

iSeries



Guide to Output

iSeries



Guide to Output

Note

Before using this information and the product it supports, be sure to read the information in "Notices" on page 361.

Sixth Edition (May 2004)

This edition applies to OS/400 Version 5 Release 3 Modification 0, and to all subsequent releases and modifications until otherwise indicated in new editions or technical newsletters. Be sure to use the correct edition for the level of the system.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

The IBM Printing Systems Division welcomes your comments. A form for reader's comments is provided at the back of this publication. If the form has been removed, you may send your comments to the following address:

INFORMATION DEVELOPMENT
THE IBM PRINTING SYSTEMS DIVISION
DEPARTMENT H7FE BUILDING 004M
PO BOX 1900
BOULDER CO 80301-9191
U.S.A.

If you prefer to send comments electronically, use one of the following methods:

- Internet: printpub@us.ibm.com
- Fax: 1-800-524-1519 or 1-303-924-6873

Internet

Visit our home pages at <http://www.ibm.com/printers/r5psc.nsf/web/as400overview>
and <http://www.ibm.com/servers/eserver/series/>.

© Copyright International Business Machines Corporation 1996, 2004. All rights reserved.

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

Contents

Figures	ix
Tables	xiii
About iSeries Guide to Advanced Function Presentation (S544–5319)	xv
Who Should Read This Book	xvi
Conventions and Terminology Used in This Book	xvi
Prerequisite and Related Information	xvii
How to Send Your Comments	xvii
Summary of Changes	xix

Part 1. Introduction to Advanced Function Presentation. 1

Chapter 1. iSeries Output	3
Re-engineering Business Documents?	3
iSeries Output Architecture	4
Page Architecture	5
Printer Files and DDS	6
AFP Page Resources Continued	7
Printing	7
E-business output	8
Chapter 2. What is Advanced Function Presentation (AFP)?	11
AFP Data Streams	12
Elements of AFP Output	12
PSF for OS/400	14
Print Flow with PSF	15
Printers and PSF	16
When PSF Is Required	16
Chapter 3. AFP Applications	19
Enabling Applications	19
Complementary Applications	21
Chapter 4. Introduction to Printer Files	25
Printer File Flow	25
Printer File Parameters	26
Using Printer Files to Enhance Your Output	27
Specifying Line and Font Changes	27
Specifying Multi-Up Printing	29
Specifying Computer Output Reduction	30
Adding Overlays	32
Converting Line Data to AFP	34
Specifying Finishing Operations	36
Chapter 5. Introduction to PSF Configuration Objects	39
Creating and Changing PSF Configuration Objects	39
Creating a PSF Configuration Object for the PDF Subsystem	40
Using a PSF Configuration Object	40
Chapter 6. Introduction to AFP Resources	41
Fonts	41

What is a Font?	41
Font Architecture	45
Where Fonts are Stored	49
How to Find Fonts on Your iSeries.	50
How PSF Finds an AFP Font.	52
Where to Get Fonts	56
Images and Graphics	61
Images and Graphics in AFP Output	62
Image and Graphic Terminology	63
Image File Formats	64
Image Compression	65
Overlays	65
Why Use an Overlay?	66
Options for Building Overlays	66
Bar Codes	67
Bar Code Terminology	68
Bar Code Basics	69
Bar Code Symbolologies	70
Page Definitions and Form Definitions	72
Page Definitions	72
Form Definitions	74
How Page Definitions and Form Definitions are Used.	76
Sources of Page Definitions and Form Definitions	77
Object Containers	78
Chapter 7. Working with Fonts	79
Create a Document with Several Fonts	79
Example	80
Printing a Font Catalog	82
Using Outline Fonts	82
Working with Font Mapping Tables	82
Capturing Fonts	83
Anamorphic Font Character Scaling	84
Steps to Restore Font Libraries From the Font Collection CD to the iSeries	84
Chapter 8. Working with Images	87
Building Images	87
Scanning the Source Image	87
Touching up the Image	87
Converting Images to IOCA	90
Uploading and Compiling the Image	92
Including Images in your OS/400 Output	93
Chapter 9. Working with Overlays	95
Overlay Design Considerations	95
Performance Considerations	95
Fonts	96
Images	96
Object Size	96
Designing Overlays Using Infoprint Designer	97
Designing Overlays Using DDS	97
Creating Overlays from Existing Images.	97
Using Overlays with OS/400 Output	99
Chapter 10. Working with Bar Codes	101
Making a Bar Code Decision	102

Which Symbology?	102
What Density?	102
What Height?	103
What Orientation?	103
What Print Volume?	103
What Print Window?	103
Specifying Bar Codes with PPFA	103
Specifying Bar Codes with DDS	104
Specifying Bar Codes in Infoprint Designer	105
Specifying Bar Codes in Print Format Utility	107
Specifying Bar Codes in APU	108
Chapter 11. Working with Object Containers	109

Part 2. Super Sun Seeds Case Study. 111

Chapter 12. Introduction to the Super Sun Seeds Case Study	113
INVPRE Application Program Example.	114
INVPRE Application Program General Processing Flow	117
DDS Source for INVPRE Example	117
Choosing the Proper Tool	119

Chapter 13. Using Printer Files	121
Additional Considerations	123

Chapter 14. Using Data Description Specifications	125
Printer File DDS	125
Keywords for AFP Applications	126
DDS Functions	128
DDS Example	129
RPG for Example	134
Floating Document Elements	135
Positioning	136
Using DDS for Super Sun Seeds.	137
DDS for New Invoice Output	141
RPG Source for New Invoice Output (INVNEW1)	144
Beginning the Invoice	147
Processing Items	147
Overflow Items (ITMCNT = 35)	147
Ending the Invoice (No Room for Offer)	148
Ending the Invoice	148
Enhancing the Super Sun Seeds Invoice	148

Chapter 15. Using Advanced Print Utility	153
What You Can Do with APU	154
Why Use APU?	155
Data Mapping and Field Selection	155
APU Data Structure.	155
Example of a Single-Page Format Document	157
Super Sun Seeds Case Study	159
Expanded Case Study with APU	171
Printing with APU	175
Manually Associating a Print Definition with a Spooled File	175
Using the Apply Print Definition Command	178
Automatic Printing with APU Monitor	178
Understanding How the APU Monitor Works	179

