



System i  
Programming  
Infoprint Server for iSeries commands

*Version 6 Release 1*







System i  
Programming  
Infoprint Server for iSeries commands

*Version 6 Release 1*

**Note**

Before using this information and the product it supports, be sure to read the information in "Notices," on page 37.

This edition applies to version 6, release 1, modification 0 of IBM Infoprint Server for iSeries (product number 5722-IP1) and to all subsequent releases and modifications until otherwise indicated in new editions. This version does not run on all reduced instruction set computer (RISC) models nor does it run on CICS models.

© Copyright International Business Machines Corporation 1998, 2008. All rights reserved.

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

---

## Contents

Create AFP Data (CRTAFPDTA) . . . . .	1	Start Font Downloader (STRFNTDWN)	29
Create PDF Map (CRTPDFMAP) . . . . .	15	Start Transform Manager (STRTFMMGR) . . . . .	33
Display PDF Map Entries (DSPPDFMAPE) . . . . .	19	Work with PDF Map Entries (WRKPDFMAPE) . . . . .	35
End Font Downloader (ENDFNTDWN)	25	Appendix. Notices . . . . .	37
End Transform Manager (ENDTFMMGR)	27		



---

## Create AFP Data (CRTAFPDTA)

Where allowed to run: All environments (\*ALL)  
Threadsafe: No

Parameters  
Examples  
Error messages

The Create AFP Data (CRTAFPDTA) command provides the following functions:

1. Convert line data or AFPDS line data into AFPDS data. Input is a spooled file.
2. Index a document to enhance one's ability to view, archive, or retrieve individual pages or groups of pages from large documents. The document index is written to a stream file. The input data may be line, AFPDS line or AFPDS.
3. Retrieve and package AFP resources needed for viewing a document. The resources are written to a stream file. The input data may be line, AFPDS line or AFPDS.
4. Merge the above-mentioned stream files so that the spooled can be viewed on a workstation using the AFP Viewer.

Input is taken from a spooled file. Output is to one or more stream files.

**Note:** The following values are obtained from the spooled file attributes, they cannot be specified on CRTAFPDTA: Control character, Table reference characters and DBCS SO/SI spacing.

Refer to the InfoPrint Server for OS/400: User's Guide for additional information.

---

## Error messages for CRTAFPDTA

### \*ESCAPE Messages

#### IPS301E

CRTAFPDTA command terminated with errors.

Top

---

## Parameters

Keyword	Description	Choices	Notes
FROMSPLF	From spooled file	<i>Name</i>	Required, Positional 1
TOSTMF	To stream file	<i>Path name</i> , *DFT	Optional, Positional 2
JOB	Job name	Single values: * Other values: <i>Qualified job name</i>	Optional, Positional 3
	Qualifier 1: Job name	<i>Name</i>	
	Qualifier 2: User	<i>Name</i>	
	Qualifier 3: Number	000000-999999	
SPLNBR	Spooled file number	1-999999, *ONLY, *LAST, *ANY	Optional, Positional 4
JOBSYSNAME	Job system name	<i>Character value</i> , *ONLY, *CURRENT, *ANY	Optional, Positional 5

Keyword	Description	Choices	Notes
CRTDATE	Spooled file created	Single values: <b>*ONLY</b> , <b>*LAST</b> Other values: <i>Element list</i>	Optional, Positional 6
	Element 1: Creation date	<i>Date</i>	
	Element 2: Creation time	<i>Time</i> , <b>*ONLY</b> , <b>*LAST</b>	
FORMDF	Form definition	Single values: <b>*SPLF</b> , <b>*INLINE</b> Other values: <i>Qualified object name</i>	Optional, Positional 7
	Qualifier 1: Form definition	<i>Name</i>	
	Qualifier 2: Library	<i>Name</i> , <b>*LIBL</b> , <b>*CURLIB</b>	
PAGDFN	Page definition	Single values: <b>*SPLF</b> , <b>*INLINE</b> Other values: <i>Qualified object name</i>	Optional, Positional 8
	Qualifier 1: Page definition	<i>Name</i>	
	Qualifier 2: Library	<i>Name</i> , <b>*LIBL</b> , <b>*CURLIB</b>	
TOIDXSTMF	To index stream file	<i>Path name</i> , <b>*NONE</b> , <b>*DFT</b>	Optional, Positional 9
TORSCSTMF	To resource stream file	<i>Path name</i> , <b>*NONE</b> , <b>*DFT</b>	Optional, Positional 10
IDXTRG	Index trigger definition	Values (up to 4 repetitions): <i>Element list</i>	Optional, Positional 11
	Element 1: Record number	0-255, *	
	Element 2: Column number	1-32756, *	
	Element 3: Trigger value	<i>Character value</i>	
IDXTAGFLD	Index field definition	Values (up to 16 repetitions): <i>Element list</i>	Optional, Positional 12
	Element 1: Record number or literal value	<i>Character value</i>	
	Element 2: Column number	1-32756	
	Element 3: Length	1-250	
IDXTAG	Index tag definition	Values (up to 8 repetitions): <i>Element list</i>	Optional, Positional 13
	Element 1: Literal value	<i>Character value</i>	
	Element 2: Field	Values (up to 16 repetitions): <b>*IDXTAGFLD1</b> , <b>*IDXTAGFLD2</b> , <b>*IDXTAGFLD3</b> , <b>*IDXTAGFLD4</b> , <b>*IDXTAGFLD5</b> , <b>*IDXTAGFLD6</b> , <b>*IDXTAGFLD7</b> , <b>*IDXTAGFLD8</b> , <b>*IDXTAGFLD9</b> , <b>*IDXTAGFLD10</b> , <b>*IDXTAGFLD11</b> , <b>*IDXTAGFLD12</b> , <b>*IDXTAGFLD13</b> , <b>*IDXTAGFLD14</b> , <b>*IDXTAGFLD15</b> , <b>*IDXTAGFLD16</b>	
IDXCDEPAG	Index code page identifier	1-65535, <u>500</u>	Optional, Positional 14
IDXPAGNAM	Generate page names	<b>*GEN</b> , <b>*FILE</b>	Optional, Positional 15
IDXGRPNAM	Generate group names	<b>*YES</b> , <b>*NO</b>	Optional, Positional 16
IDXGRP	Select index for group names	<b>*IDXTAG1</b> , <b>*IDXTAG2</b> , <b>*IDXTAG3</b> , <b>*IDXTAG4</b> , <b>*IDXTAG5</b> , <b>*IDXTAG6</b> , <b>*IDXTAG7</b> , <b>*IDXTAG8</b>	Optional, Positional 17
IDXOBJ	Information for index file	<b>*GROUP</b> , <b>*ALL</b>	Optional, Positional 18
IDXPAGLMT	Indexing page limit	1-99, <u>1</u>	Optional, Positional 19
RSCDTA	Resource data	Single values: <b>*ALL</b> Other values (up to 7 repetitions): <b>*BARCODE</b> , <b>*FONT</b> , <b>*FORMDF</b> , <b>*GOCA</b> , <b>*IOCA</b> , <b>*OVL</b> , <b>*PAGESEG</b>	Optional, Positional 20
TOMRGSTMF	To merged stream file	<i>Element list</i>	Optional, Positional 21
	Element 1: Stream file name	<i>Path name</i> , <b>*NONE</b> , <b>*DFT</b>	
	Element 2: Keep or delete input	<b>*KEEPINPUT</b> , <b>*DLTINPUT</b>	

