

---

## Chapter 1. OS/400 V5R1 printing enhancements

This chapter describes the new printing enhancements at OS/400 V5R1. These may be grouped into the following categories:

- Internet Printing Protocol Server
- Printer file and DDS enhancements
- Unicode
- Java print support
- Miscellaneous print enhancements

---

### 1.1 Internet Printing Protocol Server

With V5R1, OS/400 becomes an IPP-enabled print server. IPP is the Internet Printing Protocol, a standard likely to replace the default TCP/IP printing protocol, LPR/LPD (Line Printer Requester/Line Printer Daemon). Most iSeries users will know the latter through implementation of a remote output queue, usually printing to an ASCII printer on the LAN. LPR/LPD has a number of limitations, including no page range support and little or no printer status or error recovery.

With IPP, the end user has a single interface from which they may submit and manage print jobs over TCP/IP to any iSeries-attached printer. The process is the same for print jobs from the local area network, intranet or the Internet. This is particularly convenient for users working remotely, for example a travelling sales representative or a home-office worker.

#### 1.1.1 IPP as a standard

The Internet Printing Protocol was developed by the Printer Working Group, (<http://www.pwg.org/>), a consortium of companies, including IBM, organized to promote printing standards.

IPP is transported over HTTP/1.1 and uses port number 631. The application layer in which HTTP operates is at a higher level than the standard LPR/LPD function. IPP is a client-server protocol: an IPP client submits a print request to an IPP server. The latter is either an IPP-capable printer or a print server with IPP capability. The target IPP printer/print server would usually be defined by an internet URL, but could also be defined directly by an IP address or via a directory that provides mapping to the URL or IP address.

##### 1.1.1.1 IPP object types

The IPP standard defines two basic object types:

- an IPP printer object
- an IPP job object

These objects can each have *operations* (performed by them and requested of them) and *attributes* (their current and default state). Operations include not only the actual printing but also a variety of other tasks such as getting the printer status, holding a print job, releasing a print job, restarting a print job, cancelling a print job and getting a print job list. Attributes include printer language, print orientation, paper size and the number of copies.

### 1.1.2 IPP Clients, Servers, Printers

Table 1, “Examples of IPP Clients, Servers and printers” on page 4 shows various clients and servers utilizing IPP technology. Note that some IPP clients also have IPP server capability.

Table 1. Examples of IPP Clients, Servers and printers

IPP Clients	IPP Servers	IPP Printers
Windows 95	OS/400 V5R1	IBM Infoprint Color 8
Windows 98	OS/390 V2R8	IBM Infoprint 21
Windows 2000		IBM Infoprint 70
OS/390 V2R8		
Infoprint Manager for Windows NT and 2000	Infoprint Manager for Windows NT and 2000	
Infoprint Manager for AIX	Infoprint Manager for AIX	

In this context, an “IPP Printer” is one which can receive IPP print jobs without an intermediate IPP server (in fact the server function is performed within the printer, although it cannot act as a server for other printers). As we shall see, you do not necessarily require an IPP-enabled printer in order to perform IPP printing: this is where the iSeries IPP Server comes in.

### 1.1.3 IPP on the iSeries

With OS/400 V5R1, the iSeries acts as an IPP print server. OS/400 is responsible for passing on the print request to an attached printer. This might involve a further print process, for example Host Print Transform (for printing to an ASCII printer) or PSF/400 (for printing to an AFP printer) but from this point on the print process is “business as normal”. It is important to realize that the target printer does not itself have to be IPP-capable but it does have to be attached to the iSeries server. In Figure 1, “The iSeries as an IPP Server” on page 4, the iSeries printer could be any already-attached printer, even a twinaxial printer. All that is required on the IPP client is an appropriate printer driver.

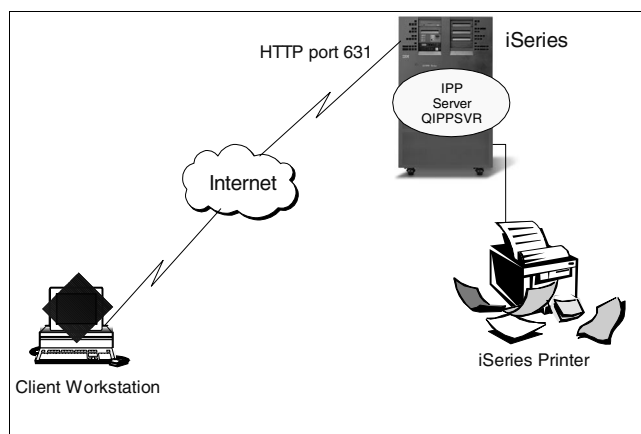


Figure 1. The iSeries as an IPP Server

The IPP client can request and obtain the status of the print job, but this can only be as good as the status OS/400 itself can obtain about the print job. Examples of such requests could include:

- Inquiry about the capabilities of an iSeries printer
- Submission of a print job to this printer, with modified job characteristics such as number of copies, if required
- Inquiry about the status of the print job
- Management of the print job to completion

This is therefore a “Stage 1” implementation of IPP support on the iSeries (formally, the support is IPP version 1.0). At the present time you cannot print from the iSeries (using it as an IPP client) via the Internet to an IPP-enabled printer. However note that IPP clients such as Microsoft Windows 98 and 2000, and IBM Infoprint Manager for Windows NT / 2000 and AIX could be used. IBM also markets an increasing number of IPP-enabled printers such as the Infoprint 21 and Infoprint 70.

#### **1.1.3.1 iSeries IPP printer operations**

OS/400 uses standard iSeries printers, output queues and spool files for IPP printing. Of these, the iSeries output queues are central to IPP printer operations. Examples of IPP printer operations required by the IPP standard include:

- Print-job
- Validate-job
- Get-printer-attributes
- Get-jobs
- Cancel-job
- Get job-attributes

The following optional IPP operations are also supported:

- Pause-printer
- Resume-printer
- Purge-jobs
- Hold-job
- Release-job
- Restart-job

What may not be immediately obvious is that the IPP Server on the iSeries allows IPP printer operations to access iSeries spool files that were not created by the IPP Server. In other words, the list of “normal” spool files on that output queue. This raises the possibility of using IPP to manage spool files from an IPP client by retrieving a list of current print jobs from the output queue.

#### **1.1.3.2 iSeries IPP job operations**

IPP job operations are performed on iSeries spool files. Examples of job operations include:

- Cancel-job
- Get-job-attributes
- Hold-job
- Release-job
- Restart-job

### 1.1.3.3 IPP job template attributes

Job template attributes are the characteristics of the submitted IPP print jobs. On the iSeries, the equivalent would be spool file attributes. Support for job template attributes, and whether or not they are included in create requests, is optional in the IPP standard. Generally speaking, OS/400 supports most of these through iSeries spool file attributes. See Table 2, "Correlation between IPP job template attributes and actual iSeries operations" on page 6.

Table 2. Correlation between IPP job template attributes and actual iSeries operations

IPP job template attribute	Description	iSeries IPP-specific description
Job-priority	IPP value of 1-100	The IPP value is mapped into the iSeries spool file "output priority" value of 1-9.
Job-hold-until	IPP possible values of:  no-hold indefinite day-time evening weekend etc.	The IPP values of 'no-hold' and 'indefinite' are mapped to the iSeries spool file "hold spool file" attribute. The IPP values of 'day-time', 'evening', 'weekend', etc. are not supported. iSeries spool files do not support an "allowable print time" attribute, even though iSeries output queues do have such an attribute.
Job-sheets	IPP possible values of:  none standard	The IPP values of 'none' and 'standard' are mapped to the iSeries spool file "file separator" attribute if the iSeries output queue does not specify job separators. If the iSeries output queue specifies job separators, only the IPP value of 'standard' is supported.
Copies	IPP integer value	The IPP value is mapped into the iSeries spool file "copies" attribute. The iSeries has a limit of 255 copies.
Finishings	IPP possible values of:  none staple saddle-stitch edge-stitch staple-top-left etc.	The IPP value is mapped into the iSeries spool file "corner staple", "edge stitch", and "saddle-stitch" attributes. The physical device must support these operations. Supported by spool files of type AFPDS.

IPP job template attribute	Description	iSeries IPP-specific description
Page-ranges	IPP supports single or multiple page ranges such as:  1-3 5-7,9-11	iSeries spooled files only support one page range. The first IPP page range will be used, any additional pages ranges will be returned to the client as unsupported attributes. Supported by spool files of type AFPDS, SCS, LINE.
Sides	IPP possible values of:  one-sided two-sided-long-edge two-sided-short-edge	The IPP value is mapped into the iSeries spool file "print on both sides" (duplex) attribute. The physical device must support these operations. Supported by spool files of type AFPDS, SCS, LINE.
Number-up	IPP possible values of:  1 2 4	The IPP value is mapped into the iSeries spool file "multiple up" attribute. The physical device must support these operations. Supported by spool files of type AFPDS, SCS, LINE.
Orientation-requested	IPP possible values of:  portrait landscape reverse-landscape reverse-portrait	The IPP value is mapped into the iSeries spool file "page rotation" attribute. The physical device must support these operations. Supported by spool files of type AFPDS, SCS, LINE.
Media	IPP media size values	The IPP value is mapped into the iSeries spool file "measurement method", "page width" and "page length" attributes.
Media	IPP input-tray values	The IPP value is mapped into the iSeries spool file "source drawer" attribute.
Print-quality	IPP possible values of:  draft normal high	The IPP value is mapped into the iSeries spool file "print quality" attribute. The physical device must support these operations. Supported by spool files of type AFPDS, SCS, LINE.