An Example of APU Monitor Processing	179
Chapter 16. Using Page Printer Formatting Aid	183
Creating Page Definitions and Form Definitions with the PPFA Compiler	183
Creating a Page Definition Using the PPFA Compiler	184
Creating a Form Definition using the PPFA Compiler	187
Using Page Definitions and Form Definitions to Format OS/400 Output.	189
Super Sun Seeds Case Study	191
Record Format and PPFA	194
Using the Record Formatting Functions	194
Defining Color.	195
Drawing Graphics	195
Creating Variable-Length Graphics	196
Chapter 17. Using Infoprint Designer	197
Overview	197
What You Can Do with Infoprint Designer.	198
Overlay Editor.	199
Image Editor	199
Layout Editor	200
Super Sun Seeds Case Study	201
Chapter 18. Using IBM AFP Toolbox	217
What Can I Do with AFP Toolbox?	217
AFP Toolbox Document and Page Structure.	218
Super Sun Seeds Case Study	219
Chapter 19. Using the Print Format Utility	223
Introduction to PFU.	223
Super Sun Seeds Packing List Labels	224
Summary	230

Part 3. Working with AFP Output 231

Chapter 20. Viewing Files	233
Benefits of Viewing Your Output	233
iSeries Access Viewer Details	234
AFP External Resources	234
Overlays and Page Segments	235
Fonts	235
Viewing Spooled Files with iSeries Access	235
Viewing OS/400 Stream Files with iSeries Access	237
Viewing Spooled Files with iSeries Access for Web	238
Viewing AFP Files with the AFP Viewer Plug-In	239
Calling the AFP Viewer Directly from Your Application	239
Chapter 21. Sending Your OS/400 Output by Fax	241
Super Sun Seeds Case Study	241
Chapter 22. Working with Infoprint Server	245
How Can I Use Infoprint Server?	245
Publish Output for Electronic Access	246
Electronic Report Distribution Using Segmentation	247
E-Business Customer Documents	248
Segmenting Customer Documents	249
Consolidating Enterprise Printing	250

Delivering Documents and Reports over Intranet	251
Creating Electronic Documents Interactively	252
Integrating Web Images	253
Transforming Your OS/400 Output to PDF	254
Accessing the PDF Subsystem	255
Intelligent Routing	256
E-Mail Support	256
Indexing Output with the Create AFP Data Command	257
Transforming ASCII Image Data to AFP	259
Printing PCL, PDF, and PostScript Data on IPDS Printers.	259
Chapter 23. Working with Your iSeries from a Windows Client	261
iSeries Access	261
iSeries Navigator.	262
iSeries Access for Web	263
Using iSeries Access for Web to Convert and View a Spooled File in PDF	264
Using iSeries Access	268
Installing AFP Manager	268
Converting a Spooled File to PDF for E-mailing, Spooling, or Storing	269
Creating a PSF Configuration Object	271
Changing a PSF Configuration Object	272
Importing Resources to the OS/400	273
Creating a Printer Share	274
Chapter 24. Indexing, Archiving, and Retrieving Output with Content Manager OnDemand for iSeries.	279
Super Sun Seeds Case Study	279
Defining the Invoices to CM OnDemand	279
Retrieving the Archived Invoices from CM OnDemand	282
Chapter 25. Deciding on an Output Strategy and Set of Tools.	287

Part 4. Appendixes 291

Appendix A. iSeries Font Libraries	293
Appendix B. Additional DDS Examples.	297
INVNEW2 Enhanced Super Sun Seeds Invoicing	297
INVNEW2 RPG Source	298
INVNEW2 DDS Source	303
INVNEW3 Floating Super Sun Seeds Invoicing	305
INVNEW3 RPG Source	307
INVNEW3 DDS Source	312
Appendix C. Source Code and Line Data for Super Sun Seeds	317
Page Definition Source Code	317
Form Definition Source Code	323
Sample Line Data	328
Appendix D. Sample Toolbox Source Code	331
Sample COBOL Program using AFP Toolbox	331
Description of COBOL Program Listing	342
Sample RPG Program using AFP Toolbox	343
Appendix E. Migrating from OfficeVision/400	347

Appendix F. Performing Advanced Font Tasks with iSeries Navigator	349
Viewing and Changing Font Mapping Tables	349
Creating a Font Mapping Table	350
Activating Font Capture	351
Appendix G. Finding OS/400 Objects in the Integrated File System . . .	353
Introducing the Integrated File System	353
Finding an Object in the Integrated File System	354
Moving Data between File Systems	354
How PSF Searches for Resources	357
How PSF Searches for Object Containers	358
Specifying Resource Library Lists	359
Using AFP Manager to Specify Resource Library Lists	359
Notices	361
Programming Interface Information	362
Trademarks.	363
Glossary	365
Bibliography	375
Advanced Function Presentation	375
Business Graphics Utility	375
Data Stream and Object Architectures	375
Facsimile Support for iSeries	376
GDDM	376
Fonts	376
IBM AFP PrintSuite for iSeries	376
IBM Content Manager OnDemand for iSeries	377
Infoprint Designer for iSeries	377
Infoprint Server for iSeries	377
OS/400	377
Redbooks	377
Index	379

Figures

1. iSeries Print Flow	4
2. AFP Page Architecture	5
3. Sample of AFP Output Using PSF	13
4. PSF Print Flow	15
5. Override with Printer File (OVRPRTF) Command	26
6. Change Spooled File Attributes (CHGSPLFA) Command	26
7. Override with Printer File Display - First Screen	28
8. Override with Printer File Display - Second Screen	28
9. Override with Printer File Display - Third Screen	29
10. Multi-Up Printing: Example of 2-up and 4-up Printing	30
11. Override with Printer File Display - Pages Per Side	30
12. Example of COR Function	31
13. Override with Printer File Display - Reduce Output	32
14. Greenbar Overlay	33
15. Override with Printer File Display - Overlay Parameters	34
16. Override with Printer File Display - Convert Line Data	35
17. Override with Printer File Display - Finishing Parameters	37
18. Create PSF Configuration Screen.	39
19. Fonts on a Sample Page	42
20. Raster Font Characters with Different Resolutions	43
21. Translation of a Keyboard Character into a Printed Character	47
22. How Code Pages Work	48
23. Image and Graphic on the Sample Page	62
24. Overlay on the Sample Page	66
25. Bar Codes on the Sample Page	68
26. Linear Bar Code Elements	70
27. UPC Bar Code Example	71
28. Postnet Bar Code Example	71
29. PDF417 Bar Code Example	71
30. Traditional Printing Compared to Printing with Page Definitions and Form Definitions	72
31. How a Page Definition is Used.	73
32. Page Definition Functions	74
33. How a Form Definition is Used.	75
34. Form Definition Functions	76
35. Document with Several Fonts	81
36. DDS Keywords Used to Create Sample Document	82
37. Image Editor	88
38. Super Sun Seeds Logo Scanned into the Infoprint Designer Image Editor	88
39. Zooming in on the Super Sun Seeds Logo Image	89
40. Terms and Conditions Overlay Created in a Windows Application	98
41. Bar Codes Printed on an IPDS Printer	102
42. The Bar Code Tool Button	105
43. Defining the Bar Code	106
44. Specifying the Bar Code Properties	106
45. Change Bar Code Detail Display	107
46. Change Bar Code Detail Display	107
47. Defining a Bar Code in APU	108
48. Defining a Bar Code in APU	108
49. Super Sun Seeds Preprinted Invoice	113
50. Create Printer File - Adding an Overlay	122
51. Externally Described Printer File with DDS	126
52. DDS Functionality	129
53. Super Sun Seeds Invoice as Floating Document.	136