Keyword	Description	Choices	Notes
AFPCHARS	AFP characters	Values (up to 4 repetitions): <i>Character value</i> , <b>*SPLF</b> , *NONE	Optional, Positional 22
IMAGEOUT	Image output	<b>*ASIS</b> , *IOCA	Optional, Positional 23

Top

---

## From spooled file (FROMSPLF)

This is a required parameter.

Specifies the name of the spooled file that contains the data which will be processed by CRTAFPDTA.

The possible values are:

*spooled-file-name*

Specifies the name of the spooled file to be processed by CRTAFPDTA.

The following types of data can be processed by CRTAFPDTA:

- LINE
- AFPDSLIN
- AFPDS

Top

---

## To stream file (TOSTMF)

Specifies the stream file to which the transformed data is written.

The possible values are:

**\*DFT** The transformed data will be written to file **output.afp** in the current working directory. If the data is being written to qsys.lib, the file name used is **outputafp.mbr**.

*to-stream-file-path*

Specifies the path name of the stream file to which transformed data is written. All directories in the path name must exist. New directories are not created. If the stream file does not exist, it is created. If you specify the file name without a path, the output file is placed in your current working directory. If the file exists, it is overwritten. If you name a directory or path but not a file name, a file named **output.afp** is written. If the path name is qualified, it must be enclosed in apostrophes.

Top

---

## Job name (JOB)

Specifies the name of the job that created the spooled file whose data records are to be read.

The possible values are:

**\*** The job that issued this command is the job that created the spooled file.

*job-name*

Specifies the name of the job that created the spooled file.

*user-name*

Specifies the user name that identifies the user profile under which the job is run.

*job-number*

Specifies the system-assigned job number.

Top

---

## Spooled file number (SPLNBR)

Specifies the number of the spooled file whose data records are to be read.

The possible values are:

\*ONLY

Only one spooled file from the job has the specified file name; therefore, the number of the spooled file is not necessary. If \*ONLY is specified and more than one spooled file has the specified name, an error message is issued.

\*LAST

The highest-numbered spooled file with the specified file name is read.

*spooled-file-number*

Specify the number of the spooled file whose data records are read.

Top

---

## Job system name (JOBSYSNAME)

Specifies the name of the system where the job that created the spooled file (JOB parameter) ran. This parameter is considered after the job name, user name, job number, spooled file name, and spooled file number parameter requirements have been met.

The possible values are:

\*ONLY

There is one spooled file with the specified job name, user name, job number, spooled file name, spooled file number, and spooled file create date and time.

\*CURRENT

The spooled file created on the current system with the specified job name, user name, job number, spooled file name, spooled file number, and create date and time is used.

*job-system-name*

Specify the name of the system where the job that created the spooled file ran.

Top

---

## Spooled file created (CRTDATE)

Specifies the date and time the spooled file was created. This parameter is considered after the job name, user name, job number, spooled file name, spooled file number, and job system name parameter requirements have been met.

The possible single values are:

**\*ONLY**

There is one spooled file with the specified job name, user name, job number, spooled file name, spooled file number, and job system name.

**\*LAST**

The spooled file with the latest create date and time of the specified job name, user name, job number, spooled file name, spooled file number, and job system name is used.

**spooled-file-create-date**

Specify the date the spooled file was created.

The possible create time values are:

**\*ONLY**

There is one spooled file with the specified job name, user name, job number, spooled file name, spooled file number, job system name, and spooled file create date.

**\*LAST**

The spooled file with the latest create time of the specified job name, user name, job number, spooled file name, spooled file number, job system name, and spooled file create date is used.

**spooled-file-create-time**

Specify the time the spooled file was created.

Top

---

## Form definition (FORMDF)

Specifies the form definition to use when processing the spooled file. A form definition is a resource object that defines the characteristics of the form (physical page) such as: overlays, position of page data on the form, number of copies of pages, and modification to pages.

The possible values are:

**\*SPLF** The value specified for **Form definition** (FORMDF) in the spooled file's attributes is used. CRTAFPDTA cannot use \*DEVDF from the spooled file's FORMDF attribute. Either change the spooled file attribute by naming a form definition or specify a form definition to CRTAFPDTA.

**\*INLINE**

A form definition is included in the input data.

***form-definition-name***

Specifies the name of the form definition. This form definition must exist in the library specified. Valid values range from 1 to 8 characters.

The possible library values are:

**\*LIBL** All libraries in the job's library list are searched until the first match is found.

**\*CURLIB**

The current library for the job in which the form definition is located. If no library is specified as the current library for the job, QGPL is used.

***library-name***

Specifies the library where the form definition is located.

If the form definition is not found when either \*LIBL or \*CURLIB is specified for the library, the font libraries QFNTCPL, QFNT01-QFNT19, and QFNT61-QFNT69 are searched.

Top

---

## Page definition (PAGDFN)

Specifies the name of the page definition to be used to format line data. This parameter is used only if the input data stream is either line (\*LINE) or mixed mode (\*APFDLINE).

The possible values are:

**\*SPLF** The value specified for **Page definition** (PAGDFN) in the spooled file's attributes is used. CRTAFPDTA cannot use \*DEVDTA from the spooled file's PAGDFN attribute. Either change the spooled file attribute by naming a page definition or specify a page definition to CRTAFPDTA.

**\*INLINE**

A page definition is included within the input data.