#### 1.1.4 Setup of IPP Server for iSeries

In outline, the required steps for setting up an IPP Server at OS/400 V5R1 are:

1. Confirm your system meets the IPP Server requirements
2. Start the iSeries HTTP server

3. Access the iSeries Tasks page through your web browser
4. Configure the IPP Server for iSeries
5. Configure an IPP printer(s)
6. Start the iSeries IPP Server
7. Configure an IPP client

#### 1.1.4.1 IPP Server requirements

Although support for IPP is built into OS/400 V5R1 (option 3 of OS/400, 5722-SS1), the following products are also required to configure, manage and administer the service:

- HTTP Server for iSeries (5722-DG1)
- IBM Developer Kit for Java (5722-JV1)

If you require printing over a secure connection you also require SSL support, obtained using the following products:

- Digital Certificate Manager (option 24 of OS/400, 5722-SS1)
- Cryptographic Access provider (5722-AC2 or AC3)

#### 1.1.4.2 Starting the HTTP server

Use the command:

```
WRKACTJOB SBS (QHTTPSVR)
```

to check the status of the iSeries HTTP server. There should be two or more ADMIN jobs running in subsystem QHTTPSVR. If not, start the server using:

```
STRTCPSVR SERVER (*HTTP) HTTPSVR (*ADMIN)
```

You can also achieve the above using Operations Navigator. Click on **Network - Servers - TCP/IP** and look for the status of a server called **HTTP Administration**.