54. Grid Overlay	137
55. New One Page DDS Version of Super Sun Seeds Invoice	138
56. Page One of Two-Page DDS Version of Super Sun Seeds Invoice	139
57. Page Two of Two-Page DDS Version of Super Sun Seeds Invoice	140
58. Electronic Overlays on New DDS Super Sun Seeds Invoice	141
59. Customer Copy - INVNEW2	150
60. Super Sun Seeds Packing List	151
61. Typical APU Application	153
62. APU Print Flow	154
63. APU Data Structure	156
64. APU Step by Step	157
65. Super Sun Seeds Invoice	158
66. APU Main Display	159
67. Work with Print Definitions Display	160
68. Print Definition Attributes (First Display)	160
69. Print Definition Attributes (Second Display)	161
70. Work with Print Definitions Display	161
71. Work with Copies Display	161
72. Define a Copy Display	162
73. Select a Sample Spooled File Display	162
74. Set Page Layout Options Display	163
75. Back Overlay (Terms and Conditions) - INVBAC	164
76. Define Field Mapping Display	164
77. Define Field Mapping Display	165
78. Map Bar Code Display	165
79. Define Constants Display	166
80. Create Constant Text Display	166
81. Select a Font Display	167
82. Create Constant Text Display	167
83. Define Constants Display	168
84. Define Overlays Display	168
85. Sample INVALL Overlay	169
86. Work with Copies Display	169
87. Work with Copies Display	170
88. Define Price Suppression for Packing List	170
89. Define Price Suppression for Packing List	171
90. Specifying Field and Rules Selection	172
91. Defining Selection Fields	172
92. Selection Field Pop-up	173
93. Selection Field Naming	173
94. Define Selection Rules for Page 1	174
95. Define Selection Rules for All Other Pages	174
96. Summary of Selection Rules	175
97. Select a Spooled File Panel	176
98. Apply Print Definition Panel	176
99. Apply Print Definition Panel	177
100. Apply Print Definition Panel	178
101. APU Monitor Processing Sequence	179
102. APU Monitor Before Processing	180
103. APU Monitor - Action Example	181
104. APU Monitor Example After Processing	182
105. Creating Page Definitions and Form Definitions Using PPFA	184
106. Convert PPFA Source Display	185
107. Create Page Definition (CRTPAGDFN) Display	186
108. Page Definition Source Code	187
109. Convert PPFA Source (CVTPPFASRC) Display	188

110. Create Form Definition (CRTFORMMDF) Display	189
111. Form Definition Source Code	189
112. Formatted Output - Customer 1	190
113. Formatted Output - Customer 2	191
114. Three Copies of a Single-Page Customer Invoice	192
115. Three Copies of a Two-Page Customer Invoice	193
116. Variable-Length Boxes	196
117. Infoprint Designer Application Flow	198
118. Overlay Editor	199
119. Super Sun Seeds Logo Scanned into the Image Editor	200
120. Layout Editor Window	201
121. The SCS File to be Formatted	202
122. Super Sun Seeds Invoice	203
123. New Project	204
124. Box Properties Dialog - Rounding Page	205
125. Select Font Dialog	206
126. Selecting the Spooled File	207
127. Layout Properties Dialog	208
128. Spooled File Displayed in the Data Window	209
129. Positioning Text in the View/Edit Window	210
130. Bar Code Options Dialog	211
131. Finished Overlay with Data	212
132. Upload to AS/400 Dialog, AS/400 Connection Page	213
133. Override with Printer File - First Screen	214
134. Override with Printer File - Specifying the Page Definition and Form Definition.	214
135. AFP Toolbox Document and Page Structure	218
136. Packing List Label Printout.	224
137. AFP Utilities Main Menu.	226
138. Work with PFD Definitions Display	226
139. Change PFD Definition Display	227
140. Specify Database File Display	227
141. Design Record Layout Display	228
142. Change Text Detail Display	228
143. Design Record Layout Display	229
144. Design Page Layout Display	229
145. Change Record Layout Detail Display.	230
146. Viewing OS/400 Output Using iSeries Access	235
147. The iSeries Access Window	236
148. iSeries Navigator Window	236
149. Spooled File Displayed	237
150. File Displayed in Acrobat Reader	238
151. Outbound Fax Commands	241
152. Work with Fax Output Queue (WRKFAXQ)	242
153. Work with Fax Output Queue	242
154. Send Fax Display	242
155. Send Fax Display (continued).	243
156. Sample Fax Overlay (SSSFAX)	243
157. Work with Fax Output Queue	244
158. Display Spooled File	244
159. Publishing iSeries Output for Electronic Access	246
160. Electronic Report Distribution with Segmentation.	247
161. E-Business Customer Documents	248
162. Segmenting Customer Documents	249
163. Consolidating Enterprise Printing	250
164. Delivering Documents and Reports over Intranet.	251
165. Creating Electronic Documents Interactively	252

166. Integrating Web Images	253
167. Transforming a Spooled File to PDF	255
168. How the Infoprint Server Create AFP Data Command Works	258
169. iSeries Access Components	261
170. iSeries Navigator Window	262
171. AFP Manager	263
172. iSeries Access for Web Home Page	264
173. iSeries Access for Web Home Page	265
174. PDF Output Settings Page	265
175. File Displayed in Acrobat Reader	266
176. Personal Folder Page	267
177. Settings for Mail an Attachment Option	267
178. iSeries Navigator Window – Install Additional Components	268
179. Selective Setup Window.	269
180. Component Selection Window	269
181. Selecting a Spooled File to Convert to PDF	270
182. Convert Printer Output to PDF Window	271
183. New PSF Configuration Window.	272
184. Configure PSF Configuration Window.	273
185. Import Page Segment Window	274
186. iSeries Access Components	275
187. iSeries NetServer Print Share Dialog	276
188. Starting the CM OnDemand Report Wizard.	280
189. Adding a Trigger	281
190. Defining an Index Field	282
191. List of Folders	283
192. The Search Window	284
193. List of Invoices that Meet Search Criteria	284
194. Viewing an Invoice.	285
195. INVNEW2 Sample Invoice	297
196. INVNEW3 Sample Invoice	306
197. Invoice Heading Overlay for Page 1	306
198. Super Sun Seeds Case Study Created with AFP Toolbox	331
199. Open Font Mapping Table Window	349
200. New Font Mapping Table Window	350
201. Import Code Page Window	351
202. Enabling Font Capture	352
203. iSeries Navigator File Systems	355
204. How PSF Searches for Resources	358