*page-definition-name*

Specify the name of the page definition that must exist in the library specified. Valid values range from 1 to 8 characters.

The possible library values are:

**\*LIBL** All libraries in the job's library list are searched until the first match is found.

**\*CURLIB**

The current library for the job in which the page definition is located. If no library is specified as the current library for the job, QGPL is used.

*library-name*

Specifies the library where the page definition is located.

If the page definition is not found when either \*LIBL or \*CURLIB is specified for the library, the font libraries QFNTCPL, QFNT01-QFNT19, and QFNT61-QFNT69 are searched.

Top

---

## To index stream file (TOIDXSTMF)

Specifies the stream file to which the index object file is written.

The possible values are:

**\*NONE**

The index object information is not written to a stream file.

**\*DFT** The index object information is written to the file **output.idx** in the current working directory. If the data is being written to qsys.lib, the file name used is **outputidx.mbr**.

*to-index-stream-file*

Specifies the path name of the stream file to which the document index is written. All directories in the path name must exist. New directories are not created. If the stream file does not exist, it is created. If you specify the file name without a path, the output file is placed in your current working directory. If the file exists, it is overwritten. If you name a directory or path but not a file name, a file named **output.idx** is written. If the path name is qualified, it must be enclosed in apostrophes.

Top

---

## To resource stream file (TORSCSTMF)

Specifies the stream file to which resources are written.

The possible values are:

### **\*NONE**

Resources are not written to a stream file.

**\*DFT** Resources are written to the file **output.rsc** in the current working directory. If the data is being written to `qsys.lib`, the file name used is **outputrsc.mbr**.

### *to-resource-stream-file-path*

Specifies the path name of the stream file to which the resource data is written. All directories in the path name must exist. New directories are not created. If the stream file does not exist, it is created. If you specify the file name without a path, the output file is placed in your current working directory. If the file exists, it will be overwritten. If you name a directory or path but not a file name, a file named **output.rsc** is written. If the path name is qualified, it must be enclosed in apostrophes.

Top

---

## Index trigger definition (IDXTRG)

Specifies the locations and values of data fields within the input file that are to be used to define indexing groups in the file. These data fields are referred to as triggers, because their presence in the file triggers a processing action. The number of IDXTRG specifications required to uniquely identify the beginning of a group of pages within the file depends on the complexity of the application output. The first IDXTRG specification is special. Each record in the file containing the value specified in the first IDXTRG specification is referred to as an indexing anchor record.

This parameter is only valid if TOIDXSTMF has a value other than \*NONE.

The possible values are:

### *record-number*

Specifies the relative record number from the indexing anchor record. The anchor record is the first IDXTRG specification in the list of IDXTRG values. After the first IDXTRG specification has been found, all other IDXTRG specifications are specified as a relative offset from the first IDXTRG specification.

The possible values for **record-number** are:

### **record-number**

A value ranging from -255 to 255.

**\*** This value may be specified only in the first IDXTRG specification. This special value indicates that every record should be checked for the presence of the first IDXTRG specification.

### *column-number*

Specifies the byte offset from the beginning of the record where the IDXTRG value is located.

The possible values for **column-number** are:

### **column-number**

A value ranging from 1 to 32756. A value of 1 refers to the first byte in the record. For files containing carriage control characters, column 1 refers to the carriage control

character. CRTAFPDTA compares the trigger-value to the input data. If the specified value exceeds the physical length of the record, CRTAFPDTA considers the comparison "false" and continues processing.

- \* The record is scanned from left to right, looking for the trigger-value.

#### *trigger-value*

Specifies the actual alphanumeric (case-sensitive) or hexadecimal value of the trigger. CRTAFPDTA does not perform any validity checking on this value, but uses it in performing a byte-for-byte comparison with the records in the file. The trigger-value can be from 1-250 bytes in length. If the combined values of column and the trigger-value length exceed the physical length of the record, CRTAFPDTA considers the comparison "false" and continues processing.

A maximum of four IDXTRG values may be specified. CRTAFPDTA will number each trigger based on the sequence in which they are specified. For example, if a message indicates an error with respect to 'Index trigger definition number 4', a reference is being made to the fourth trigger in the series of triggers you specified.

Top

---

## Index field definition (IDXTAGFLD)

Specifies the data fields to be used to construct the indexing information. These data fields can be specified as either literal values or CRTAFPDTA can retrieve the data from the input records of the file.

This parameter is only valid if TOIDXSTMF has a value other than \*NONE.

The possible values are:

#### *record-number or literal-value*

Specifies either the relative record number from the indexing anchor record or a literal value. When CRTAFPDTA indexes a file, it uses the information specified in the IDXTRG parameters to determine a page-group boundary. When all the specified IDXTRG values are true, CRTAFPDTA defines the indexing anchor record as the record where the first IDXTRG definition is located. The first IDXTRG definition becomes the reference point from which all indexing information is located.

You may specify either a record number or literal value.

- A record number value specifies the relative record number from the indexing anchor record. If you specify a record number, you must also specify the column and length fields. To specify a record number, enter the number without quotes.
- A literal value specifies a constant value. Literal values do not have column or length values associated with them. You may not specify column and length values when specifying a literal value. To specify a literal numeric value, enclose the value in apostrophes. If the value begins with a non-numeric value, quotes are not needed.

#### *column*

The byte offset from the beginning of the record. A value of 1 refers to the first byte in the record. For files containing carriage control characters, column 1 refers to the carriage control character. For those applications that use a specific carriage control character to define page boundaries (for example, skip to channel 1), consider defining the value of the carriage control character as one of the IDXTRG parameters. If the specified value exceeds the physical length of the record, CRTAFPDTA reports an error condition and terminates processing.

Specify a column number only when a record number has been specified.

**length** The number of contiguous bytes (characters) starting at the value specified for column that compose this field.

The field can extend outside the record length, as long as the column where it begins lies within record length. In this case, CRTAFPDTA adds padding blanks (X'40') to fill out the record. If the field begins outside the maximum length of the record, CRTAFPDTA reports an error condition and terminates processing.

Specify a length only when a record number has been specified.

A maximum of 16 fields can be defined. Each field is numbered automatically by CRTAFPDTA based on the sequence in which you specify the values, starting with the number 1.

You may use IDXTAGFLD definitions as part of an index definition using the IDXTAG parameter. For example, if you wish to reference the third IDXTAGFLD specification when specifying an IDXTAG, then you must specify \*IDXTAGFLD3.

