAP         ____________________________________________________ (extend)
_ XML         ____________________________________________________ (extend)
_ Other       ____________________________________________________ (extend)

_ Resubmit for filtering

Transforms to AFP:
   Fail on error . . . .  2   1. No  2. Error  3. Warning
   Trailer error page. .  1   1. No  2. Error  3. Warning
⋮
```

Explanation of fields:

- The PCL data format is selected because the `pcl2afp2.dll` filter can transform PCL documents to AFP format. The transform options are:
  - The -c filter option causes the transform to use the letter_300 transform class, which is defined in the transform configuration file.

    Because the -c option is specified to the left of %filter-options, any -c option specified in the `filter-options` job attribute overrides the -c option that is specified in the printer definition.
  - The %filter-options filter option causes the transform to use transform options that are specified in the filter-options job attribute.
- The Line data and MO:DCA-P data formats are selected because PSF can print these data formats on AFP printers.
- The Text data format is selected because Infoprint Server automatically converts text data to line data when it is processing data for PSF printers.
- In the **Fail on error** field, option Error is selected. If a data stream error occurs during the transform, the transform fails and does not create an output document.
- In the **Trailer error page** field, option No is selected. The transform does not write any warning or error messages to a trailer error page.

Requesting the PDF to AFP transform

This information describes how you can set up your printer definitions so that Infoprint Server automatically calls the PDF to AFP transform when it processes PDF documents.

PDF to AFP transform filter

In the printer definitions for AFP printers, you can specify the PDF to AFP transform filter and associate it with the PDF data format. (A filter is a program that modifies the input data before it is sent to the printer.) When you associate the transform filter with the PDF data format, Infoprint Server automatically calls the PDF to AFP transform to process a document with the PDF data format.

Table 9 on page 96 shows the filter name and filter options that you can specify in the **Filter** field of a printer definition.

Tip: The same `ps2afp.dll` filter transforms both PostScript and PDF documents to AFP format, but the filter options are different for each data format.

Table 9: PDF to AFP filter name and filter options

Field name (attribute name)	Filter name	Filter options
Filter (filters)	ps2afp.dll	[%filter-options] [-a *imagetype*] [-c *transformclass*] [-l *length*] [-p *pagerange*] [-r *resolution*] [-t *outputtype*] [-w *width*] [-x *xmargin*] [-y *ymargin*] [-I] [-T]

The filter options mean:

%filter-options
Causes options that are specified in the filter-options job attribute to be passed to the transform.

You can type the %filter-options option in any position relative to the other filter options. If you specify filter options to the right of %filter-options, those options override the same options that are specified in the filter-options job attribute.

-c *transformclass*
Specifies the name of a transform class that is defined in the transform configuration file, `aopxfd.conf`. The name is case-sensitive. The job submitter can also specify this option in the filter-options job attribute.

For a description of the other PDF to AFP filter options, see "pdf2afp—Transform PDF data to AFP data" on page 21.

Steps for editing printer definitions for the PDF to AFP transform

To edit printer definitions, use either Infoprint Server ISPF panels or the Printer Inventory Definition Utility (PIDU) program. This information describes how to use the ISPF panels. For information about the PIDU program, see *z/OS Infoprint Server Operation and Administration*.

Before you begin: You must be authorized to edit the Printer Inventory. You must have UPDATE access to the AOP.ADMINISTRATOR profile in the RACF PRINTSRV class.

Steps for editing printer definitions:

1. (Optional) Create a Processing component. If you need to specify the PDF to AFP transform in a large number of printer definitions, a component can simplify administration. If you need to change your transform filter in the future, you can make the change in your Processing component.
2. On the Processing panel of either the printer definition or the component, select the **PDF** data format.

   Tip: In PSF printer definitions, also select the **Line data**, **MO:DCA-P**, and **Text** data formats. PSF can print line data and MO:DCA-P data on AFP printers, and Infoprint Server automatically converts text data to line data.
3. Next to the **PDF** data format, specify the `ps2afp.dll` filter and filter options in the **Filter** field. Type the absolute path name if the filter is not in a directory that is named in the LIBPATH environment variable. For filter options, see "PDF to AFP transform filter" on page 95.
4. In PSF printer definitions, do not select the **Resubmit for filtering** field.
5. (Optional) In the **Fail on error** field, select whether you want the transform to fail when a transform warning or data stream error occurs during the transform. When a transform fails, the transform does not produce an output document.

The Infoprint Server fail-on-transform-error job attribute overrides this field.

If you do not select an option in this field and fail-on-transform-error is not specified, the AOP_FAIL_ON_ERROR environment variable determines when the transform fails.

6. (Optional) In the **Trailer error page** field, select the type of messages you want the transform to write on a page at the end of the AFP output (called the *trailer error page*).

   The Infoprint Server trailer-transform-error-page job attribute overrides this value.

   If you do not select an option in this field, the AOP_TRAILER_ERROR_PAGE environment variable determines the type of messages the transform writes.

7. (Optional) If you created a Processing component, specify the name of the Processing component in the **Component name** field on the PSF printer definition panel. Do this step in all printer definitions to which the transform applies.

   After you include a Processing component, check the Processing panel in the printer definition to ensure that the transform is specified correctly. You might need to remove (space over) any filters that are specified on the Processing panel of the printer definition itself. If any filters are specified in the printer definition, the filters that are specified in the component are not used.

8. (Optional) On the Allocation panel, you can specify the name of the form definition that PSF uses when it prints the AFP image data that the transform creates.

Example -- ISPF Processing panel for the PDF to AFP transform

This ISPF panel shows how to specify the PDF to AFP transform in a printer definition for a PSF printer. Only a portion of the Processing panel is shown.

```
                                Processing

Printer definition name . afp-printer_____
⋮
Supported data formats and associated filters:
Data format:  Filter:

/ Line data   ____________________________________________________ (extend)
/ MO:DCA-P    ____________________________________________________ (extend)
_ PostScript  ____________________________________________________ (extend)
/ Text        ____________________________________________________ (extend)
_ PCL         ____________________________________________________ (extend)
/ PDF         ps2afp.dll  %filter-options -r 300_________________ (extend)
_ SAP         ____________________________________________________ (extend)
_ XML         ____________________________________________________ (extend)
_ Other       ____________________________________________________ (extend)