Tables

1.	AFP Page Elements and Respective Sub-architecture	6
2.	OS/400 Printer Writers	8
3.	Printer File Parameters that Require PSF	17
4.	DDS Keywords that Require PSF	17
5.	Code Point Translations	47
6.	Commonly Used Font Collection Font Libraries.	49
7.	IBM Infoprint 2085 Fonts (Partial List)	56
8.	DBCS Font Families Available	58
9.	Font Products Simulated by DBCS Simulation Fonts.	59
10.	Transformers, Utilities, and Tools	59
11.	Image and Graphic Terminology	63
12.	Commonly Used Image File Formats	64
13.	Bar Code Terminology	68
14.	DDS Keywords for AFP Applications	126
15.	Recommended Approaches — by Task	287
16.	Comparing iSeries Formatting Approaches	289
17.	Best Application of Each Tool	290

About iSeries Guide to Advanced Function Presentation (S544–5319)

This guide is based on the standard iSeries course on printing. It is designed to be used as an educational guide as opposed to a reference book. The guide is divided into three parts:

Part I introduces you to the iSeries output architecture. This includes how documents are composed and stored on iSeries as well as how they are printed or delivered electronically. Part I breaks down the iSeries document into logical components - fonts, overlays, images, and barcodes, describing each in detail.

Part II looks at the techniques and applications used to create documents on the iSeries. One typical document application, an invoice, is used as a case study. In this part, we examine how each technique or application would be applied to create the invoice.

Part III turns our attention to document delivery. Once our invoice has been generated and is sitting in an iSeries output queue, what options are available to deliver the document to the consumer of that document?

Part 1, "Introduction to Advanced Function Presentation," on page 1 includes these chapters:

- Chapter 1, "iSeries Output," on page 3
- Chapter 2, "What is Advanced Function Presentation (AFP)?," on page 11
- Chapter 3, "AFP Applications," on page 19
- Chapter 4, "Introduction to Printer Files," on page 25
- Chapter 5, "Introduction to PSF Configuration Objects," on page 39
- Chapter 6, "Introduction to AFP Resources," on page 41
- Chapter 7, "Working with Fonts," on page 79
- Chapter 8, "Working with Images," on page 87
- Chapter 9, "Working with Overlays," on page 95
- Chapter 10, "Working with Bar Codes," on page 101

Part 2, "Super Sun Seeds Case Study," on page 111 contains these chapters:

- Chapter 12, "Introduction to the Super Sun Seeds Case Study," on page 113
- Chapter 13, "Using Printer Files," on page 121
- Chapter 14, "Using Data Description Specifications," on page 125
- Chapter 17, "Using Infoprint Designer," on page 197
- Chapter 15, "Using Advanced Print Utility," on page 153
- Chapter 16, "Using Page Printer Formatting Aid," on page 183
- Chapter 18, "Using IBM AFP Toolbox," on page 217
- Chapter 19, "Using the Print Format Utility," on page 223

Part 3, "Working with AFP Output," on page 231 includes these chapters:

- Chapter 20, "Viewing Files," on page 233
- Chapter 21, "Sending Your OS/400 Output by Fax," on page 241
- Chapter 22, "Working with Infoprint Server," on page 245
- Chapter 23, "Working with Your iSeries from a Windows Client," on page 261

- Chapter 24, "Indexing, Archiving, and Retrieving Output with Content Manager OnDemand for iSeries," on page 279
- Chapter 25, "Deciding on an Output Strategy and Set of Tools," on page 287

The book also contains these appendices:

- Appendix A, "iSeries Font Libraries," on page 293
- Appendix B, "Additional DDS Examples," on page 297
- Appendix C, "Source Code and Line Data for Super Sun Seeds," on page 317
- Appendix D, "Sample Toolbox Source Code," on page 331
- Appendix E, "Migrating from OfficeVision/400," on page 347
- Appendix F, "Performing Advanced Font Tasks with iSeries Navigator," on page 349
- "How PSF Searches for Resources" on page 357

Other sections are:

Glossary

"Bibliography" on page 375

Who Should Read This Book

The information in this publication is directed at people in applications and operational functions that want to learn how to use iSeries™ tools to create and distribute electronic documents.

This publication is written with the assumption that you have experience with application programming. It is also assumed that you are familiar with the system and using it to print.

Conventions and Terminology Used in This Book

In this manual, the term "AFP™ Viewer" refers to both the AFP Workbench Viewer, available with iSeries Access, and the AFP Viewer Plug-in, available as a free download from IBM®. In this publication, the term "iSeries Access" refers to both Client Access and iSeries Access. The term "printer" refers to both physical and virtual printers. Virtual printers have no hardware and return electronic output instead of paper output. For example, to transform certain types of data you send the spooled file to a virtual printer. To the OS/400®, the transform looks like a printer. Print Services Facility™ for OS/400 is referred to as PSF throughout this publication.

These rules apply to OS/400 command syntax throughout this publication:

- Variable data is printed in italics. Enter specific data to replace the characters in italics. For example, for *indexing-page-limit* you could enter 50. Italics also identify the names of publications.
- Commands are printed in all upper case letters. They must be entered exactly as they appear.
- Values are enclosed in parenthesis, for example, (*file*). Enter the parenthesis as part of the parameter.

Monospacing, for example, `this is monospaced`, is used for examples and Web addresses.

Prerequisite and Related Information

Use the iSeries Information Center as your starting point for looking up iSeries technical information.

You can access the Information Center two ways:

- From the following Web site:
<http://www.ibm.com/eserver/iseries/infocenter>
- From the *iSeries Information Center*, SK3T-4091-04 CD-ROM. This CD-ROM ships with your new iSeries hardware or IBM Operating System/400 software upgrade order. You can also order the CD-ROM from the IBM Publications Center:
<http://www.ibm.com/shop/publications/order>

The iSeries Information Center contains new and updated iSeries information such as software and hardware installation, Linux, WebSphere, Java, high availability, database, logical partitions, CL commands, and system application programming interfaces (APIs). In addition, it provides advisors and finders to assist in planning, troubleshooting, and configuring your iSeries hardware and software.