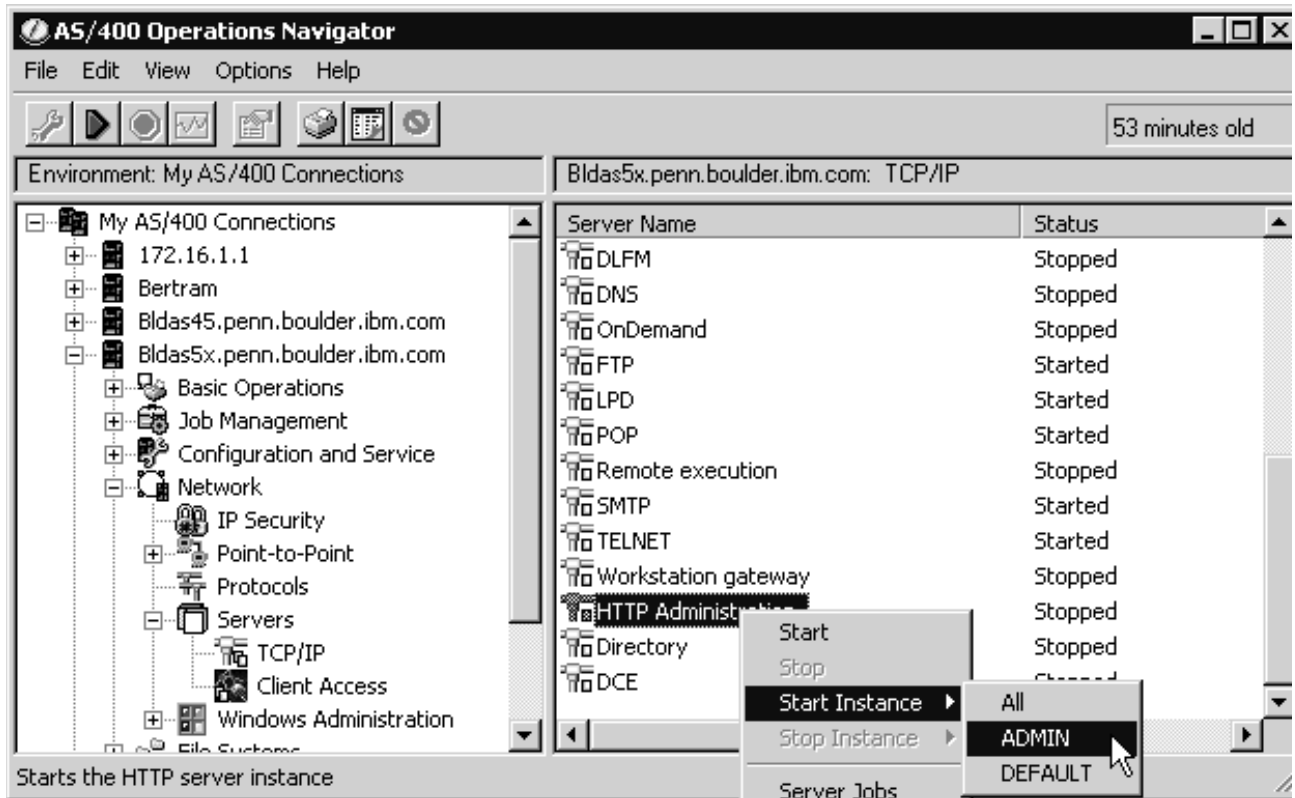


Figure 2. Starting the HTTP server using Operations Navigator

#### 1.1.4.3 Configuring the IPP Server

Start your web browser and enter the following URL to access the iSeries Tasks page:

`http://your_system_name:2001`

Your web browser must support frames and JavaScript, and your iSeries userid must have special authority \*IOSYSCFG. You also need read/write authority to the following files:

- /QIBM/UserData/OS400/lpp/conf/qippsvr-cust.conf
- /QIBM/UserData/OS400/lpp/conf/printer.properties

You will be prompted for your iSeries userid and password.



Figure 3. iSeries Tasks web page

Select the **IBM IPP Server** link. This starts the IPP Server Administrator. Next, click on the **Configuration** link.

The **Basic configuration** menu option provides a page to enable secure sockets. See Figure 4, “Configuring the SSL connection in IPP Server” on page 11. If you want your print data transmitted over a secure connection, you should enable SSL connections. You must select a unique port number on your system: 6310 is suggested. If you need to check for possible port conflicts, use the iSeries command

```
NETSTAT OPTION (*CNN)
```

or use Operations Navigator to navigate to **Network - TCP/IP Configuration - Connections**.

When SSL connections are enabled the IPP Server Administrator registers the IPP Server with the Digital Certificate Manager. The application name registered is QIBM\_IPP\_QIPPSVR. Using the Digital Certificate Manager, you should associate a system digital certificate with the QIBM\_IPP\_QIPPSVR application. For non-SSL data, the IPP Server will use port 631.



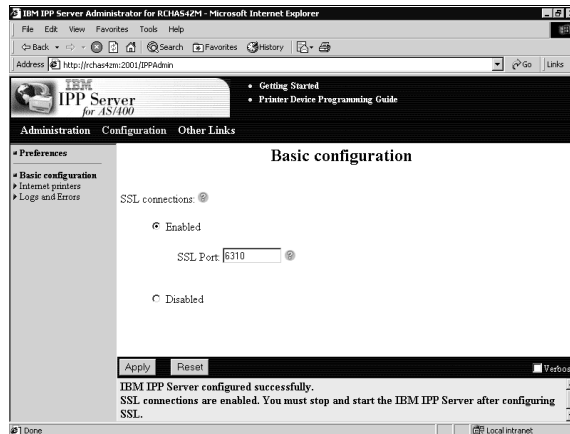


Figure 4. Configuring the SSL connection in IPP Server

Make sure that you have configured your SSL connection, if you want to use this, before you move on to the next step, the creation of the IPP printer descriptions. Also note that you must stop and restart the IPP Server again if you make any server configuration changes, including any IPP printer configuration (create/delete) changes.

#### 1.1.4.4 Configuring an IPP printer

From the IPP Server web page, click on **Configuration - Basic Configuration**. To create an Internet printer configuration you can use either an iSeries printer or an iSeries output queue. The latter is provided for use with remote writers, for remote output queues that do not normally have an associated printer device description. It will probably be more logical to use a print device as the basis for the new IPP device, in which case an output queue of the same will be used by the IPP Server. Whichever you use, the objects must already exist on the iSeries server.

Continue with the “wizard” steps. We do not recommend you associate a specific printer file with the IPP printer, at least not during an initial setup. If you opted to use SSL you can now select it for use with this particular printer. When prompted for an authentication method, select Basic.

You will reach a page asking for the type of data to be transmitted. This will largely depend on the target printer you have chosen:

- Select **AFPDS** if the printer is configured as \*IPDS, AFP=\*YES in its device description.
- Select **SCS** if the printer is configured as a native twinaxial printer (typically 3812 or an impact printer emulation).
- If you will be sending PCL or Postscript data from your IPP client and the printer is capable of printing these data streams (for example a LAN-attached ASCII laser printer), select **Other**. This will cause the spool file data to be sent as \*USERASCII and the iSeries will not convert or modify it. An exception to the latter involves use of the Infoprint Server product. One of the functions enabled by this product is the automatic transformation of ASCII data into that of the target printer (AFP or PostScript). This is described in [cross-ref to the Infoprint Server chapter](#).