_ Resubmit for filtering

Transforms to AFP:
   Fail on error . . . .  2   1. No  2. Error  3. Warning
   Trailer error page. .  1   1. No  2. Error  3. Warning
⋮
```

Explanation of fields:

- The **PDF** data format is selected because the `ps2afp2.dll` filter can transform PDF documents to AFP format. The transform options are:
  - The %filter-options filter option causes the transform to use transform options that are specified in the filter-options job attribute.
  - The -r option causes the transform to format the output for a 300-pel resolution printer.

    Because the -r option is specified to the right of %filter-options, the transforms ignore the -r option specified in the filter-options job attribute.

- The **Line data** and **MO:DCA-P** data formats are selected because PSF can print these data formats on AFP printers.
- The **Text** data format is selected because Infoprint Server automatically converts text data to line data when it processes data for PSF printers.
- In the **Fail on error** field, option `Error` is selected. If a data stream error occurs during the transform, the transform fails and does not create an output document.
- In the **Trailer error page** field, option `No` is selected. The transform does not write any warning or error messages to a trailer error page.

Requesting the PostScript to AFP transform

This information describes how you can set up your printer definitions so that Infoprint Server automatically calls the PostScript to AFP transform when it processes PostScript documents.

PostScript to AFP transform filter

In the printer definitions for AFP printers, you can specify the PostScript to AFP transform filter and associate it with the PostScript data format. (A filter is a program that modifies the input data before it is sent to the printer.) When you associate the transform filter with the PostScript data format, Infoprint Server automatically calls the PostScript to AFP transform when it processes a document with the PostScript data format.

Table 10 on page 98 shows the filter name and filter options that you can specify in the **Filter** field of a printer definition.

Tip: The same `ps2afp.dll` filter transforms both PostScript and PDF documents to AFP format but the filter options are different for each data format.

Table 10: PostScript to AFP filter name and filter options

Field name (attribute name)	Filter name	Filter options
Filter (filters)	ps2afp.dll	[%filter-options] [-a *imagetype*] [-c *transformclass*] [-l *length*] [-o *outputfile*] [-r *resolution*] [-t *outputtype*] [-w *width*] [-x *xmargin*] [-y *ymargin*] [-I] [-T] [*inputfile...*]

The filter options mean:

%filter-options
Causes options that are specified in the filter-options job attribute to be passed to the transform.

You can type the %filter-options option in any position relative to the other filter options. If you specify filter options to the right of %filter-options, those options override the same options that are specified in the filter-options job attribute.

-c *transformclass*
Specifies the name of a transform class that is defined in the transform configuration file, `aopxfd.conf`. The name is case-sensitive. The job submitter can also specify this option in the filter-options job attribute.

For a description of the other PostScript to AFP filter options, see "ps2afp—Transform PostScript data to AFP data" on page 30.

Steps for editing printer definitions for the PostScript to AFP transform

To edit printer definitions, use either Infoprint Server ISPF panels or the Printer Inventory Definition Utility (PIDU) program. This information describes how to use the ISPF panels. For information about the PIDU program, see *z/OS Infoprint Server Operation and Administration*.

Before you begin: You must be authorized to edit the Printer Inventory. You must have UPDATE access to the AOP.ADMINISTRATOR profile in the RACF PRINTSRV class.

Steps for editing printer definitions:

1. (Optional) Create a Processing component. If you need to specify the PostScript to AFP transform in a large number of printer definitions, a component can simplify administration. If you need to change your transform filter in the future, you can make the change in your Processing component.
2. On the Processing panel of either the printer definition or the component, select the **PostScript** data format.

   Tip: In PSF printer definitions, also select the **Line data**, **MO:DCA-P**, and **Text** data formats. PSF can print line data and MO:DCA-P data on AFP printers, and Infoprint Server automatically converts text data to line data.
3. Next to the **PostScript** data formats, specify the `ps2afp.dll` filter and filter options in the **Filter** field. Type the absolute path name if the filter is not in a directory that is named in the LIBPATH environment variable. For filter options, see "PostScript to AFP transform filter" on page 98.
4. In PSF printer definitions, do not select the **Resubmit for filtering** field.
5. (Optional) In the **Fail on error** field, select whether you want the transform to fail when a transform warning or data stream error occurs during the transform. When a transform fails, the transform does not produce an output document.

   The Infoprint Server fail-on-transform-error job attribute overrides this field.

   If you do not select an option in this field and fail-on-transform-error is not specified, the AOP_FAIL_ON_ERROR environment variable determines when the transform fails.
6. (Optional) In the **Trailer error page** field, select the type of messages you want the transform to write on a page at the end of the AFP output (called the *trailer error page*).

   The Infoprint Server trailer-transform-error-page job attribute overrides this value.

   If you do not select an option in this field, the AOP_TRAILER_ERROR_PAGE environment variable determines the type of messages the transform writes.
7. (Optional) If you created a Processing component, specify the name of the Processing component in the **Component name** field on the PSF printer definition panel. Do this step in all printer definitions to which the transform applies.

   After you include a Processing component, check the Processing panel in the printer definition to ensure that the transform is specified correctly. You might need to remove (space over) any filters that are specified on the Processing panel of the printer definition itself. If any filters are specified in the printer definition, the filters that are specified in the component are not used.
8. (Optional) On the Allocation panel, you can specify the name of the form definition that PSF uses when it prints the AFP image data that the transform creates.

Example -- ISPF Processing panel for the PostScript to AFP transform

This ISPF panel shows how to specify the PostScript to AFP transform in a printer definition for a PSF printer. Only a portion of the Processing panel is shown.

```
                                Processing

Printer definition name . afp-printer_____
⋮
Supported data formats and associated filters:
Data format:  Filter:

/ Line data   ____________________________________________________ (extend)
/ MO:DCA-P    ____________________________________________________ (extend)
/ PostScript  ps2afp.dll  %filter-options -r 300_________________ (extend)
/ Text        ____________________________________________________ (extend)
_ PCL         ____________________________________________________ (extend)
_ PDF         ____________________________________________________ (extend)
_ SAP         ____________________________________________________ (extend)
_ XML         ____________________________________________________ (extend)
_ Other       ____________________________________________________ (extend)

_ Resubmit for filtering

Transforms to AFP:
   Fail on error . . . .  2   1. No  2. Error  3. Warning
   Trailer error page. .  1   1. No  2. Error  3. Warning
⋮
```

Explanation of fields:

- The **PostScript** data format is selected because the `ps2afp2.dll` filter can transform PostScript documents to AFP format. The transform options are:
  - The %filter-options filter option causes the transform to use transform options that are specified in the filter-options job attribute.
  - The -r option causes the transform to format the output for a 300-pel resolution printer.

    Because the -r option is specified to the right of %filter-options, the transforms ignore the -r option that is specified in the filter-options job attribute.
- The **Line data** and **MO:DCA-P** data formats are selected because PSF can print these data formats on AFP printers.
- The **Text** data format is selected because Infoprint Server automatically converts text data to line data when it is processing data for PSF printers.
- In the **Fail on error** field, option **Error** is selected. If a data stream error occurs during the transform, the transform fails and does not create an output document.
- In the **Trailer error page** field, option **No** is selected. The transform does not write any warning or error messages to a trailer error page.

Requesting the SAP to AFP transform

This information describes how you can set up your printer definitions so that Infoprint Server automatically calls the SAP to AFP transform when it processes SAP documents.

SAP to AFP transform filter

In the printer definitions for AFP printers, you can specify the SAP to AFP transform filter and associate it with the PCL data format. (A filter is a program that modifies the input data before it is sent to the printer.) When you associate the transform filter with the SAP data format, Infoprint Server automatically calls the SAP to AFP transform when it processes a document with the SAP data format.

Table 11 on page 101 shows the filter name and filter options that you can specify in the **Filter** field of a printer definition.

Table 11: SAP to AFP filter name and filter options		
Field name (attribute name)	**Filter name**	**Filter options**
Filter (filters)	sap2afp.dll	[%filter-options] [-a *imagetype*] [-b] [-c *transformclass*] [-r *resolution*] [-f][-h] [-k] [-s] [-u] [-I] [-T]

The filter options mean:

%filter-options
Causes options that are specified in the filter-options job attribute to be passed to the transform.

You can type the %filter-options option in any position relative to the other filter options. If you specify filter options to the right of %filter-options, those options override the same options that are specified in the filter-options job attribute.

-c *transformclass*
Specifies the name of a transform class that is defined in the transform configuration file, `aopxfd.conf`. The name is case-sensitive. The job submitter can also specify this option in the filter-options job attribute.

For a description of the other SAP to AFP filter options, see "sap2afp—Transform SAP OTF or ABAP data to AFP data" on page 39.

Steps for editing printer definitions for the SAP to AFP transform

To edit printer definitions, use either Infoprint Server ISPF panels or the Printer Inventory Definition Utility (PIDU) program. This information describes how to use the ISPF panels. For information about the PIDU program, see *z/OS Infoprint Server Operation and Administration*.

Before you begin:

You must be authorized to edit the Printer Inventory. You must have UPDATE access to the AOP.ADMINISTRATOR profile in the RACF PRINTSRV class.

Steps for editing printer definitions:

1. (Optional) Create a Processing component. If you need to specify the SAP to AFP transform in a large number of printer definitions, a component can simplify administration. If you need to change your transform filter in the future, you can make the change in your Processing component.
2. On the Processing panel of either the printer definition or the component, select the **SAP** data format.

   Tip: In PSF printer definitions, also select the **Line data**, **MO:DCA-P**, and **Text** data formats. PSF can print line data and MO:DCA-P data on AFP printers, and Infoprint Server automatically converts text data to line data.
3. Next to the **SAP** data format, specify the **sap2afp.dll** filter and filter options in the **Filter** field. Type the absolute path name if the filter is not in a directory that is named in the LIBPATH environment variable. For filter options, see "SAP to AFP transform filter" on page 100.
4. In PSF printer definitions, do not select the **Resubmit for filtering** field.
5. (Optional) In the **Fail on error** field, select whether you want the transform to fail when a transform warning or data stream error occurs during the transform. When a transform fails, the transform does not produce an output document.

   The Infoprint Server **fail-on-transform-error** job attribute overrides this field.

   If you do not select an option in this field and **fail-on-transform-error** is not specified, the AOP_FAIL_ON_ERROR environment variable determines when the transform fails.

6. (Optional) In the **Trailer error page** field, select the type of messages you want the transform to write on a page at the end of the AFP output (called the **trailer error page**).

   The Infoprint Server trailer-transform-error-page job attribute overrides this value.

   If you do not select an option in this field, the AOP_TRAILER_ERROR_PAGE environment variable determines the type of messages the transform writes.
7. (Optional) If you created a Processing component, specify the name of the Processing component in the **Component name** field on the PSF printer definition panel. Do this step in all printer definitions to which the transform applies.

   After you include a Processing component, check the Processing panel in the printer definition to make sure that the transform is specified correctly. You might need to remove (space over) any filters that are specified on the Processing panel of the printer definition itself. If any filters are specified in the printer definition, the filters that are specified in the component are not used.

Tip: The SAP to AFP transform determines the appropriate form definition, page definition, and fonts to use from its own configuration files and overrides any form definition, page definition, and fonts that are specified on the Allocation panel or in job attributes.

Example -- ISPF Processing panel for the SAP to AFP transform

This ISPF panel shows how to specify the SAP to AFP transform in printer definition for a PSF printer. Only a portion of the Processing panel is shown.