With every new hardware order, you receive the *iSeries Setup and Operations CD-ROM*, SK3T-4098-02. This CD-ROM contains IBM @server iSeries Access for Windows and the EZ-Setup wizard. iSeries Access Family offers a powerful set of client and server capabilities for connecting PCs to iSeries servers. The EZ-Setup wizard automates many of the iSeries setup tasks.

For other related information, see the “Bibliography” on page 375.

How to Send Your Comments

Your feedback is important in helping to provide the most accurate and high-quality information. If you have any comments about this book or any other iSeries documentation, fill out the readers’ comment form at the back of this book.

- If you prefer to send comments by mail, use the readers’ comment form with the address that is printed on the back. If you are mailing a readers’ comment form from a country other than the United States, you can give the form to the local IBM branch office or IBM representative for postage-paid mailing. If the form has been removed, you can send comments to this address:

INFORMATION DEVELOPMENT
THE IBM PRINTING SYSTEMS DIVISION
DEPARTMENT H7FE BUILDING 004M
PO BOX 1900
BOULDER CO 80301-9191
U.S.A.

- If you prefer to send comments by FAX, use either of the following numbers:
 - United States, Canada, and Puerto Rico: 1-800-524-1519
 - Other countries: 1-303-924-6873
- If you prefer to send comments electronically, use one of these e-mail addresses:
 - Comments about books:
printpub@us.ibm.com
 - Comments about the iSeries Information Center:
RCHINFOC@us.ibm.com

Be sure to include this information:

- The name of the book or iSeries Information Center topic.
- The publication number of a book.
- The page number or topic of a book to which your comment applies.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Summary of Changes

Summary of Changes for *iSeries Guide to Output*, S544-5319-05.

This publication contains additions and changes to information that was previously presented in *iSeries Guide to Output*, S544-5319-04, which supports OS/400 Version 5 Release 2. The technical additions and changes are marked with a revision bar (|) in the left margin.

These general changes have been made throughout the book:

- OS/400 V5R1 is no longer supported. References to it have been removed.
- The PCL, PostScript, and PDF to AFP Infoprint Server transforms are now a priced feature.
- OnDemand for iSeries is now called Content Manager OnDemand for iSeries.

The following information is new or updated:

- Infoprint Fonts for Multiplatforms is described in “Enabling Applications” on page 19.
- A description of some third party design tools has been added to “Complementary Applications” on page 21.
- Chapter 5, “Introduction to PSF Configuration Objects,” on page 39 has been updated to incorporate changes to the PSF configuration object.
- “Fonts” on page 41 has been reorganized and expanded.
- “Object Containers” on page 78 has been added.
- “Create a Document with Several Fonts” on page 79 has been added. It gives an example of how to specify several different fonts using DDS.
- Chapter 11, “Working with Object Containers,” on page 109 describes how to include object containers in your data.
- Chapter 22, “Working with Infoprint Server,” on page 245 has been updated to include information about intelligent routing.
- “Intelligent Routing” on page 256 has been added to describe new Infoprint Server functionality.
- New terms have been added to the Glossary.

Part 1. Introduction to Advanced Function Presentation

Part I introduces you to the iSeries output architecture. The iSeries output architecture is for composing and storing documents as well as printing or delivering the information electronically. Part I breaks down the iSeries document into logical components - fonts, overlays, images, and bar codes, describing element in detail.

Chapter 1. iSeries Output

iSeries output is the subject of this guide, with the focus on the options available to generate the kinds of documents required to compete in today's business environment. This is an environment where traditional standards of document content and document delivery might not be adequate. Some changes to document content, such as bar coding, have moved printed pages into the overall business process workflow. The Internet and intranets have driven e-business process reengineering where hardcopy printed pages may no longer suffice. These changes generally make an electronic document essential to most business applications and electronic delivery critical to many.

Re-engineering Business Documents?

In all likelihood, you have turned to this book to learn how new business documents could be developed or existing documents re-engineered in order to meet one or more new requirements within your organization. Those requirements might include:

Migrating preprinted forms: The costs of preprinted forms, including material costs and operational inefficiencies, has become too high.

Requirements from customers or suppliers: A very common example is the need to generate bar codes on documents. Related to the trend above, iSeries customers might be forced to upgrade print applications as a result of customer or supplier relationships. As an example, all suppliers to major retailers such as Wal-Mart must produce shipping documents, including bar codes, that conform to a supplier standard. These are "hard" standards that a vendor or supplier must implement to continue doing business. There are many additional "soft" standards where a company has strong incentives to implement application changes. These include the use of Maxicode bar code for shipping labels (driven by UPS), postal bar codes, and PDF417 bar code a high-capacity barcode used by more and more public agencies.

Document effectiveness: Your current statements are identical in form and content for all of your customers but you want to use information about individual customers to tailor their statement dynamically. This trend is accelerated with data warehouses and BI (Business Intelligence) tools that enable communications to customers to be much more customized.

Document design changes: Your documents change on a frequent basis and the current tools to make design changes cannot respond.

Document processes: Once documents and reports are fully electronic, the information flow that they represent can be re-engineered. For example, instead of printing sales reports each month, the reports can be produced electronically and organized within an archive, such as IBM Content Manager OnDemand for iSeries (OnDemand), for retrieval as required. Documents such as invoices, policies, or statements can be similarly stored for access by Customer Service (CRM) applications.

Competitive advantage or disadvantage: Your competitors are far ahead in document content, effectiveness, and process, which is starting to affect sales.

Internet application reengineering: You are transforming key business processes for the Web and your documents need to be fully electronic to support that.

Electronic delivery: You are looking at the possibilities of Infoprint® Server for iSeries: PDF and e-mail, electronic delivery of documents and reports — and again, you need fully electronic documents to take advantage of these capabilities.

IT Costs: The current system of changing design requires programming changes, escalating costs for what should be simple changes, requiring unacceptable delays, and causing disruption in stable application code.

iSeries Output Architecture

A good place to start is with a short primer about the iSeries output architecture. Figure 1 shows a conceptualized view of the output flow.