The last few screens concern the naming of the new IPP device. You can use the same name as the iSeries printer, or a different one. When the wizard finishes, the URL of the new IPP printer is displayed in the browser message frame (at the foot of the page). For example:

```
http://your_system_name:631/ipp/printers/your_ipp_printer_configuration_name
```

This is the URL by which your IPP client will access (print to) the IPP printer.

#### 1.1.4.5 Starting the IPP Server

Use the Administration page to start the IPP Server - simply click on the “Start” button.

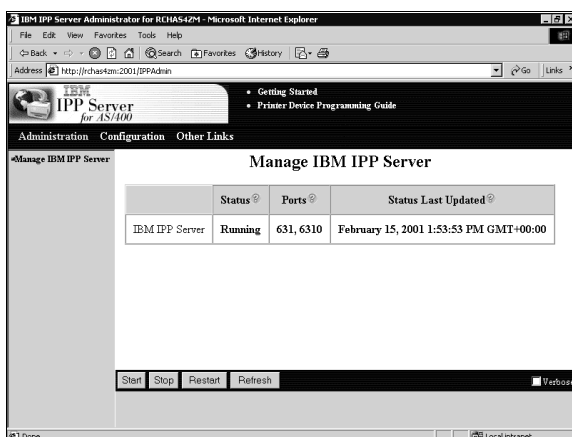


Figure 5. Starting the IPP Server

If you still have a WRKACTJOB screen (or Operations Navigator) displayed, you will now see two IPP Server jobs (QIPPSVR) running in QHTTSPVR. The fully-qualified job name is also displayed in the browser message frame.

#### 1.1.4.6 Setup of an IPP client

With the IPP Server running and a known IPP printer URL created, you can now configure an IPP client. For example, in Windows 2000, select **Start - Settings - Printers - Add Printer - Network Printer - Connect to a printer on the Internet or on your intranet**. Enter the URL of your IPP printer configuration. You may be prompted for a userid and password, and for a security certificate, depending on the level of security you chose earlier. If you receive a message about an “unknown” printer select OK.

When the Add Printer wizard completes, you can use this printer to print to your iSeries printer using IPP. Remember that you need to use the appropriate printer driver for the target iSeries printer - the iSeries will not be performing any data stream transformations to your output.

---

## 1.2 DDS and printer file enhancements

The functions available through OS/400 printer files have been expanded at V5R1.

### 1.2.1 CVTLINDTA

At OS/400 V5R1 there is a single printer file parameter change with the introduction of the CVTLINDTA facility. CVTLINDTA (Convert line data) will cause the output from a printer file specifying \*LINE data and a page definition to be converted to AFP before the data is spooled. The result is an AFPDS formatted spool file.

The reason for including this option is because \*LINE data spool files:

- cannot be viewed with the AFP Viewer within Operations Navigator
- cannot be sent to an ASCII printer using Host Print Transform

An AFPDS spool file overcomes these limitations.

```
Change Printer File (CHGPRTF)

Type choices, press Enter.

Back side overlay:
  Overlay . . . . . *FRONTOVL      Name, *FRONTOVL, *NONE, *SAME
  Library . . . . .          Name, *LIBL, *CURLIB
  Offset down . . . . .          0-57.790, *SAME
  Offset across . . . . .          0-57.790, *SAME
  Constant back . . . . .          *NOCONSTANT, *CONSTANT, *SAME
  Convert line data . . . . . *NO      *NO, *YES, *SAME
  IPDS pass through . . . . . *DEV D   *DEV D, *NO, *YES, *SAME
  User resource library list . . . *DEV D   Name, *DEV D, *NONE...
      + for more values
  Corner staple . . . . . *NONE      *NONE, *BOTRIGHT...
```

Note that there is significant difference between the CVTLINDTA printer file parameter (provided free of charge with OS/400) and the CRTAFPDTA CL command (provided with the priced Infoprint Server for iSeries product). The table below illustrates these differences:

Table 3. Comparison of CVTLINDTA and CRTAFPDTA

CVTLINDTA	CRTAFPDTA
Printer file parameter	CL command
Acts on data to be spooled	Acts on existing spooled file
Creates an AFPDS spooled file, on the OS/400 spool	Creates up to four physical file members: AFP output, indexing, AFP resources and a merged AFP file, in the iSeries integrated file system (IFS)
AFPDS spool file may contain references to unresolved resources	Merged AFP file may contain resolved AFP resources and/or AFP indexes
Included with OS/400 at no charge	Included with the priced program product Infoprint Server for iSeries

You would typically use CVTLINDTA for use with the AFP Viewer, perhaps while developing an AFP application. You can also use it to produce an AFP spooled file for printing to an ASCII printer (Host Print Transform cannot convert \*LINE data printer files).