Top

---

## Index tag definition (IDXTAG)

Specifies the content of the indexing tags for the entire file.

If literal values are specified for every IDXTAG specification, CRTAFPDTA treats the entire file as one page group and uses this information to index the document. CRTAFPDTA stops processing the input file if literal values are specified for all IDXTAG parameters and if any IDXTRG parameters are also specified.

This parameter is only valid if TOIDXSTMF has a value other than \*NONE.

The possible values are:

### *literal-value*

Specifies a user-defined attribute name to be associated with the actual index value. For example, assume the first IDXTAG definition is a person's bank account number. The string 'account number' would be a meaningful attribute name. The value of IDXTAG would be the account number (for example, 1234567). Think of the attribute name as a label for the actual index value. The attribute name is an EBCDIC character string from 1-250 bytes in length. CRTAFPDTA does not perform any validity checking on the contents of the attribute name.

**field** Specifies one or more IDXTAGFLD definitions that compose the index value. A maximum of 16 IDXTAGFLD definitions can be specified. If more than one IDXTAGFLD definition is specified, CRTAFPDTA concatenates them into one physical string of data. No delimiters are used between the concatenated fields. Because an IDXTAG value has a maximum length of 250 bytes, the total of all specified IDXTAGFLD definitions for a single index cannot exceed this length. CRTAFPDTA reports an error condition and terminates processing if this occurs.

A maximum of 8 indexes can be defined, and each index can be made up of more IDXTAGFLD definitions.

For identification purposes, CRTAFPDTA will number each index tag based on the sequence in which they are specified. For example, if a message indicates an error with respect to 'Index tag definition number 2', a reference is being made to the second index in the series of index tags you specified. You will need to specify index tag values by number (i.e., \*IDXTAG1) for the IDXGRP parameter.

Top

---

## Index code page identifier (IDXCDEPAG)

Specifies the code page identifier for the index values and attribute names produced by CRTAFPDTA.

This value is used by programs that must display indexing information. These programs use this value in conjunction with code page translation tables to represent the information.

This parameter is only valid if TOIDXSTMF has a value other than \*NONE.

The possible values are:

**500** Code page identifier 500.

*code-page-identifier*

A code page identifier between 1-65535.

Top

---

## Generate page names (IDXPAGNAM)

Specifies whether CRTAFPDTA generates page names using an 8-byte counter or uses structured field tokens found in the input data stream. If the input data contains BPGs with FQNs, CRTAFPDTA does not generate page names.

Use of this parameter requires technical knowledge of the makeup of the input file. Such users will understand the explanation of this parameter. All other users should use the default value of \*GEN.

This parameter is only valid if TOIDXSTMF has a value other than \*NONE.

The possible values are:

**\*GEN** Page names are generated using an 8-byte counter.

**\*FILE** Page names are generated using structured field tokens in the input data stream.

Top

---

## Generate group names (IDXGRPNAM)

Specifies whether CRTAFPDTA creates a unique group name by generating an 8-character numeric string and appending the string to the group name.

This parameter is only valid if TOIDXSTMF has a value other than \*NONE.

The possible values are:

**\*YES** CRTAFPDTA generates an 8-character numeric string and appends the string to the group name.

**\*NO** CRTAFPDTA does not generate the string.

Top

---

## Select index for group names (IDXGRP)

Specifies which of the eight possible IDXTAG values should be used as the group name for each index group. Using the most unique index value for the group name is recommended. The intent is to have a unique group name for every group CRTAFPDTA produces in the output file. The value includes the IDXTAGFLD definitions from the IDXTAG parameter but does not include the attribute name.

The AFP Viewer displays this value along with the attribute name and index value. You can use the group name to select a group of pages to be viewed.

This parameter is only valid if TOIDXSTMF has a value other than \*NONE.

The possible values are:

**\*IDXTAG1**

Use the first IDXTAG value.

**\*IDXTAG2-\*IDXTAG8**

Use the specified IDXTAG value.

Top

---

## Information for index file (IDXOBJ)

Specifies how much information CRTAFPDTA writes to the index object file.

This parameter is only valid if TOIDXSTMF has a value other than \*NONE.

The possible values are:

**\*GROUP**

Group-level entries only are put into the index object file.

**\*ALL** Both page-level and group-level entries are put into the index object file. Only specify this value if you have TLEs in the input datastream. For information about TLEs, refer to [AFP Toolbox for Multiple Operating Systems: User's Guide](#) or the [DDS Reference: Printer Files](#) topic in the iSeries Information Center.

Top

---

## Indexing page limit (IDXPAGLMT)

Specifies the output page number by which CRTAFPDTA must find an indexing field, if CRTAFPDTA is indexing the file. If CRTAFPDTA does not find an indexing field, processing stops.

This parameter is helpful if your file contains header pages. You can specify a page number one greater than the number of header pages, so that CRTAFPDTA will continue to look for matches for the number of pages specified for this parameter.

This parameter is only valid if TOIDXSTMF has a value other than \*NONE.

The possible values are:

**1** An indexing field must be found in the first output page.

***indexing-page-limit***

Any number from 1 through 99.

---

## Resource data (RSCDTA)

Specifies the types of resources to be written to the stream file specified on the TORSCSTMF parameter.

This parameter is only valid if TORSCSTMF has a value other than \*NONE. One or more resource data types must be specified if TORSCSTMF has a value other than \*NONE.

A maximum of 7 resources may be specified if you are naming specific resource types.

The possible values are:

**\*ALL** All resources required to print or view the output document, specified with TOSTMF, be included in the file specified with the TORSCSTMF parameter.

When \*ALL is specified, you may not specify other resource types.

**\*BARCODE**

Specifies that all BCOCA objects included by an Include Object (IOB) structured field required to print or view the output document be included in the resource stream file.

**\*FONT**

Specifies that all font character sets, code pages and coded fonts required to print or view the output file be included in the resource stream file.

**\*FORMDF**

Specifies that the form definition used in processing the file be included in the resource stream file.

**\*GOCA**

Specifies that all GOCA objects included by an IOB structured field required to print or view the output document file be included in the resource stream file.

**\*IOCA**

Specifies that all IOCA objects included by an IOB structured field required to print or view the output document file be included in the resource stream file.

**\*OVL** Specifies that all overlays required to print or view the output document file be included in the resource stream file.

**\*PAGSEG**

Specifies that all page segments required to print or view the output document file be included in the resource stream file.