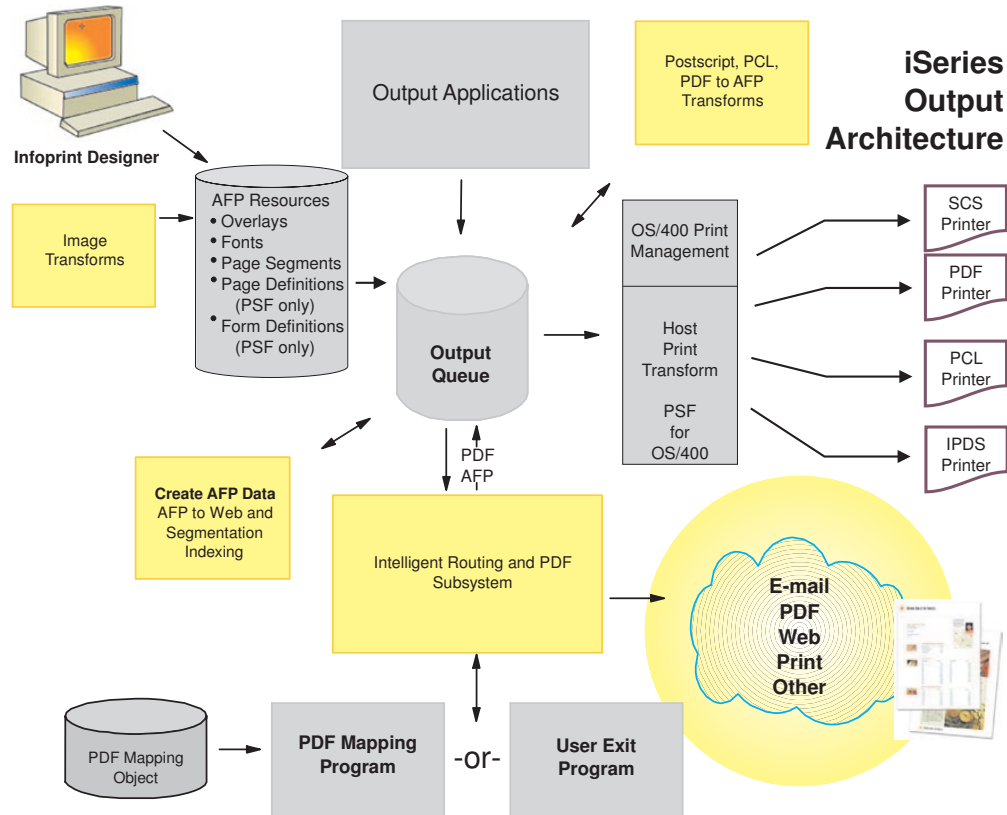


Figure 1. iSeries Print Flow

In this framework, iSeries applications produce output such as documents and reports by working through a printer file and optionally, Data Description Specifications (DDS). The generated print files, called *spooled files*, are written to iSeries output queues. Output queues are holding areas for spooled files and as "queue" implies, the files are ordered to be delivered out of the output queue in a first in, first out (FIFO) sequence. Also, non-iSeries applications can produce print files and route them to iSeries output queues.

If printing is the delivery method, Figure 1 illustrates the different printer drivers supported - OS/400, Host Print Transform, and Print Services Facility for OS/400 (PSF/400). The printer driver is selected automatically based on the type of spooled file and the protocol of the target printer.

In summary, business documents and reports are produced, staged within output queues, and then delivered to the appropriate consumer in the desired format.

Page Architecture

Before we progress any farther, we should zoom into the output queue and focus on what is contained within spooled files; namely, pages of information. Figure 2 shows a logical schematic of a typical business page.

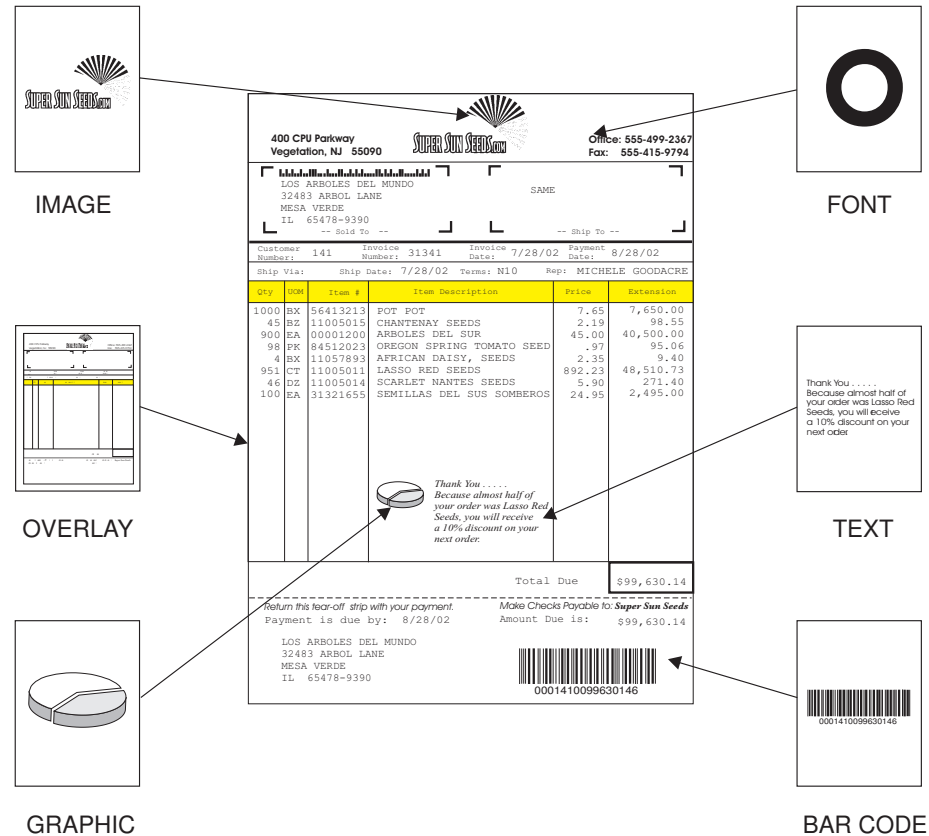


Figure 2. AFP Page Architecture

The document contains various logical components -- images, overlays, bar codes, and static and variable text in a variety of fonts. Advanced Function Presentation™ (AFP) is the integrated architecture for iSeries pages. For those applications that only produce simple text, the iSeries data stream is SNA Character Set (SCS). SCS does not support graphical page elements and is normally used for simple reports or documents that use pre-printed forms.

Because of the myriad requirements in today's IT environment for creating complex, fully electronic documents, we will focus on the graphical page and AFP. Besides handling the required graphical complexity, an additional key attribute of AFP page architecture is its component sub-architecture. Most of the page components are separate and external to the page data-stream. AFP was designed for business output, and certain operating characteristics -- performance, reliability, and security -- are essential for this environment. There are certainly other page data streams, such as PCL and PostScript. Both of these page data-streams support complex graphical elements but were designed for personal print applications, not business applications.

What does this "business design" mean? Let's take up performance. Business production printers currently top out at 2200 pages per minute, and speeds are increasing. With complex graphical applications, the system must be able to get all

of the page components to the printer at that throughput speed. With AFP, the page elements such as fonts, images (page segments), and overlays are separate iSeries objects. They are not embedded in the print data stream as with PostScript and PCL. This enables graphical objects that are used on multiple pages to be handled separately. In the case of high-speed printing, fonts, images, and overlays are cached in printer memory and are only sent to the printer if they are not already in printer memory.