With the advent of Infoprint Designer for iSeries, it is likely that increasing numbers of customers will be generating line data in order to use the AFP resources created by the Designer product.

## 1.2.2 DDS enhancements

DDS (Data Description Specifications) can be used to create more advanced printer files. Only an overview of the enhancements for OS/400 V5R1 is described here. For more detailed information, including syntax, refer to the V5R1 DDS Reference available online at the iSeries Information Centre at

<http://publib.boulder.ibm.com/pubs/html/as400/infocenter.htm>

### 1.2.2.1 BARCODE

A total of twenty barcode symbologies and variations are now supported using this DDS keyword. They are listed below, together with their descriptive names:

<b>MSI</b>	MSI Data Corporation (Modified Plessey)
<b>UPC-A</b>	Universal Product Code-A
<b>UPC-E</b>	Universal Product Code-E
<b>UPC-2</b>	Universal Product Code 2-digit supplement
<b>UPC-5</b>	Universal Product Code 5-digit supplement
<b>EAN-8</b>	European Article Numbering-8
<b>EAN-13</b>	European Article Numbering-13
<b>EAN-2</b>	European Article Numbering 2-digit supplement
<b>EAN-5</b>	European Article Numbering 5-digit supplement
<b>INDUST2OF5</b>	Industrial 2 of 5 (Standard 2 of 5)
<b>MATRIX2OF5</b>	Matrix 2 of 5
<b>INTERL2OF5</b>	Interleaved 2 of 5
<b>CODEABAR</b>	Codabar
<b>CODE128</b>	Code 128
<b>CODE3OF9</b>	Code 3 of 9 (Code 39, USD-3)
<b>POSTNET</b>	US Postal Service Postnet (Postal Numeric Encoding Technique)
<b>RM4SCC</b>	Royal Mail 4 State Customer Code (UK)
<b>AP4SCC</b>	Australian Postal 4 State Customer Code
<b>DUTCHKIX</b>	Dutch 4 State Customer Code (Klantindex)
<b>JPBC</b>	Japan Postal Bar Code

Barcode height may now be specified for most of these barcode symbologies, expressed either in print lines (1 to 9) or in centimeters or inches. The UOM parameter in the printer file determines which of the latter to use.

For V5R1, the performance of multiple barcodes printed on the same page has been improved.

### 1.2.2.2 BOX

Color support for the box lines has been added, using one of the five color models (see the description for the COLOR keyword below). Shaded boxes may also be created.

### 1.2.2.3 CCSID

CCSID stands for Coded Character Set Identifier and is used to "tag" data with information about its character encoding, character set and code page, character size and any special coding requirements. A value of 65535 indicates that no conversion is done to the data; this is usually the default system value setting for

QCCSID. For V5R1, additional CCSID values have been added to support UCS-2 (Unicode) printing. See Section 1.3, “Unicode” on page 16 for more information.

**1.2.2.4 CDEFNT**

CDEFNT invokes an AFP Coded Font (character set/code page combination). For scalable (outline) fonts, a width value may now be specified, giving anamorphic scaling to characters (where the height is different from the width). If the width is specified, the height must also be specified. The point size may also be expressed in fractions of a point. See the figure below for examples.

Figure 6. Various ways of specifying a coded font

```

|...+...1...+...2...+...3...+...4...+...5...+...6...+...
...8
A*
A          R REC1
A          FLD1          8A    10 13CDEFNT(QFNTCPL/X0BRTR)
A*
A          FLD2          10A   11 13CDEFNT(QFNTCPL/X0BRTP
A                                     (*POINTSIZE 10.1))
A          FLD3          10A   20 13CDEFNT(QFNTCPL/X0BRTP
A                                     (*POINTSIZE 5.0 3.0))
    
```

Note that this is only possible for scalable (outline) fonts, they must exist on the iSeries host and the printer must be capable of receiving and printing such downloaded outline fonts.

**1.2.2.5 COLOR**

This field-level keyword creates color output, using one of the following five color models, of which the last four are new at V5R1:

- Color name
- RGB (red/green/blue)
- CMYK (cyan/magenta/yellow/black)
- CIELAB
- Highlight color

The color name model supports eight named colors: black, blue, brown, green, pink, red, turquoise, yellow.