Top

---

## To merged stream file (TOMRGSTMF)

Specifies the stream file in which the following stream files are written:

- To stream file (TOSTMF)
- To index stream file (TOIDXSTMF), if a stream file name is specified.
- To resource stream file (TORSCSTMF), if a stream file name is specified.

Specify a stream file path if you wish to view the input spooled file, save the exact document for archival purposes or send the document to another system.

You may also specify whether the stream files which are used as input to the merged stream file are kept or deleted.

The possible values are:

#### Element 1: Stream File Name

##### \*NONE

Stream files are not merged.

**\*DFT** Stream files will be written to file **output.mrg** in the current working directory. If the data is being written to qsys.lib, the file name used is **outputmrg.mbr**.

##### *to-merged-stream-file-path*

Specifies the path name of the stream file to which the above-mentioned stream files are written. All directories in the path name must exist. New directories are not created. If the stream file does not exist, it is created. If you specify the file name without a path, the output file is placed in your current working directory. If the file exists, it will be overwritten. If the file exists, it will be overwritten. If you name a directory or path but not a file name, a file named **output.mrg** is written. If the path name is qualified, it must be enclosed in apostrophes.

#### Element 2: Keep or Delete Input Files

##### \*KEEPINPUT

No input stream files are deleted.

##### **\*DLTINPUT**

The stream files used as input for the merged stream file will be deleted after the merged stream file has been written.

Top

---

## AFP characters (AFPCHARS)

Specifies one or more AFP characters (coded fonts) to be used with line data and a page definition.

The possible values are:

**\*SPLF** The value for AFP characters as specified in the spooled file attributes for the named spooled file will be used.

##### *coded-font*

Specify up to four 4-byte names. The 4-byte names are concatenated to 'X0' to identify up to four coded fonts which are to be used when **Table reference characters** (TBLREFCHR) are being used within the data.

##### **\*NONE**

No AFP characters (coded fonts) specified.

Top

---

## Image output (IMAGEOUT)

Specifies the format of the image data produced by CRTAFPDTA in the output document.

The possible values are:

**\*ASIS** Image data is output exactly as in the input file.

**\*IOCA**

All image data is output in uncompressed IOCA format.

[Top](#)

---

## **Examples**

None

[Top](#)

---

## **Error messages**

### **\*ESCAPE Messages**

#### **IPS301E**

CRTAFPDTA command terminated with errors.

[Top](#)

---

## Create PDF Map (CRTPDFMAP)

Where allowed to run: All environments (\*ALL)  
Threadsafe: No

Parameters  
Examples  
Error messages

The Create PDF Map (CRTPDFMAP) command creates a PDF map.

This PDF map can be used by Infoprint Server and PSF/400 when converting spooled files to PDF. Each PDF map can be made up of several PDF map entries. The PDF map entries allow you to specify how spooled files should be processed after they have been converted.

These functions for processing are available:

- Storing the PDF file in the integrated file system.
- E-mailing the PDF file as an attachment.
- Spooling the PDF file to an output queue.
- Spooling the output AFP file to an output queue.

Add an entry to the PDF map using the WRKPDFMAPE command for each type of processing you want. For example, if you want to e-mail some spooled files, respool some as AFP files, and do both actions on some spooled files, you will need at least three PDF map entries.

### Restrictions:

1. Infoprint Server for iSeries is required to use this command.
2. \*IOSYSCFG authority is required to use this command.

Top

---

## Parameters

Keyword	Description	Choices	Notes
PDFMAP	PDF Map	<i>Qualified object name</i>	Required, Positional 1
	Qualifier 1: PDF Map	<i>Name</i>	
	Qualifier 2: Library	<i>Name, *CURLIB, *LIBL</i>	
TEXT	Text 'description'	<i>Character value, *BLANK</i>	Optional, Positional 3
AUT	Authority	<i>Name, *LIBCRTAUT, *CHANGE, *ALL, *USE, *EXCLUDE</i>	Optional, Positional 2

Top

---

## PDF Map (PDFMAP)

Specifies the name of the PDF map.

This is a required parameter.

### Qualifier 1: PDF Map

*pdf-map-name*

Specifies the name of the PDF map.

**Qualifier 2: Library**

**\*CURLIB**

The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

**\*LIBL** All libraries in the job's library list are searched until the first match is found.

*library-name*

Specifies the name of the library to be searched.

Top

---

## Text 'description' (TEXT)

Specifies the text that briefly describes the PDF map.

**\*BLANK**

No text is specified.

*'text description'*

Specify no more than 50 characters of text, enclosed in apostrophes.

Top

---

## Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority to the PDF map, who are not on an authorization list, and whose group profile has no specific authority to the PDF map.

**\*LIBCRTAUT**

The system determines the authority for the PDF map by using the value specified on the Create authority prompt (CRTAUT parameter) on the Create Library command (CRTLIB) for the library containing the PDF map to be created. If the value specified on the Create authority prompt (CRTAUT parameter) is changed, the new value will not affect any existing PDF maps.

**\*CHANGE**

Change authority allows the user to change and perform basic functions on the PDF map. Change authority provides PDF map operational authority and all data authorities.

**\*ALL** The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the PDF map's existence, specify the security for the PDF map, change the PDF map, and perform basic functions on the PDF map. The user can change ownership of the PDF map.

**\*USE** Use authority provides PDF map operational authority, read authority, and execute authority.

**\*EXCLUDE**

The user cannot access the PDF map.

*authorization-list-name*

Specify the name of an authorization list to be used for authority to the PDF map. Users included in the authorization list are granted authority to the PDF map as specified in the list. The authorization list must exist when the PDF map is created.

Top

---

## Examples

```
CRTPDFMAP    PDFMAP(QGPL/BIGSALES)
              TEXT('High-end sales letters')
```

This example shows how to create a PDF map.

[Top](#)

---

## Error messages

### \*ESCAPE Messages

#### CPF2283

Authorization list &1 does not exist.

#### CPF88C1

Printer resource type &1 &2 was not created in library &3.

#### CPF9810

Library &1 not found.

#### CPF9820

Not authorized to use library &1.

[Top](#)



---

## Display PDF Map Entries (DSPPDFMAPE)

Where allowed to run:

- Interactive job (\*INTERACT)

Threadsafe: No

Parameters  
Examples  
Error messages

The Display PDF Map Entries (DSPPDFMAPE) command displays entries in the specified PDF map.