This design function of AFP requires another element - interactive management. You will see two different data streams mentioned with AFP, AFP Data Stream (AFPDS) and Intelligent Printer Data Stream™ (IPDS™). IPDS is slightly different from the source AFPDS and includes operation code specific to the printer being addressed. The IPDS data stream facilitates the interactive dialog between iSeries and printer, controlled by PSF/400 (hereafter referred to as PSF). This dialog supports the page object caching we just described. It also enables dynamic verification that the object in printer memory is the latest version (and if it is not, PSF will automatically re-cache the latest version). The IPDS dialog is also central to the print process, managing errors and print recovery.

The separation of page elements into objects with AFP also provides for security. For example, a MICR font used in a check application is like any AFP font, a separate font object. As such, standard OS/400 security can be used to ensure that the font is only accessed by authorized users or applications.

As “Font Architecture” on page 45 depicts, the objects within AFP page architecture have their own definition. Another name for AFP is Mixed Object: Document Content Architecture for Presentation (MO:DCA-P). The page element objects follow an “OCA” pattern, as follows:

Table 1. AFP Page Elements and Respective Sub-architecture

Page Element	OCA Sub-Architecture
Font character sets, coded fonts, code pages	Font Object Content Architecture - FOCA
Bar codes	Bar Code Object Content Architecture - BCOCA
Overlays	Comprised of all OCA objects
Images (OS/400 page segments)	Image Object Content Architecture - IOCA
Graphics	Graphics Object Content Architecture - GOCA

Subsequent chapters within Part I describe each of these object content architectures in detail.

An additional note about ASCII data streams such as PostScript and PCL: OS/400 lets you queue ASCII output data by specifying the USERASCII data type. However, these spooled files are not integrated into the functions supported with SCS and AFP data streams such as:

- Spooled file view (with iSeries Access)
- Archive
- “Downstream” output functions such as fax, e-mail, and transform to PDF
- IPDS print management

Printer Files and DDS

Let’s go back to Figure 1 on page 4 and focus on the flow between the application program and the output queue. Every program on iSeries that produces output has

a printer file associated with it. The basic printer file determines overall attributes for the job. These include simplex or duplex, page range, finishing, orientation, and positioning (including N-up). As we will see, the printer file can also contain references to page definition and form definition objects, which can provide job attributes and page formatting specifications.

An optional element of the printer file is DDS. DDS is used in other areas of OS/400 to define database files, communication files, and other objects. The DDS within a printer file specifies how application data is placed on the page. Again, DDS is optional. If it is not used, then output data definition and placement must be defined within the application program. DDS is external to the application program but tightly linked to it. As we shall see, this linkage is important. The programmer can specify the data and position of every page element (text, lines, boxes, overlays, bar codes, and images) dynamically. The programmer can, in essence, create the page “on the fly” using DDS in conjunction with application program logic.

There are several disadvantages to using DDS for output design. Those disadvantages are in large part the reason for this guide and its case study of alternative document enabling and design tools. First, DDS, while not compiled with the application program, is intertwined in logic with it. Therefore, DDS is not application-independent (like Infoprint Designer for iSeries, for example) and DDS formatting changes normally involve application changes. Second, DDS is a programming interface and does not have a graphical design interface. With more and more complex fully graphical documents, this makes design and design changes more difficult. Finally, there are a few functions, such as dynamic formatting of paragraphs, that are not supported with DDS. Still, DDS remains a very powerful interface to the page.

AFP Page Resources Continued

If we go back to Figure 1 on page 4 and our iSeries output architecture, there are a number of page resources listed to the left of the output queue. We have already introduced the “OCA” page components – fonts, overlays, and page segments. Two additional page resources remain – page definitions and form definitions. These are less well-known but no less important. These OS/400 objects are job and page formatting resources as opposed to page components. A page definition is concerned with information layout on the page. It provides function similar to DDS. A form definition is concerned with job structure, such as simplex or duplex, number of copies, and orientation. It provides some of the functions found in the printer file. Both the page definition and form definition are, in reality, programming languages, thus the source statements of the language must be compiled in order to create the resource object.

These page resources can be used in conjunction with the printer file and DDS. The key advantage is application independence. You can design job logic and page content without changing the application program. The architecture of the page and form definition is sometimes called Page Printer Formatting Aid (PPFA). As we will see in Part II, these resources can be used in at least two ways. With the standard OS/400 source editor, PPFA source can be written and compiled with PPFA/400. Alternatively (and preferably) Infoprint Designer provides a fully graphical interface to the design of these page resources.

Printing

Once a spooled file has been created in an output queue, one delivery option is to print the file. There are several print deliverers, or printer writers, available. The

selection of the printer writer is automatic by OS/400 and based on the input data stream and the output printer. The three printer writers and their roles are summarized below:

Table 2. OS/400 Printer Writers

Printer Writer	Input Data Stream	Output Printer
OS/400 Printer Writer	SCS, Limited IPDS	SCS, Limited IPDS
Print Services Facility/400 (PSF/400)	AFP, IPDS	IPDS
LPR Driver (using Host Print Transform)	AFP, SCS, ASCII	ASCII
PJL Driver (using Host Print Transform)	AFP, SCS, ASCII	ASCII
SNMP Driver (using Host Print Transform)	AFP, SCS, ASCII	ASCII

OS/400 Printer Writer

The SCS printer driver is part of OS/400. It is the printer writer for Twinax-attached printers that can accept either SCS or IPDS. Because SCS is part of the Systems Network Architecture (SNA) protocol, it cannot be sent over TCP/IP protocol. To print to a TCP/IP-connected printer, the data stream must be converted to either IPDS or ASCII to make the transit. As Twinax becomes much less prevalent, this printer writer is less of a factor in iSeries printing.

Print Services Facility/400 (PSF/400)

PSF/400 (a priced feature of OS/400) is the print manager for AFP data streams and IPDS printers. If the target printer is IPDS, then control is passed from OS/400 to PSF/400. PSF/400 is central to making the AFP page architecture and IPDS print management process work.

ASCII Printer Drivers

If the target printer is ASCII – either line-oriented ASCII (such as ProPrinter) or page-oriented ASCII (PCL), then one of three print drivers can be selected. Those are (1) LPR (also known as remote output queue), (2) PJL driver, and (3) SNMP driver. All of these ASCII print drivers are intended for lower volume, less mission-critical printers. Without the AFP page architecture and IPDS-enabled print dialog, they cannot compete with IPDS in terms of functionality, printing management, and error recovery. LPR is the most limited in function – the ASCII data is simply sent to a TCP/IP address with no management. The PJL and SNMP print drivers add a little more intelligence (such as page range support). Target printers must support PJL or SNMP commands.