The RGB color model requires three integer values for the red, green and blue values within a range of 0 0 0 (black) and 100 100 100 (white).

The CMYK color model requires four integer values, each expressed as a percentage from 0 to 100.

The CIELAB color model requires three values. The first is the luminance value (0 to 100.00). The second two values specify the chrominance differences, from -127 to 127.

The Highlight color model requires two values. The first is the required color, from 0 to 65535 and is device-dependent. A value of 0 uses the default color of the

printer device. The second value indicates the coverage, as a percentage from 0 to 100.

Typically the Highlight color model would be used with the IBM Infoprint Hi-Lite Color Printer (Model 4005-HCI). It can also be used to create gray scaling on monochrome IPDS printers such as the Infoprint 60, 62, 200, 300 and 4000.

Host Print Transform supports only the original color name model.

#### **1.2.2.6 FNTCHRSET**

This specifies an AFP font character set. The same enhancements for CDEFNT are also available for FNTCHRSET.

#### **1.2.2.7 FONT**

This specifies a printer-resident AFP scalable (outline) font. The same enhancements for CDEFNT and FNTCHRSET are also available for the FONT keyword.

#### **1.2.2.8 IGCCDEFNT**

This specifies an DBCS (double-byte character set) AFP coded font. The same enhancements for CDEFNT are also available for the IGCCDEFNT keyword.

#### **1.2.2.9 LINE**

Support for drawing lines in one of the five color models has been added at V5R1. See the COLOR DDS keyword section for more information.

### **1.2.3 DDS support for Record Format Line Data**

Limited DDS support for printer files that specified a printer device type of \*LINE was added at V4R3. This allowed customers to produce spooled files using many of the DDS functions but also with a standard AFP page definition. Page definitions are now architected to support "record formats" (which are a key part of DDS) where the original line of unformatted data may be selectively split up and reformatted (for example, positioned over several lines, with different fonts and other attributes).

At V5R1, support has been added to DDS such that when compiling a \*LINE printer file, the name of each DDS record format is added to each line data record.

---

## **1.3 Unicode**

Unicode is an encoding scheme which provides a unique number for *every* character that may be stored in a system, no matter what the language. It is defined by the Unicode Consortium, a public body which publishes the Unicode Standard. It is also known as UCS-2, UTF8, UTF16, ISO/IEC 10646 or CCSID 1208, depending on the context.

By way of comparison, it may be noted that familiar encoding schemes such as ASCII encode some 256 characters whereas UTF-8 covers in excess of 60,000 characters. The lower range of UTF-8 characters coincide with those of ASCII single-byte characters. UTF-16 is used to provide a function approximately equivalent to DBCS (double-byte character set).

Support for Unicode has been adopted by several operating systems, including Sun Solaris, Microsoft Windows NT and 2000, and OS/400. It is now further developed in OS/400 V5R1.

### 1.3.1 OS/400 implementation of Unicode

Unicode actually provides standards for various other character representations such as line layout and character shape, but currently the only support provided by OS/400 is for Unicode code points that point to font glyphs (an absolute-specific image of a character, for example the lower-case character “a”, in raster 300-pel format in the Helvetica roman medium character set at 12 point). This interim support for Unicode is actually the UCS-2 subset of Unicode.

At OS/400 V5R1, the implementation of Unicode-encoded data is achieved by one of two methods:

1. Unicode data can be mapped to selected single- or double-byte EBCDIC encodings before the data is placed on the spool
2. The AFP Unicode Font RPQ program (5799-GJJ). This RPQ (Request for Price Quotation: a less formal program product offering) provides a set of AFP Unicode font equivalent to those in the AFP Font Collection (5648-B45). The application must select a font from this program for any Unicode-defined fields in the printer file.

Unicode-encoded data in \*SCS files will always be converted to EBCDIC before spooling but for files of type \*AFPDS, \*LINE and \*AFPDSLIN, the data may be converted in this way or passed directly into the spool file. The implementation is only available with DDS printer files (using the CCSID keyword).

### 1.3.2 Examples of use

Use this file-, record-, or field-level keyword to specify that a G-type field supports UCS-2 level 1 data instead of DBCS-graphical data. Each UCS-2 character is two bytes long.

The format of the keyword is:

```
CCSID(UCS2-CCSID | &UCS-2-CCSID-field | *REFC
      [*CONVERT | *NOCONVERT]
      [alternate-field-length])
```

The UCS-2-CCSID parameter is required. Use the UCS-2-CCSID parameter to specify a CCSID that uses the UCS-2 Level 1 encoding scheme for this field. You can specify the UCS-2-CCSID parameter either as a number up to 5 digits long or as a program-to-system field. You must define the program-to-system field with a length of 5 and with the S data type.