A filter can be specified to control which entries are displayed. Each map entry has criteria to specify which spooled files it applies to. The filter you create for the Display PDF map entries command searches those criteria to determine which entries to display.

All the fields for the filter are used when scanning each entry in the PDF map for a match.

Top

---

### Parameters

Keyword	Description	Choices	Notes
PDFMAP	PDF Map	<i>Qualified object name</i>	Required, Positional 1
	Qualifier 1: PDF Map	<i>Name</i>	
	Qualifier 2: Library	<i>Name, *LIBL, *CURLIB</i>	
FILTER	Filter	<i>Element list</i>	Optional, Positional 2
	Element 1: Output queue	<i>Qualified object name</i>	
	Qualifier 1: Output queue	<i>Generic name, name, *ALL</i>	
	Qualifier 2: Library	<i>Generic name, name, *LIBL, *CURLIB, *ALL</i>	
	Element 2: Spooled file	<i>Generic name, name, *ALL</i>	
	Element 3: Job name	<i>Generic name, name, *ALL</i>	
	Element 4: User	<i>Generic name, name, *ALL</i>	
	Element 5: User data	<i>Character value, *ALL</i>	
	Element 6: Form type	<i>Character value, *ALL</i>	
Element 7: Routing tag	<i>Character value, *ALL</i>		

Top

---

### PDF Map (PDFMAP)

Specifies the name of the PDF map to be displayed.

This is a required parameter.

#### Qualifier 1: PDF map

*pdf-map-name*

Specifies the name of the PDF map.

#### Qualifier 2: Library

**\*LIBL** All libraries in the job's library list are searched until the first match is found.

## **\*CURLIB**

The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

### *library-name*

Specifies the name of the library to be searched.

Top

---

## **Filter (FILTER)**

Specifies which entries in the PDF map to display. If you want to see all entries, specify \*ALL for each filter entry. This is the default. If you want to see only certain types of entries, for example, all entries for spooled files on a specific output queue, enter values for any of the filter entries.

### **Element 1: Output queue**

#### **Qualifier 1: Output queue**

**\*ALL** Specifies that any value for the output queue field in the PDF map entry satisfies this field in the filter. Do not specify a value for the library.

#### *output-queue-name*

Specifies the output queue name used as a search argument when scanning PDF map entries.

#### **generic-name**

Specify the generic name of the object to be considered a match. A generic name is a character string that contains one or more characters followed by an asterisk (\*). All objects that have names with the same prefix as the generic name are considered a match for this field.

#### *output-queue-name*

Specifies that only a specific value in the Output queue name field in a PDF map entry satisfies this field in the filter. You must specify a value for the library.

Specify a library name only if you have specified a value other than \*ALL for the output queue name. Otherwise, the library value is ignored.

#### **Qualifier 2: Library**

#### *library-name*

Specifies a specific value for the output queue library name argument. Only an exact match in the Library field in a PDF map entry satisfies this field in the filter.

#### **generic-name**

Specify the generic name of the output queue's library on the PDF map entry. A generic name is a character string that contains one or more characters followed by an asterisk (\*). All objects that have names with the same prefix as the generic name are considered a match.

**\*LIBL** Specifies that only a value of \*LIBL in the output queue's Library field in a PDF map entry satisfies this field in the filter.

## **\*CURLIB**

Specifies that only a value of \*CURLIB in the output queue's Library field in a PDF map entry satisfies this field in the filter.

**\*ALL** Specifies that any value for the output queue's Library field in a PDF map entry satisfies this field in the filter.

### **Element 2: Spooled file**

Specifies the spooled file name to be used as a search argument when scanning PDF map entries.

**\*ALL** Specifies that any value for the PDF map entry's Spooled file field satisfies this field in the filter.

***spooled-file-name***

Specifies a specific value for the PDF map entry's Spooled file argument. Only an exact match in the Spooled file field satisfies this field in the filter.

**generic-name**

Specify the generic name of the object to be considered a match. A generic name is a character string that contains one or more characters followed by an asterisk (\*). All objects that have names with the same prefix as the generic name are considered a match for this field.

**Element 3: Job name**

Specifies the job name to be used as a search argument when scanning PDF map entries.

**\*ALL** Specifies that any value for the PDF map entry's Job name field satisfies this field in the filter.

***job-name***

Specifies a specific value for the PDF map entry's Job name argument. Only an exact match in the Job name field satisfies this field in the filter.

**generic-name**

Specify the generic name of the object to be considered a match. A generic name is a character string that contains one or more characters followed by an asterisk (\*). All objects that have names with the same prefix as the generic name are considered a match for this field.

**Element 4: User**

Specifies the user name to be used as a search argument when scanning PDF map entries.

**\*ALL** Specifies that any value for the PDF map entry's User field in the PDF map entry satisfies this field in the filter.

***user-name***

Specifies a specific value for the PDF map entry's user name argument. Only an exact match in the User field satisfies this field in the filter.

**generic-name**

Specify the generic name of the object to be considered a match. A generic name is a character string that contains one or more characters followed by an asterisk (\*). All objects that have names with the same prefix as the generic name are considered a match for this field.

**Element 5: User data**

Specifies the user data used as a search argument when scanning PDF map entries.

**\*ALL** Specifies that any value for the User data field in the PDF map entry satisfies this field in the filter.

***user-data***

Specifies a specific value for the user data argument. Only an exact match in the User data field in a PDF map entry satisfies this field in the filter.

**Element 6: Form type**

Specifies the form type to be used as a search argument when scanning PDF map entries.

**\*ALL** Specifies that any value for the PDF map entry's Form type field in the PDF map entry satisfies this field in the filter.

### *form-type*

Specifies a specific value for the PDF map entry's Form type argument. Only an exact match in the Form type field satisfies this field in the filter.

### **Element 7: Routing tag**

Specifies the routing tag used as a search argument when scanning PDF map entries.

**\*ALL** Specifies that any value for the PDF map entry's Routing tag field in the PDF map entry satisfies this field in the filter.

### *routing-tag*

Specifies a specific value for the PDF map entry's Routing tag argument. Only an exact match in the routing tag field satisfies this field in the filter.

Top

---

## **Examples**

### **Example 1:**

```
DSPPDFMAPE    PDFMAP(QGPL/BIGSALES)
               FILTER(*ALL *ALL *ALL *ALL *ALL *ALL)
```

This command displays all map entries in the PDF map object.