```
                              Processing

Printer definition name . afp-printer_____
:
Supported data formats and associated filters:
Data format:  Filter:

/ Line data   ____________________________________________________ (extend)
/ MO:DCA-P    ____________________________________________________ (extend)
_ PostScript  ____________________________________________________ (extend)
/ Text        ____________________________________________________ (extend)
_ PCL         ____________________________________________________ (extend)
_ PDF         ____________________________________________________ (extend)
/ SAP         sap2afp.dll %filter-options -r 300_________________ (extend)
_ XML         ____________________________________________________ (extend)
_ Other       ____________________________________________________ (extend)

_ Resubmit for filtering
Transforms to AFP:
  Fail on error . . . .  2   1. No  2. Error  3. Warning
  Trailer error page. .  1   1. No  2. Error  3. Warning
:
```

Explanation of fields:

- The **SAP** data format is selected because the `sap2afp2.dll` filter can transform SAP documents to AFP format. The transform options are:
  - The %filter-options filter option causes the transform to use transform options that are specified in the filter-options job attribute.
  - The -r option causes the transform to format the output for a 300-pel resolution printer.

    Because the -r option is specified to the right of %filter-options, the transforms ignore the -r option specified in the filter-options job attribute.
- The **Line data** and **MO:DCA-P** data formats are selected because PSF can print these data formats on AFP printers.

- The **Text** data format is selected because Infoprint Server automatically converts text data to line data when it processes data for PSF printers.
- In the **Fail on error** field, option **Error** is selected. If a data stream error occurs during the transform, the transform fails and does not create an output document.
- In the **Trailer error page** field, option **No** is selected. The transform does not write any warning or error messages to a trailer error page.

Chapter 5. Diagnosing errors

This information describes the diagnostic facilities that the transforms provide.

Submitting APARs

Report any difficulties using the transforms to your IBM Support Center. If an APAR is required, the Support Center can tell you where to send the required diagnostic information.

When you submit an APAR, use the component ID 5655N6001.

Using error messages

The transforms write error messages to these locations:

- `Trailer error page:` The transforms add to the end of the AFP output when a warning or data stream error occurs during the transform. The trailer error page contains warning and error messages from the transform to help you diagnose problems.

  To print AFP output without transform messages, you can suppress all messages on the trailer error page, or you can choose to suppress only transform warning messages. The transform always writes warning and error messages to the Infoprint Server common message log and the transform's `stderr` file so that you can find messages even if the transform does not write a trailer error page.

  You can use these methods to suppress messages on a trailer error page:

  – Specify the Infoprint Server trailer-transform-error-page job attribute with the print request.
  – Set the **Trailer error page** field in the printer definition that is used to print the document.
  – Set the AOP_TRAILER_ERROR_PAGE environment variable in the Infoprint Server transform configuration file, `aopxfd.conf`.
- `Infoprint Server common message log:` The transforms write all warning and error messages to the Infoprint Server common message log if the common message log is enabled in Infoprint Server.
- `Transform stderr file:` The transform's `stderr` file is a file that Infoprint Server administrators can access. In general, the transforms write messages in both the `stderr` file and the Infoprint Server common message log.

  For some error conditions, the PCL to AFP transform writes more detailed error messages to the transform's `stderr` file when tracing is turned on. If you have trouble diagnosing a problem, you can turn tracing on and look for more messages in the `stderr` file.

Finding the transform stderr file

You can find a transform's `stderr` file in the directory named *`base-directory`*`/xfd`, where *base-directory* is determined by the value of the base-directory attribute in the Infoprint Server configuration file, `aopd.conf`. The default base directory is named `/var/Printsrv`.

The `stderr` file-naming convention is:

```
transform[_class].#.stderr
```

transform
Specifies the transform name, which is defined in the Infoprint Server Transform Manager configuration file, `aopxfd.conf`. For example, `pcl2afp`.

class
Specifies the transform class, which is specified in the -c option when the transform is called. Transform classes are defined in the Infoprint Server Transform Manager configuration file, `aopxfd.conf`.

#
A unique number that the transform assigns. This number is incremented each time a new transform is started.

Tip: To read the transform's `stderr` file, you must be a member of the AOPADMIN group. For information about how to establish security for Infoprint Server administrators, see *z/OS Infoprint Server Customization*.

Running traces

This information describes how to run a trace of a transform. The service representative in the IBM Support Center might ask you to run a trace to aid in diagnosing a problem. If so, the representative will tell you how and where to send the trace information. You do not have to interpret the trace. Send it to your service representative.

Tracing can slow performance considerably. Turn tracing on for only as long as necessary to capture the error.

Tip: For some error conditions, the PCL to AFP transform writes more detailed error messages to the transform's `stderr` file when tracing is turned on. If you have trouble diagnosing a problem, you can turn tracing on and look for more messages in the `stderr` file.

To trace transforms, you can either specify the -T transform option or set the AOPTRACEON environment variable:

- Specify the -T option to trace a specific transform request, or to trace all transform requests for a specific printer.
- Set the AOPTRACEON environment variable to trace all transform requests, or to trace all transforms that use a transform class that you configure for tracing.

You can specify the -T option in these ways:

- On the `pcl2afp`, `pdf2afp`, `ps2afp`, or `sap2afp` command.
- In the filter-options job attribute on a print command. By using the `lp` command, the Infoprint Port Monitor for Windows, and some other print commands you can specify Infoprint Server job attributes.
- In the **Filter options** field in a printer definition in the Printer Inventory.

When you request a trace by using either the -T option or the AOPTRACEON environment variable, the transform writes the trace in a file in the directory that is specified in the AOPTRACEDIR environment variable or to the default directory (`/var/Printsrv/trace` or `/var/Printsrv/xfd`). One trace file can contain trace information for multiple print jobs because each instance of a transform appends trace information to the same trace file. The transform's `stderr` file contains either the name of the trace file or the trace information itself.

These environment variables in the Infoprint Server transform configuration file, `aopxfd.conf`, turn tracing on and specify where the trace is written:

AOPTRACEON
Set this variable to any value to turn on tracing. The transform traces all transform requests that use this transform class. Any value turns on tracing. To turn tracing off, do not specify this environment variable.

Default: Tracing is turned off.

Example: `environment = {AOPTRACEON -> 1}`

AOPTRACEDIR

Specify the full path name of the directory where the transform writes trace information. You can specify the same directory for different transform classes. The name of the trace file identifies the transform and transform class, and contains a time stamp.

This directory must exist. If the directory does not exist, the transform writes trace information to the transform's `stderr` file. For information about how to find the `stderr` file, see "Finding the transform stderr file" on page 105.

Defaults:

sap2afp and pcl2afp transforms

`AOPTRACEDIR -> /var/Printsrv/trace`

pdf2afp and ps2afp transforms

- When AOPTRACEON is turned on or when the permissions of `/var/Printsrv/trace` are `drwxrwxrwx` or `drwxrwx-wx`, `AOPTRACEDIR -> /var/Printsrv/trace`
- Otherwise, `AOPTRACEDIR -> /var/Printsrv/xfd`

When trace information is written to `/var/Printsrv/xfd`:

- The trace information is appended to the `ps2afp.#.stderr` file for each instance of the transform. The `stderr` file is automatically cleaned up when you restart Infoprint Server Transform Manager.
- RACF message ICH408I is written to the log. You can ignore it.

Examples:

```
environment = {AOPTRACEDIR -> /var/Printsrv/xfd}
environment = {AOPTRACEDIR -> .}
```

For an example of a transform entry in the transform configuration file that specifies these variables, see "Trace the transform" on page 67.

Tip: Specifying `AOPTRACEDIR -> .` has these benefits and limitations:

- The trace is recorded in the transform's working directory, for example, `/var/Printsrv/ps2afp.0.d`. This location keeps the trace closely associated with any other items that might be produced when the job runs (for example, a memory dump). The transform's working directory is a secure location, which protects any sensitive information trace might contain.
- The trace is automatically cleaned up when you restart the Infoprint Server Transform Manager.
- However, when you restart the Infoprint Server Transform Manager, the transform's working directory and its contents are deleted. Be sure to look in all the active transform directories to find the trace that you are interested in.

Examples:

1. To request a trace by specifying the -T option on a transform command:

   ```
   pdf2afp -o outputfile -T myfile.pdf
   ```

2. To request a trace by specifying the -T option on a transform command when you use the AOPBATCH program:

   ```
   //AOPBATCH JOB  ...
   //TRANSFRM EXEC PGM=AOPBATCH,PARM='/ps2afp -T -o //DD:OUTPUT DD:INPUT'
   //INPUT    DD   DSN=HLQ.INPUT.PS,DISP=SHR
   //OUTPUT   DD   DSN=HLQ.OUTPUT.AFP,DISP=(NEW,CATLG,DELETE),
   // DCB=(RECFM=VBM,LRECL=32756,BLKSIZE=32760),SPACE=(CYL,(1,1))
   //STDOUT   DD   SYSOUT=*
   //STDERR   DD   SYSOUT=*
   ```

   Note: To continue the PARM parameter:

   - Extend the parameter to column 71. Do not code an apostrophe in column 71.
   - Code // in columns 1 and 2 of the following statement.

- Continue the parameter in column 16 of the following statement even if doing this splits the parameter.

3. To request a trace by specifying the -T option on an `lp` command:

```
lp -d myprinter -o "filter-options='-T'" myfile.pdf
```

4. To request a trace by specifying the -T option in a printer definition:

```
                                Processing

Supported data formats and associated filters:
Data format:  Filter:
/ PostScript  ps2afp.dll %filter-options  -T_________________ (extend)
/ PCL         pcl2afp.dll %filter-options -T_________________ (extend)
/ PDF         pdf2afpdll %filter-options  -T_________________ (extend)
/ SAP         sap2afpdll %filter-options  -T_________________ (extend)
```

5. To submit a transform request to a transform class that specifies the AOPTRACEON environment variable. If the administrator defined a transform class that is called `trace` in the Infoprint Server transform configuration file (`aopxfd.conf`), use these commands to submit a trace request to it.

```
pdf2afp -c trace -o myfile.afp myfile.pdf
lp -o "filter-options='-c trace'" -d myprinter myfile.pdf
```

Chapter 6. Messages

This information describes the messages that the transforms produce.

For information about all the messages from Infoprint Server, including messages that are related to transforms, see *z/OS Infoprint Server Messages and Diagnosis*.

Message format

The messages have this format:

AOP*nnnnt*

AOP
Identifies messages from Infoprint Server

nnn
The message number

t
One-character type code:

Type code	Meaning
E	An error occurred.
I	Information message.
W	A warning situation occurred.

message_text
The text of the message.

Messages

AOP2026W The format of the input document might not be valid for the transform.

Explanation

The PostScript or PDF input data stream submitted to the PDF to AFP or PostScript to AFP transform might not conform to Adobe standards.

System action

Processing continues if possible.

System programmer response

Not applicable.

User response

If additional error information follows or the output is not correct, make sure your input datastream contains either PDF or PostScript data, and conforms to Adobe standards.

Problem determination

Not applicable.

Module

Not applicable.

Source

Infoprint Transforms to AFP for z/OS.

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

AOP2028I **A transform message was generated: *message***

Explanation

The data stream transform sent a message.

In the message text:

message
Message from the transform.

System action

The severity of the error determines whether Infoprint Server continues processing the document.

Operator response

Not applicable.

System programmer response

If the transform message indicates a transform problem, correct the problem and submit the transform or print request again.

User response

If the transform message indicates an error in the input data stream, correct the error and submit the transform or print request again.

Problem determination

Not applicable.

Module

Not applicable.

Source

Infoprint Server.

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

AOP2029E **The transform detected an error or degradation in the input datastream, or an error or degradation occurred while transforming the document. No output was produced.**

Explanation

A transform error occurred.

System action

The transform request was not completed. No output was produced.

Operator response

Not applicable.

System programmer response

For information about the transform problem, see any associated messages from the transform. After you correct the problem, you might need to restart the Infoprint Server Transform Manager.

To continue processing documents when this type of error occurs, do one of these:

- Set the AOP_FAIL_ON_ERROR -> no environment variable for the transform class in the transform configuration file, `aopxfd.conf`. For information about the environment variable, see the documentation for the transform.
- Set the **Fail on error** field to **No** in the printer definition.

User response

If there is an error in the input data stream, correct the error and submit the transform or print request again. If the input data stream is correct, notify the system programmer that this error occurred.

To continue processing the document when this error occurs, resubmit the print or transform request that specifies the fail-on-transform-error=no job attribute. For information about the fail-on-transform-error job attribute, see *z/OS Infoprint Server User's Guide*.

Problem determination

Not applicable.

Module

Not applicable.

Source

Infoprint Server.

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

AOP2061E The document contains EBCDIC data. The SAP to AFP transform can only transform ASCII or Unicode SAP data.

Explanation

The SAP to AFP transform detected EBCDIC data. The transform accepts only ASCII or Unicode data.

System action

The request is not completed.

System programmer response

Not applicable.

User response

In the SAP R/3 output device, select either SAPGOF (Generic Output Format) or SAPGOFU (Unicode encoded data).

Problem determination

Not applicable.

Module

Not applicable.

Source

Infoprint Transforms to AFP for z/OS

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

AOP2201E The owner of the ps2afpd transform file must not have a UID of 0, and the set-user-ID flag must be turned on.

Explanation

The Infoprint Server Transform Manager attempted to start the `ps2afpd` daemon. However, for security reasons, the Infoprint Server Transform Manager requires that (1) the owner of the executable file for the transform *not* have a UID of 0 and (2) the set-user-ID flag for the file is turned on. The message contains the name of the executable file.

System action

The Infoprint Server Transform Manager does not start the `ps2afpd` daemon. The Infoprint Server Transform Manager attempts to start other transform daemons configured in the transform configuration file, `aopxfd.conf`.

System programmer response

Use the `ls` command to list the owner of the file and to verify that the set-user-ID flag is on:

```
ls -l /usr/lpp/Printsrv/bin/ps2afpd
```

Output from the `ls` command should look like this, assuming that the owning user name is NOBODY and that the owning group name is NOGROUP:

```
-rws------  1  NOBODY NOGROUP ...
```

The lowercase letter s in the owner permissions section indicates that the set-user-ID flag is on and that the owner has permission to execute the file. Use the `id` command to determine the UID of the user.

If the user has a UID of 0, enter the z/OS UNIX `chown` command to change the owner of the file. For example, to change the owner of file `ps2afpd` to NOBODY, type:

```
chown NOBODY /usr/lpp/Printsrv/bin/ps2afpd
```

If the set-user-ID flag is off, or if you entered the `chown` command, use the `chmod` command to turn on

the set-user-ID flag. For example, to turn the flag on for file ps2afpd, type:

```
chmod u+s /usr/lpp/Printsrv/bin/ps2afpd
```

For more information, see "Customizing the PDF to AFP and PostScript to AFP transforms" on page 60.

User response

Contact your system programmer.

Problem determination

Not applicable.

Module

Not applicable.

Source

Infoprint Transforms to AFP for z/OS

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

AOP2500W Font substitution information: *date-time*

Explanation

The PDF to AFP or PostScript to AFP transform did not find one or more fonts that a document requested, so it substituted similar fonts. To determine the substitute fonts, the transform used a font-substitution algorithm that you cannot modify.

In the message text, *date-time* is the date and time the message was written.

The text that follows this message identifies the fonts that the document requested and the fonts that the transform substituted.

The AOP_FONT_SUBSTITUTION_MESSAGES environment variable determines whether the transform writes this message when it substitutes fonts in a document if no other errors occurred.

System action

The document is transformed without error.

System programmer response

In most cases, the substitute font results in acceptable output and no response is necessary. However, if the output is not acceptable, you might be able to add the original font to the transform. For information, see "Adding fonts" on page 69.

If you added a font to the transform and the transform still substitutes another font, make sure that you specified the correct name of the font directory in the AOP_RESOURCE_PATH environment variable in the transform configuration file, aopxfd.conf. For information, see Appendix A, "Environment variables," on page 123.

User response

None.

Problem determination

Not applicable.

Module

Not applicable.

Source

Infoprint Transforms to AFP for z/OS

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

AOP2501E The transform detected an error in the input data stream, or an error occurred while transforming the document. No output was produced.

Explanation

A transform error occurred.

System action

The transform request was not completed. No output was produced.

Operator response

None.

System programmer response

For information about the transform problem, see the accompanying messages from the transform. After you correct the problem, you might need to restart the Infoprint Server Transform Manager daemon.

If you do not want transform requests to fail when this type of error occurs, set the AOP_FAIL_ON_ERROR -> no environment variable for the transform instance in the transform configuration file, `aopxfd.conf`. If AOP_FAIL_ON_ERROR->no, the return code from the transform is 0. For information about how to edit the transform configuration file, see Chapter 3, "Customizing transforms," on page 53.

User response

If there is an error in the input data stream to be transformed, correct the error. If the input data stream is correct, notify the system programmer that this error occurred. After the error is corrected, resubmit the transform request.

Problem determination

Not applicable.

Module

Not applicable.

Source

Infoprint Transforms to AFP for z/OS

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

AOP2502I The transform attempted to convert the document to AFP format. The input data stream might not be valid. Diagnostic information from transform *transform* follows.

Explanation

The information that follows this message is from the transform. You can use this information to diagnose the problem. In the message text, *transform* is the name of the transform in the format: *transform_class.#*.

System action

Processing continues if possible.

System programmer response

If no data stream error is found, contact an IBM service representative.

User response

Use the diagnostic information to correct any data stream errors.

Problem determination

Not applicable.

Module

Not applicable.

Source

Infoprint Transforms to AFP for z/OS

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

AOP2503I The transform attempted to convert the document to AFP format. Trace information from transform *transform* follows.

Explanation

The diagnostic information that follows this message is from the transform. IBM can use this information to diagnose the transform problem. In the message text, *transform* is the name of the transform in the format: *transform_class.#*.

System action

Processing continues if possible.

System programmer response

If you cannot correct the error, contact an IBM service representative. Provide the diagnostic information to IBM.

User response

Notify your system programmer.

Problem determination

Not applicable.

Module

Not applicable.

Source

Infoprint Transforms to AFP for z/OS

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

AOP2504E The transform initialization failed. Diagnostic information from transform *transform* follows.

Explanation

The transform could not be initialized. In the message text, *transform* is the name of the transform in the format: *transform_class.#*.

System action

The transform request was not completed.

System programmer response

Contact an IBM service representative. Report the diagnostic information to IBM.

User response

Notify your system programmer.

Problem determination

Not applicable.

Module

Not applicable.

Source

Infoprint Transforms to AFP for z/OS

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

AOP2505E Not enough memory is available to transform the data stream. Make sure the region size is large enough and the -m option is not set in the transform configuration file.

Explanation

The transform could not obtain enough memory to convert the data stream. The data stream might be large or complex.

System action

The transform request is not completed. No output is produced.

System programmer response

Increase the amount of storage available to the transform:

- Increase the region size:
  - If you use the AOPSTART JCL procedure to start Infoprint Server, specify the region size in the REGION parameter of the EXEC statement.
  - If you issue the `aopstart` command from the z/OS UNIX command line during a TSO session, specify the region size in the SIZE option on the logon panel or logon procedure for the TSO user ID.
- Increase the maximum address space size for Infoprint Server:

  - Check the MAXASSIZE value in the BPXPRMxx member of SYS1.PARMLIB. MAXASSIZE sets the system-wide maximum address space size. You can set MAXASSIZE dynamically with the SETOMVS command. You can check MAXASSIZE from the operator console with this Display command: `d omvs,o`
  - Check the RACF ASSIZEMAX value for the user ID that starts Infoprint Server. ASSIZEMAX overrides the MAXASSIZE value. You can increase the ASSIZEMAX value on the RACF ALTUSER command.
- If you have IEFUSI exits that limit region sizes, make sure the limits in the IEFUSI exits do not apply to OMVS. Enter one of these commands:
  - If you do not want any exits to apply to OMVS:

    ```
    SETSMF SUBSYS(OMVS,NOEXITS)
    ```

  - If you need some exits to apply to OMVS:

    ```
    SETSMF SUBSYS(OMVS,EXITS(exits))
    ```

    In the EXITS parameter, specify the exits to apply to OMVS, but do not specify IEFUSI.

  For more information about the IEFUSI exit, see *z/OS MVS Installation Exits*.
- Make sure that the -m option is not set in the entry for this transform in the Infoprint Server transform configuration file, `aopxfd.conf`.

User response

Notify your system programmer. After the problem is corrected, resubmit the transform or print request.

Problem determination

Not applicable.

Module

Not applicable.

Source

Infoprint Transforms to AFP for z/OS

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

AOP2506W The transform converted the first page of the input file to a page segment or overlay because no page number was specified in the -p option.

Explanation

The transform request specified the -t page segment or -t overlay option to convert the input file to an AFP page segment or overlay. The transform can create an AFP page segment or overlay for only one page in the input file. Because the transform request did not identify which page in the input file to use for the page segment or overlay, the transform used the first page.

System action

The transform completed successfully.

System programmer response

None.

User response

If the transform did not create a page segment of overlay of the correct page, resubmit the request and specify the page in the transform -p option.

Problem determination

Not applicable.

Module

Not applicable.

Source

Infoprint Transforms to AFP for z/OS

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

AOP2507E This function is not allowed because the Infoprint Transforms to AFP product is not enabled.

Explanation

The product registration request failed.

System action

The transform creates no output.

System programmer response

Ensure that the product being used is listed and set to ENABLED on the system where it is being run. Check the enablement policy settings in SYS1.PARMLIB(IFAPRDxx) or contact an IBM representative. For more information, see "Enabling V2.4 transforms in the SYS1.PARMLIB member" on page 119.

User response

Contact your system programmer.

Problem determination

Not applicable.

Module

Not applicable.

Source

Infoprint Transforms to AFP for z/OS

Routing code

Not applicable.

Descriptor code

Not applicable.

Automation

Not applicable.

Chapter 7. Migrating to the latest release

This information describes how to migrate to Infoprint Transforms to AFP 2.4 for z/OS from Version 2.3 of the product. It describes these actions:

- Migrating from V2.3 to V2.4
- Using the new functions introduced in V2.4

Migrating from V2.3 to V2.4

Table 12 on page 117 lists the tasks you might need to do when you migrate from Infoprint Transforms to AFP V2.3 to V2.4. Required tasks apply to all installations. Optional tasks are required only if the listed condition applies to your installation. The table also indicates when to do each task.

Table 12: Tasks for migrating from Infoprint Transforms V2.3 to V2.4

Task	Condition	When to do the task
"Completing the installation" on page 117	Required.	After installing V2.4
"Removing the V2.3 resource directories" on page 118	Required.	After installing V2.4
"Customizing the aopxfd.conf file" on page 119	Required if the `aopxfd` file has been altered so that environment variables point to non-standard V2.3 resources.	After installing V2.4
"Enabling V2.4 transforms in the SYS1.PARMLIB member" on page 119	Required.	After installing V2.4

Completing the installation

The following information describes how to complete the installation of Infoprint Transforms to AFP.

About this task

Starting with V2.4, the IBM Infoprint Transforms to AFP are no longer installed in the `/usr/lpp/Printsrv` directory. The default directory path is `/usr/lpp/IBM/XformsToAFP/V2R4`. Changing the default directories from `/usr/lpp/Printsrv` provides you with the ability to conditionally install the product through Product ServerPac. To complete the installation and configuration, a one-time manual setup is required after you install the IBM Print Transforms to AFP.

Procedure

1. Run one of these commands to verify that the installation of the transform product was successful:
   - If the transforms are installed in the default directories, run this command :

     ```
     /usr/lpp/IBM/XformsToAFP/V2R4/samples/aoxcheck
     ```

   - If the transforms are not installed in the default directory, run this command:

     ```
     Transform_Install_Path/samples/aoxcheck Transform_Install_Path
       Transform_Install_Path/lib
     ```

2. **Required:** Create the symbolic links from the directory where Infoprint Server is installed to the directory where the transforms are installed by running one of these commands:

- If the transforms are installed in /usr/lpp/IBM/XformsToAFP/V2R4 and Infoprint Server is installed in /usr/lpp/Printsrv, run this command :