The \*CONVERT parameter is optional and specifies whether the UCS-2 data is converted to a target CCSID specified on the CHRID parameter of the CRTPRTF, CHGPRTF, or OVRPRTF commands. \*CONVERT is the default. If you specify the CCSID keyword with \*NOCONVERT, the UCS-2 data is not converted to the target CCSID.

If \*NOCONVERT is active for a printer file whose DEVTYPE is \*AFPDS, \*LINE, or \*AFPDSLIN, the application must also use one of the AFP Unicode migration

fonts. If you do not specify an AFP Unicode migration font, the output will be interpreted as single-byte data and will probably be unprintable.

If \*NOCONVERT is active and the file DEVTYPE is \*AFPDS, specify an AFP Unicode migration font character set and code page with the FNTCHRSET keyword for printer files. If the file DEVTYPE is \*LINE or \*AFPDSLIN, specify the AFP Unicode migration font character set and code page in the page definition for the printer file.

If \*NOCONVERT is specified for a printer file whose DEVTYPE is \*SCS, a diagnostic message is issued when the printer file is used, and the UCS-2 data is converted to the target CCSID.

The alternate-field-length parameter is optional and is valid only when you specify the CCSID keyword at the field level and the \*CONVERT parameter is active. Specify the alternate-field-length as the number of UCS-2 characters.

When UCS-2 data is involved in an output operation and the \*CONVERT parameter is active, the data is converted from the associated UCS-2 CCSID to the target CCSID. Generally, the length of the data will change when this conversion occurs. Therefore, you can use the alternate-field-length value to specify a printed field length that is different from the default printed field length. The default printed field length of a 'G' data type field is twice the number of characters that are specified for the field length.

For other syntax rules, refer to the online DDS Reference Guide

---

## 1.4 Java print support

This is mainly covered in a separate chapter ([link to Mira / Ken D's chapter on Java print support here](#)) but the following is a summary of the new Java print classes available with OS/400 V5R1.

- New Java classes for creating formatted documents and reports using XSL (eXtensible Style Language) Formatting Object (FO) technology
- New Java classes for creating AFP (Advanced Function Presentation and Printing) documents

These classes are included in the V5R1IBM Toolbox for Java product. For a good starting point, see:

<http://www.ibm.com/series/toolbox>

### 1.4.1 Java classes for creating documents with XSL FOs

These classes provide a higher level interface for building and printing reports and documents. The output document formats include HP-PCL and Adobe PDF.

### 1.4.2 Java classes for creating documents with AFP

These classes provide a programming interface for writing application data to an OS/400 output queue in the form of record format line data. The application can specify whether the data is converted immediately to AFP when the spool file is created ("pre-spool") or whether it is written to spool as line data. In the latter case, PSF/400 would convert the data to AFP when the spool file is printed



(“post-spool”). AFP page and form definitions (formatting objects) would be used to format the line data into AFP. The creation and use of these objects is covered elsewhere in this publication ([cross-ref to the PPFA chapter](#)).

---

## 1.5 Miscellaneous print enhancements

The following enhancements have been enabled at OS/400 V5R1 through feedback from customers:

- Maximum number of spool files created per job increased from 9,999 to 999,999, through the QMAXSPLF system value.
- The ability to associate a data queue with an output queue is already part of OS/400. In addition to this data queue support, a data queue may be associated with a job, or the system itself, using the environmental variable QIBM\_NOTIFY\_CRTSPLF together with the ADDENVVAR CL command. Spool file attributes such as the fully-qualified job name, spooled file name and number are stored in the data queue when the spool file is created. Such a data queue would be useful when spooled files have been created under a job that has transferred to another user profile. This function is documented in APAR SA87008 for V5R1 and *Printer Device Programming Version 5*, SC41-5713-04. The function is available for releases back to V4R2 via a PTF.
- The Retrieve Writer Information API (QSPRWTRI) now supports remote print writers in addition to “normal” print writers.
- The performance of the Retrieve Spool File List APIs QUSLSPL and QGYOLSPL has been improved.
- New support has been added to allow information about iSeries printers and NetServer print shares to be published into LDAP (Lightweight Directory Access Protocol) directories. This allows the user to write applications that query the LDAP directory for iSeries printer information such as a printer’s location or capabilities. It also allows the user to configure iSeries printers directly on their Windows 2000 desktop by using the Add Printer wizard, and specifying that the information necessary to configure the printer be obtained from the Windows 2000 Active Directory.