### **Example 2:**

```
DSPPDFMAPE    PDFMAP(QGPL/BIGSALES)
               FILTER(QGPL/SPECIAL *ALL *ALL *ALL *ALL *ALL)
```

This command displays all map entries in BIGSALES that specify an output queue of QGPL/SPECIAL.

### **Example 3:**

```
DSPPDFMAPE    PDFMAP(QGPL/BIGSALES)
               FILTER(*LIBL/SPECIAL *ALL BIGTICKET *ALL *ALL *ALL)
               OUTPUT(*PRINT)
```

This command displays all map entries in BIGSALES that specify an output queue of \*LIBL/SPECIAL and have a value for User data of 'BIGTICKET'.

Top

---

## **Error messages**

### **\*ESCAPE Messages**

#### **CPF9801**

Object &2 in library &3 not found.

#### **CPF9802**

Not authorized to object &2 in &3.

**CPF9803**

Cannot allocate object &2 in library &3.

**CPF9810**

Library &1 not found.

**CPF9820**

Not authorized to use library &1.

[Top](#)



---

## End Font Downloader (ENDFNTDWN)

Where allowed to run: All environments (\*ALL)  
Threadsafe: No

[Parameters](#)  
[Examples](#)  
[Error messages](#)

The End Font Downloader (ENDFNTDWN) command ends the font downloader server job and closes the TCP/IP port being used by the server job.

The server job will be ended immediately, whether or not it is receiving a font from a font downloader client application.

---

### Error messages for ENDFNTDWN

#### \*ESCAPE Messages

##### IPS2204

Infoprint Server Font Downloader was not started.

##### IPS2205

Font Downloader ended.

##### IPS2207

Infoprint Server Font Downloader could not be ended.

[Top](#)

---

### Parameters

None

[Top](#)

---

### Examples

None

[Top](#)

---

### Error messages

#### \*ESCAPE Messages

##### IPS2204

Infoprint Server Font Downloader was not started.

##### IPS2205

Font Downloader ended.

##### IPS2207

Infoprint Server Font Downloader could not be ended.

[Top](#)



---

# End Transform Manager (ENDTFMMGR)

**Where allowed to run:** All environments (\*ALL)  
**Threadsafe:** No

[Parameters](#)  
[Examples](#)  
[Error messages](#)

The End Transform Manager (ENDTFMMGR) command is used to end any Infoprint Server for iSeries transform manager server jobs that are running in the QUSRWRK subsystem. If you are currently converting any spool files when the ENDTFMMGR command is issued, the spool files will be held on the output queue.

## Restrictions:

1. \*JOBCTL special authority is required to run this command.

---

## Error messages for ENDTFMMGR

### \*ESCAPE Messages

CPF222E

&1 special authority is required.

[Top](#)

---

## Parameters

None

[Top](#)

---

## Examples

None

[Top](#)

---

## Error messages

### \*ESCAPE Messages

CPF222E

&1 special authority is required.

[Top](#)



---

## Start Font Downloader (STRFNTDWN)

Where allowed to run: All environments (\*ALL)  
Threadsafe: No

Parameters  
Examples  
Error messages

The Start Font Downloader (STRFNTDWN) command starts the Infoprint Server font downloader server job in the QUSRWRK subsystem. This server job listens on a TCP/IP port for Infoprint Manager DBCS Font Downloader connections. After connecting, new or refreshed PostScript fonts can be sent to the iSeries server for use with Infoprint Server. The font downloader server job will receive and install these fonts. These fonts will be installed into the /QOpenSts/QIBM/UserData/InfoprintServer/Transforms/ps2afp/fonts directory. Any existing files by the same name in this directory will be overwritten. The files will be owned by user profile QIPJOB. Once installed, the new PostScript fonts can be used by Infoprint Server for iSeries.

Only one font downloader server job can be active. An error message will be sent if STRFNTDWN is run and a font downloader server job is active. You can run the End Font Downloader (ENDFNTDWN) command to end the active font downloader server job.

**Restriction:** Only one Font Downloader daemon may be running at a time.

---

## Error messages for STRFNTDWN

### \*ESCAPE Messages

#### IPS2201

Infoprint Server Font Downloader could not create required working directory &2.

#### IPS2202

Infoprint Server Font Downloader could not create the required working file &2.

#### IPS2203

Infoprint Server Font Downloader could not start the font installation job.

#### IPS2204

Infoprint Server Font Downloader was not started.

#### IPS2205

Font Downloader ended.

#### IPS2206

Font Downloader encountered unrecoverable internal errors.

#### IPS2207

Infoprint Server Font Downloader could not be ended.

#### IPS2208

Infoprint Server Font Downloader is starting.

#### IPS2209

Infoprint Server Font Downloader is already started.

#### IPS4006

Portable App Solutions Environment option is not installed.

#### CPF9E72

Usage limit of &4 exceeded. Grace period expires in &6 days on &5.

## CPF9E78

The license key for product &1, license term &2, feature &3 is no longer valid.

Top

---

## Parameters

Keyword	Description	Choices	Notes
PORT	TCP/IP port number	1-65533, <u>8251</u>	Optional, Positional 1
INACTTIMO	Inactivity timeout	1-9999, <u>20</u>	Optional, Positional 2

Top

---

## TCP/IP port number (PORT)

Specifies the desired TCP/IP port to listen for Infoprint Manager DBCS Font Downloader for AIX/NT connections.

### *TCP/IP-port-number*

Specify the TCP/IP port number that the font downloader job should listen on.

Top

---

## Inactivity timeout (INACTTIMO)

Specify an inactive time-out period in minutes. Valid values range from 1 through 9999 minutes.

### *inactivity-timeout*

Specifies the number of minutes of inactivity between a font downloader client and the font downloader server before the font downloader server job automatically ends.

Top

---

## Examples

None

Top

---

## Error messages

### \*ESCAPE Messages

#### IPS2201

Infoprint Server Font Downloader could not create required working directory &2.

#### IPS2202

Infoprint Server Font Downloader could not create the required working file &2.

#### IPS2203

Infoprint Server Font Downloader could not start the font installation job.

- IPS2204**  
Infoprint Server Font Downloader was not started.
- IPS2205**  
Font Downloader ended.
- IPS2206**  
Font Downloader encountered unrecoverable internal errors.
- IPS2207**  
Infoprint Server Font Downloader could not be ended.
- IPS2208**  
Infoprint Server Font Downloader is starting.
- IPS2209**  
Infoprint Server Font Downloader is already started.
- IPS4006**  
Portable App Solutions Environment option is not installed.
- CPF9E72**  
Usage limit of &4 exceeded. Grace period expires in &6 days on &5.
- CPF9E78**  
The license key for product &1, license term &2, feature &3 is no longer valid.