```
/usr/lpp/IBM/XformsToAFP/V2R4/samples/aoxsymlink.sh -f slinks
```

- If either product is installed outside the default directory, run this command:

```
Transform_Install_Path/samples/aoxsymlink.sh -f slinks -i
  Transform_Install_Path -b Infoprint_Server_Install_Path
```

This is a required step because it is not likely that the PATH and LIBPATH environment variables for Infoprint Server include the transform installation directory, even if it is the default directory.

3. Stop and restart the Infoprint Server Transform Manager daemon (aopxfd) when you want your configuration changes to take effect.

Removing the V2.3 resource directories

In V2.4, all transform resources are installed in different directories. Table 13 on page 118 lists the resource directories. If you have made changes to customize files or directories, make those changes in the new resource directories before you delete the previous version's directories.

Table 13: PCL, PDF, PostScript, and SAP transform resource directories in V2.3 and V2.4

Transform	V2.3 resource directories	V2.4 resource directories
PCL to AFP	/usr/lpp/Printsrv/pcl2afpv2.3/fonts	Fonts are not installed in a resource directory, but included with the transform.
PDF to AFP	/usr/lpp/Printsrv/ps2afpv2.3/lib /usr/lpp/Printsrv/ps2afpv2.3/Resource/Init /usr/lpp/Printsrv/ps2afpv2.3/Resource	/usr/lpp/Printsrv/ps2afpv2.4/lib /usr/lpp/Printsrv/ps2afpv2.4/Resource/Init /usr/lpp/Printsrv/ps2afpv2.4/Resource
PostScript to AFP	/usr/lpp/Printsrv/ps2afpv2.3/lib /usr/lpp/Printsrv/ps2afpv2.3/Resource/Init /usr/lpp/Printsrv/ps2afpv2.3/Resource	/usr/lpp/Printsrv/ps2afpv2.4/lib /usr/lpp/Printsrv/ps2afpv2.4/Resource/Init /usr/lpp/Printsrv/ps2afpv2.4/Resource
SAP to AFP	/usr/lpp/Printsrv/sap2afpv2.3/res /usr/lpp/Printsrv/sap2afpv2.3/icu	/usr/lpp/Printsrv/sap2afpv2.4/res /usr/lpp/Printsrv/sap2afpv2.4/icu

Note: In V2.4, resource directories in /usr/lpp/Printsrv are symbolically linked to the actual resource directories, which are in /usr/lpp/IBM/XformsToAFP/V2R4. See "Completing the installation" on page 117.

To remove the PCL to AFP, PDF to AFP, PostScript to AFP, and SAP to AFP V2.3 resource directories:

1. If the AOP_RESOURCE_PATH, AOP_SAP2AFP_RESOURCES, and AOP_SAP2AFP_ICU environment variables in the aopxfd.conf transform configuration file specify the name of a V2.3 resource directory, delete the directory name. You do not need to specify any of these unless the files are not in the standard path.
2. Restart the Infoprint Server Transform Manager to pick up the changes to the transform configuration file.

3. Delete the V2.3 resource directories because the install process does not delete them (the install usually deletes the files in the directories):

```
su
rm -r /usr/lpp/Printsrv/pcl2afpv2.3
rm -r /usr/lpp/Printsrv/ps2afpv2.3
rm -r /usr/lpp/Printsrv/sap2afpv2.3
```

Customizing the aopxfd.conf file

If you customized the `aopxfd.conf` file so that environment variables point to non-standard V2.3 resources, you must change the corresponding environment variables to point to V2.4 resources. This potentially affects:

- The PostScript to AFP and PDF to AFP AOP_RESOURCE_PATH environment variables
- The SAP to AFP variables AOP_SAP2AFP_RESOURCES and AOP_SAP2AFP_ICU

For information about how to create and edit the Infoprint Server transform configuration file, see *z/OS Infoprint Server Customization*. After you update the transform configuration file, you must restart the Infoprint Server Transform Manager.

Note: You can specify the AOP_SAP2AFP_RESOURCES environment variable in the SAP to AFP entry in the `aopxfd.conf` transform configuration file or in the AOPSTART EXEC. The `aopxfd.conf` file takes precedence over the AOPSTART EXEC.

Enabling V2.4 transforms in the SYS1.PARMLIB member

After you migrate to V2.4, you must enable the transforms to run.

Before you begin

To create or change an IFAPRDxx member, you must know how to define a system environment and how to define a product. If not, review all of the information about the z/OS product enablement policy, IFAPRDxx member of SYS1.PARMLIB, in *z/OS MVS Initialization and Tuning Guide*.

About this task

The following procedure enables the transforms by adding them to the IFAPRDxx member of SYS1.PARMLIB, which defines the enablement policy, and then activates the member.

Note: Adding a product or feature might require changes to other SYS1.PARMLIB members and an IPL before the product or feature can run. Ensure that SYS1.PARMLIB contains member IFAPRDxx and that the proper transforms are enabled in the member.

Procedure

1. **Required:** Add the PRODUCT entries for your transforms to the IFAPRDxx member of SYS1.PARMLIB. See the example in Figure 12 on page 119. IBM supplies a tailored IFAPRD00 member that reflects the features that your installation ordered.

   Note: The STATE value must be set to ENABLED.

```
PRODUCT OWNER('IBM CORP')
NAME('INFOPRINT XFORMS')
ID(5655-N60)
FEATURENAME('TRANSFORM TO AFP')
STATE(ENABLED)
```

   Figure 12: PRODUCT entry for transforms in the IFAPRDxx member of SYS1.PARMLIB

2. **Required:** Activate the transform products by using the MVS console command, SET PROD=*(xx)*) or through an IPL. In SET PROD=*(xx)*, the *(xx)* variable represents the two alphanumeric characters that

indicate the IFAPRDxx member that contains the transforms enablement policy. If a policy already exists, the system performs the actions defined in the specified member to modify the existing policy.

Note:

a. You can use the SET PROD command to modify the enablement policy dynamically by specifying which IFAPRDxx member the system is to use. Statements in the member only modify an existing policy. While the change to the policy takes place immediately, it does not affect any product instances that are already running.

b. The system does not automatically list the IFAPRDxx parameters at IPL or when you issue SET PROD command. Instead, you can enter the DISPLAY PROD,STATE command to display the active enablement policy.

What to do next

When the transform product is successfully enabled, you are ready to use the transform options. If the product is not enabled before you try to use it, you can expect the transform request to fail. You can also expect to see the following error message:

```
AOP2507E This function is not allowed because the Infoprint Transform to AFP
product is not enabled.
```

Using new functions introduced in V2.4

Table 14 on page 120 lists the customization tasks that you must do to use new functions introduced in Infoprint Transforms to AFP V2.4, and whether you can do each task before or after you install V2.4. You are not required to use the new functions; these tasks are optional.

Table 14: Optional Tasks for using new functions in V2.4

Task	When to do the task
"Setting a halftone threshold" on page 120	After installing V2.4 if you did not install APAR OA48323 on V2.3, at any time if you did
"Creating AFP output for 360-pel, 720-pel, and 1200-pel printers" on page 120	After installing V2.4

Setting a halftone threshold

In V2.3 with APAR OA48323 and in V2.4, you can specify the maximum percentage of grays and colors that is rendered in black. Above this threshold, grays and colors are rendered as halftones.

For both transforms, you can specify the using the -n option in a command.

To enter a command that transforms the PDF file `myfile.pdf` into a file called `myfile.afp` in which all grays and colors are rendered as black, enter:

```
pdf2afp -o myfile.afp -n 1 myfile.pdf
```

Creating AFP output for 360-pel, 720-pel, and 1200-pel printers

In V2.4, you can specify 360, 720, or 1200 resolution output for both the PDF to AFP and PostScript to AFP transforms.

For both transforms, you can specify the resolution using the -r option in a command.

To enter a command that transforms the PDF file `myfile.pdf` into a 1200 pels per inch file called `myfile.afp`, enter:

```
pdf2afp -o myfile.afp -r 1200 myfile.pdf
```

To enter a command that transforms the PostScript file `myfile.ps` into a 360 pels per inch file called `myfile.afp`, enter:

```
ps2afp -o myfile.afp -r 360 myfile.ps
```

Appendix A. Environment variables

Table 15 on page 123 lists all the environment variables the transforms use and indicates which transforms support the variable. For information about these environment variables, see:

- "Environment variables for the PCL to AFP transform" on page 55
- "Environment variables for the PDF to AFP and PostScript to AFP transforms" on page 62
- "Environment variables for the SAP to AFP transform" on page 72

Table 15: Environment variables the transforms support

Environment variable	PCL to AFP transform	PDF to AFP transform	PostScript to AFP transform	SAP to AFP transform
_BPX_JOBNAME	Yes	Yes	Yes	Yes
_CEE_DMPTARG	Yes	Yes	Yes	Yes
AOP_FAIL_ON_ERROR	Yes	Yes	Yes	Yes
AOP_FONT_SUBSTITUTION_ MESSAGES	No	Yes	Yes	No
AOP_HORIZONTAL_MARGINS	Yes	No	No	No
AOP_MVS_RETURN_CODES (see Note)	Yes	Yes	Yes	Yes
AOP_PAGE_HEIGHT	Yes	No	No	No
AOP_PAGE_WIDTH	Yes	No	No	No
AOP_RECLEN	Yes	Yes	Yes	Yes
AOP_RESOLUTION	Yes	No	No	No
AOP_RESOURCE_PATH	No	Yes	Yes	No
AOP_TRAILER_ERROR_PAGE	Yes	Yes	Yes	Yes
AOP_TRIM	Yes	Yes	Yes	No
AOP_SAP2AFP_ICU	No	No	No	Yes
AOP_SAP2AFP_RESOURCES	No	No	No	Yes
AOP_VERTICAL_MARGINS	Yes	No	No	No
AOPTRACEDIR	Yes	Yes	Yes	Yes
AOPTRACEON	Yes	Yes	Yes	Yes
AOP_WORLDTYPE_PATH	No	No	No	Yes

Note: Do not specify AOP_MVS_RETURN_CODES in the Infoprint Server transform configuration file as you do the other environment variables. For information about where to specify this environment variable, see "AOP_MVS_RETURN_CODES environment variable" on page 51.

Appendix B. Fonts

This information lists the fonts that the PCL to AFP, PDF to AFP, and PostScript to AFP transforms support. It also lists the TrueType fonts that the SAP transform supports.

PCL to AFP transform fonts

These are the Monotype MicroType fonts that are built into the PCL to AFP transform.

Albertus Extra Bold	Clarendon Condensed Bold	Symbol
Albertus Medium	Coronet	Times New Roman
Antique Olive	Courier	Times New Roman Bold
Antique Olive Bold	Courier Bold	Times New Roman Bold Italic
Antique Olive Italic	Courier Bold Italic	Times New Roman Italic
Arial	Courier Italic	Univers Bold
Arial Bold	Garamond Antiqua	Univers Bold Italic
Arial Bold Italic	Garamond Halbfett	Univers Condensed Bold
Arial Italic	Garamond Kursiv	Univers Condensed Bold Italic
CG Omega	Garamond Kursiv Halbfett	Univers Condensed Medium
CG Omega Bold	Letter Gothic	Univers Condensed Medium Italic
CG Omega Bold Italic	Letter Gothic Bold	Univers Medium
CG Omega Italic	Letter Gothic Italic	Univers Medium Italic
CG Times	Marigold	WingDings
CG Times Bold		
CG Times Bold Italic		
CG Times Italic		

PDF to AFP and PostScript to AFP transform fonts

Table 16 on page 125 lists the Ghostscript fonts that the PDF to AFP and PostScript to AFP transforms use. The Ghostscript fonts are functional equivalents of the corresponding PostScript fonts in that the typeface styles are similar and the font metrics are identical to provide the same pagination and line endings.

Table 16: PDF to AFP and PostScript to AFP transform fonts

For this PostScript font:	**The transform uses this Ghostscript font:**
NewCenturySchlbk-BoldItalic	CenturySchL-BoldItal
NewCenturySchlbk-Bold	CenturySchL-Bold
NewCenturySchlbk-Italic	CenturySchL-Ital
NewCenturySchlbk-Roman	CenturySchL-Roma
ZapfDingbats	Dingbats
Courier-BoldOblique	NimbusMonL-BoldObli
Courier-Bold	NimbusMonL-Bold
Courier-Oblique	NimbusMonL-ReguObli

Table 16: PDF to AFP and PostScript to AFP transform fonts (continued)	
For this PostScript font:	**The transform uses this Ghostscript font:**
Courier	NimbusMonL-Regu
Times-BoldItalic	NimbusRomNo9L-MediItal
Times-Bold	NimbusRomNo9L-Medi
Times-Italic	NimbusRomNo9L-ReguItal
Times-Roman	NimbusRomNo9L-Regu
Helvetica-Narrow-BoldOblique	NimbusSanL-BoldCondItal
Helvetica-Narrow-Bold	NimbusSanL-BoldCond
Helvetica-BoldOblique	NimbusSanL-BoldItal
Helvetica-Bold	NimbusSanL-Bold
Helvetica-Narrow-Oblique	NimbusSanL-ReguCondItal
Helvetica-Narrow	NimbusSanL-ReguCond
Helvetica-Oblique	NimbusSanL-ReguItal
Helvetica	NimbusSanL-Regu
Symbol	StandardSymL
Bookman-DemiItalic	URWBookmanL-DemiBoldItal
Bookman-Demi	URWBookmanL-DemiBold
Bookman-LightItalic	URWBookmanL-LighItal
Bookman-Light	URWBookmanL-Ligh
ZapfChancery-MediumItalic	URWChanceryL-MediItal
AvantGarde-BookOblique	URWGothicL-BookObli
AvantGarde-Book	URWGothicL-Book
AvantGarde-DemiOblique	URWGothicL-DemiObli
AvantGarde-Demi	URWGothicL-Demi
Palatino-BoldItalic	URWPalladioL-BoldItal
Palatino-Bold	URWPalladioL-Bold
Palatino-Italic	URWPalladioL-Ital
Palatino-Roman	URWPalladioL-Roma

SAP to AFP transform TrueType fonts

Table 17 on page 126 lists the TrueType fonts that the `sap2afp` transform supports.

Table 17: SAP to AFP TrueType fonts		
Font file name	**Full font name**	**Localization**
wt__j__b.ttf	WT Serif J	Japanese

Table 17: SAP to AFP TrueType fonts (continued)		
Font file name	**Full font name**	**Localization**
wt__k__b.ttf	WT Serif K	Korean
wt__s__b.ttf	WT Serif SC	Simplified Chinese
wt__tt_b.ttf	WT Serif TW	Traditional Chinese Taiwan
wt__w___.ttf	WT Serif	N/A
wt_dj__b.ttf	WT SerifDuo J	Japanese
wt_dk__b.ttf	WT SerifDuo K	Korean
wt_ds__b.ttf	WT SerifDuo SC	Simplified Chinese
wt_dtt_b.ttf	WT SerifDuo TW	Traditional Chinese Taiwan
wt_dw___.ttf	WT SerifDuo	N/A
wts_w___.ttf	WT Sans	N/A
wtsdj__b.ttf	WT SansDuo J	Japanese
wtsdk__b.ttf	WT SansDuo K	Korean
wtsds__b.ttf	WT SansDuo SC	Simplified Chinese
wtsdsxb_.ttf	WT SansDuo SC xB	Simplified Chinese
wtsdtt_b.ttf	WT SansDuo TW	Traditional Chinese Taiwan
wtsdw___.ttf	WT SansDuo	N/A

Appendix C. Accessibility

Accessible publications for this product are offered through IBM Knowledge Center (www.ibm.com/support/knowledgecenter/SSLTBW/welcome).

If you experience difficulty with the accessibility of any z/OS information, send a detailed email message to mhvrcfs@us.ibm.com.

Accessibility features

Accessibility features help users who have physical disabilities such as restricted mobility or limited vision use software products successfully. The accessibility features in z/OS can help users do the following tasks:

- Run assistive technology such as screen readers and screen magnifier software.
- Operate specific or equivalent features by using the keyboard.
- Customize display attributes such as color, contrast, and font size.

Consult assistive technologies

Assistive technology products such as screen readers function with the user interfaces found in z/OS. Consult the product information for the specific assistive technology product that is used to access z/OS interfaces.

Keyboard navigation of the user interface

You can access z/OS user interfaces with TSO/E or ISPF. The following information describes how to use TSO/E and ISPF, including the use of keyboard shortcuts and function keys (PF keys). Each guide includes the default settings for the PF keys.

- *z/OS TSO/E Primer*
- *z/OS TSO/E User's Guide*
- *z/OS ISPF User's Guide Vol I*

Dotted decimal syntax diagrams

Syntax diagrams are provided in dotted decimal format for users who access IBM 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 that are offered in the USA or elsewhere.

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

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

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
United States of America

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

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

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

This information could include missing, incorrect, or broken hyperlinks. Hyperlinks are maintained in only the HTML plug-in output for the Knowledge Centers. Use of hyperlinks in other output formats of this information is at your own risk.

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

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

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Site Counsel
2455 South Road

Poughkeepsie, NY 12601-5400
USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

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

COPYRIGHT LICENSE:

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

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or

reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

IBM Online Privacy Statement

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user, or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

Depending upon the configurations deployed, this Software Offering may use session cookies that collect each user's name, email address, phone number, or other personally identifiable information for purposes of enhanced user usability and single sign-on configuration. These cookies can be disabled, but disabling them will also eliminate the functionality they enable.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at ibm.com/privacy and IBM's Online Privacy Statement at ibm.com/privacy/details in the section entitled "Cookies, Web Beacons and Other Technologies," and the "IBM Software Products and Software-as-a-Service Privacy Statement" at ibm.com/software/info/product-privacy.

Policy for unsupported hardware

Various z/OS elements, such as DFSMS, JES2, JES3, and MVS, contain code that supports specific hardware servers or devices. In some cases, this device-related element support remains in the product even after the hardware devices pass their announced End of Service date. z/OS may continue to service element code; however, it will not provide service related to unsupported hardware devices. Software problems related to these devices will not be accepted for service, and current service activity will cease if a problem is determined to be associated with out-of-support devices. In such cases, fixes will not be issued.

Minimum supported hardware

The minimum supported hardware for z/OS releases identified in z/OS announcements can subsequently change when service for particular servers or devices is withdrawn. Likewise, the levels of other software products supported on a particular release of z/OS are subject to the service support lifecycle of those products. Therefore, z/OS and its product publications (for example, panels, samples, messages, and product documentation) can include references to hardware and software that is no longer supported.

- For information about software support lifecycle, see: IBM Lifecycle Support for z/OS (www.ibm.com/software/support/systemsz/lifecycle)
- For information about currently-supported IBM hardware, contact your IBM representative.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available at Copyright and Trademark information (www.ibm.com/legal/copytrade.shtml).

Adobe and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

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.

Index

Special Characters

Numerics

A

B

C

D

E

F

H

I

J

K

L

M

N

O

P

R

S

T

U

W

X

G550-0443-06