Top



---

# Start Transform Manager (STRTFMMGR)

Where allowed to run: All environments (\*ALL)  
Threadsafe: No

Parameters  
Examples  
Error messages

The Start Transform Manager (STRTFMMGR) command starts the Infoprint Server for iSeries transform manager server job in the QUSRWRK subsystem. The transform manager job will start additional server jobs to perform print transform requests.

The STRTFMMGR command will use information in the configuration file `qxtrtfmmgr.cfg` located in the `/QOpenSys/QIBM/UserData/InfoprintServer/Transforms` directory. You can edit this configuration file by using the Edit File (EDTF) command. For the format of this configuration file, refer to the Infoprint Server for iSeries User's Guide

## Restrictions:

1. Only one transform manager job can run on an iSeries system at a time.
2. \*JOBCTL special authority is required to run this command.

---

## Error messages for STRTFMMGR

### \*ESCAPE Messages

#### IPS4001

Infoprint Server for iSeries Transform Manager already started.

#### IPS4005

Infoprint Server Transform Manager not started.

#### CPF222E

&1 special authority is required.

Top

---

## Parameters

None

Top

---

## Examples

None

Top

---

## Error messages

### \*ESCAPE Messages

#### IPS4001

Infoprint Server for iSeries Transform Manager already started.

**IPS4005**

Infoprint Server Transform Manager not started.

**CPF222E**

&1 special authority is required.

[Top](#)

---

## Work with PDF Map Entries (WRKPDFMAPE)

### Where allowed to run:

- Interactive job (\*INTERACT)

Threadsafe: No

Parameters  
Examples  
Error messages

Use the Work with PDF Map Entry display to set up map entries. These entries specify how you want Inflight Server to process spooled files after they have been converted to PDF.

You can select options to Add, Change, Copy, Remove, Display, or Print PDF Map Entries. Each PDF map entry has have two parts:

- Selection for the input spooled file. Spooled files that meet all of the selection criteria for a given PDF map entry are processed in the way specified by that entry. If a spooled file meets the selection criteria for more than one entry, it is processed by the PDF map entry with the smaller sequence number. See the Add PDF Map Entry - Input Selection Panel for more information.
- PDF map actions for the spooled file. For each spooled file, you can define one or more map actions and their associated parameters. You can specify any combination of these actions:
  - Send the output PDF file as e-mail.
  - Store the output PDF file as a stream file in the Integrated File System.
  - Place the output PDF file on an output queue.
  - Place the output AFP file on an output queue.

See the Add PDF Map Entry - Map Action Panel for more information.

Top

---

## Parameters

Keyword	Description	Choices	Notes
PDFMAP	PDF Map	<i>Qualified object name</i>	Required, Positional 1
	Qualifier 1: PDF Map	<i>Name</i>	
	Qualifier 2: Library	<i>Name, *LIBL, *CURLIB</i>	

Top

---

## PDF Map (PDFMAP)

Specifies the name of the PDF map.

This is a required parameter.

### Qualifier 1: PDF Map

*pdf-map-name*

Specifies the name of the PDF map.

### Qualifier 2: Library

\*LIBL All libraries in the job's library list are searched until the first match is found.

**\*CURLIB**

The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

***library-name***

Specifies the name of the library to be searched.

Top

---

**Examples**

None

Top

---

**Error messages****\*ESCAPE Messages****CPF2105**

Object &1 in &2 type \*&3 not found.

**CPF2110**

Library &1 not found.

**CPF2114**

Cannot allocate object &1 in &2 type \*&3.

**CPF2182**

Not authorized to library &1.

**CPF2189**

Not authorized to object &1 in &2 type \*&3.

Top

---

## Appendix. Notices

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

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

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

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

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation  
Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106-0032, Japan

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

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

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

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

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation

Software Interoperability Coordinator, Department YBWA  
3605 Highway 52 N  
Rochester, MN 55901  
U.S.A.

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, IBM License Agreement for Machine Code, 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.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. \_enter the year or years\_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

---

## Programming interface information

This Infoprint Server for iSeries commands publication documents intended Programming Interfaces that allow the customer to write programs to obtain the services of IBM Infoprint Server for iSeries.

---

## Trademarks

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

Advanced Function Printing  
AFP  
AS/400  
CICS  
COBOL/400  
C/400  
DataPropagator  
DB2  
Distributed Relational Database Architecture  
Domino  
DRDA  
IBM  
Infoprint  
InfoWindow  
i5/OS  
iSeries  
Integrated Language Environment  
Lotus  
LPDA  
OfficeVision  
Print Services Facility  
RPG/400  
System i  
System x  
SystemView  
System/36  
TCS  
Tivoli  
WebSphere  
z/OS

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

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

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

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

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

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

---

## Terms and conditions

Permissions for the use of these publications is granted subject to the following terms and conditions.

**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 works 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.

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.

---

## Code license and disclaimer information

IBM grants you a nonexclusive copyright license to use all programming code examples from which you can generate similar function tailored to your own specific needs.

SUBJECT TO ANY STATUTORY WARRANTIES WHICH CANNOT BE EXCLUDED, IBM, ITS PROGRAM DEVELOPERS AND SUPPLIERS MAKE NO WARRANTIES OR CONDITIONS EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT, REGARDING THE PROGRAM OR TECHNICAL SUPPORT, IF ANY.

UNDER NO CIRCUMSTANCES IS IBM, ITS PROGRAM DEVELOPERS OR SUPPLIERS LIABLE FOR ANY OF THE FOLLOWING, EVEN IF INFORMED OF THEIR POSSIBILITY:

1. LOSS OF, OR DAMAGE TO, DATA;
2. DIRECT, SPECIAL, INCIDENTAL, OR INDIRECT DAMAGES, OR FOR ANY ECONOMIC CONSEQUENTIAL DAMAGES; OR
3. LOST PROFITS, BUSINESS, REVENUE, GOODWILL, OR ANTICIPATED SAVINGS.

SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF DIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, SO SOME OR ALL OF THE ABOVE LIMITATIONS OR EXCLUSIONS MAY NOT APPLY TO YOU.





Printed in USA