Because OS/400 spooled files are normally encoded in EBCDIC and are structured in either SCS or AFP data stream format, transform services are needed before invoking one of the ASCII printer drivers. The transform subsystem is Host Print Transform (HPT). HPT is responsible for converting SCS and AFP spooled files to ASCII. An alternative to this is Infoprint Server's ability to convert iSeries spooled files to PDF and respool them for printing on PDF-capable printers. See Chapter 22, "Working with Infoprint Server," on page 245 for more information.

E-business output

Figure 1 on page 4 has a number of non-print structures at the bottom. These elements represent new electronic delivery function added in OS/400 Version 5 Release 1. While printing certainly remains a central and critical delivery mechanism, electronic options have rapidly emerged.

Traditional output followed a “print and distribute” model. That is, application output was always printed then distributed to the point of need. However, the business computing model has shifted, becoming more technologically efficient and embracing e-business practices and models. To remain competitive, companies must restructure many business processes. An essential part of this re-structuring is the need to adjust the flow of output to systems, suppliers and customers. This re-structuring takes advantage of the network (Internet and intranets) and delivers reduced costs, increased service levels, and competitive advantage.

Business documents (which might be the only communications a company has with its customers or suppliers) in hard copy format do not fit very well into these e-business-driven process changes. For example, output that was formerly printed and distributed now might need to be produced and distributed electronically. “E-output” is a term that defines this shift in the flow of formatted information within business.

Let’s take an example - a supply chain document application. In the traditional print model, orders were received and processed. Shipping notices were created by an iSeries application program, and these documents were promptly printed (usually on top of a pre-printed form) and sent. In a re-engineered supply chain process, orders are taken over the Web and order documents would flow out immediately and electronically.

Let’s look at e-output in a little more detail. In general, there are two basic approaches to the electronic delivery of information (pages) – inbound and outbound. With inbound, the pages are organized within the iSeries database and the user proactively accesses them by a client or browser. Note that with inbound electronic access, the information (documents and reports) can reside in either iSeries databases or in archival systems. In fact, archival systems take on a significant middleware role in the organization of information for later retrieval. iSeries archive and retrieval systems such as OnDemand and OnDemand Common Server fill this role.

With outbound delivery, the formatted pages of information are normally sent directly (by e-mail) to the desired destination.

Chapter 22, “Working with Infoprint Server,” on page 245 introduces you to Infoprint Server for iSeries. Infoprint Server is a key piece in the iSeries output architecture. It provides a number of functions to enable e-output. First, Infoprint Server provides PDF services for iSeries. PDF is the standard electronic document distribution format for the Internet environment. Documents and reports produced in PDF format can be routed to an iSeries database for inbound electronic access, sent by e-mail, or both. Infoprint Server also supports production of portable AFP documents. Ideal for intranet applications, portable AFP can be routed like PDF and viewed either with iSeries Access or with browser AFP plug-ins.

An additional feature of Infoprint Server – in the top right of Figure 1 on page 4 – is the ASCII transform server. This enables the iSeries to extend its role as a print server, taking jobs that are being produced in non-iSeries formats (specifically, PCL, PostScript, and PDF) and converting them to AFP for iSeries-managed printing. These jobs could reside on network clients or servers, or on iSeries applications (such as ERP) that create ASCII data streams.

Chapter 2. What is Advanced Function Presentation (AFP)?

AFP is an architected system of hardware and software for creating, formatting, viewing, retrieving, printing, and distributing information on a wide variety of printer and display devices. It is the native, integrated data stream on the iSeries for generating fully composed pages of data.

Traditional line-mode print applications compose a line at a time down the page (usually continuous form pages), whereas AFP composes the entire page prior to printing. In AFP, page elements such as text, images, bar codes, and overlays, can be specified in any order at any position in the page. This is called *all points addressability*. AFP data stream (AFPDS) is a printer-independent data stream that composes full pages within a document. AFP was used to produce the invoicing document shown in Figure 3 on page 13.

The AFP architecture governs the creation and control of data types (such as text, fonts, images, graphics, bar codes, fax, color, audio, and multimedia) so that computer output is more readable and attractive. AFP's specific interchange architecture, MO:DCA-P, makes information interchange possible among different platforms using different protocols. These platforms include:

- OS/390[®] and z/OS[™]
- VM
- VSE
- OS/400
- AIX[®]
- Windows NT[®] and Windows[®] 2000

The term “mixed” in the MO:DCA-P architecture refers both to the mixture of data objects and the mixture of document constructs that make up the document. Figure 3 on page 13 illustrates the object-oriented nature of AFP. Each data object type has unique processing requirements. An Object Content Architecture (OCA) has been established for each IBM data object to define its respective syntax and semantics. MO:DCA-P documents can contain data and data objects governed by the OCAs described below. Notice the similarities between the list of OCAs and the list of AFP page elements in the description of Figure 3 on page 13.

Bar Code Object Content Architecture (BCOCA)

BCOCA defines how the various industry-standard bar codes are composed. A bar code object carries the alphanumeric information that is to be presented as a bar code and the controls to present this information in a bar code symbology. See “Bar Codes” on page 67 for information about bar codes.

Image Object Content Architecture (IOCA)

IOCA represents image data in device-independent format. This lets the image be scaled to the print resolution of different printers, for example. IOCA also supports image compression, reducing the storage size of the image data by 70% or more. See “Images and Graphics” on page 61 for information about images.

Graphics Object Content Architecture (GOCA)

GOCA describes complex pictures that are built by a series of graphic commands. On iSeries, Business Graphic Utility (BGU) and Graphic Data

Display Manager (GDDM[®]) create GOCA files such as bar charts and pie charts. See “Images and Graphics” on page 61 for information about graphics.

Presentation Text Object Content Architecture (PTOCA)

PTOCA defines how text is presented and positioned in the document. PTOCA must handle text in a wide variety of types, positions, and orientations.

Font Object Content Architecture (FOCA)

FOCA defines a consistent format and structure for font resources. See “Fonts” on page 41 for information about fonts.

MO:DCA-P documents can also contain or reference some non-OCA data objects that are registered in the MO:DCA-P architecture. Such data objects can be carried in a generic MO:DCA-P object envelope called an **object container**. Some examples of data objects that can be carried in an object container are image objects in Encapsulated PostScript (EPS) format, PDF single-page objects, and color mapping tables (CMT).

With MO:DCA-P IOB support, applications can include images, bar codes, page segments, and graphics in documents, taking advantage of scaling and rotating capabilities. For example, you can include a single copy of a company logo in a resource library, scaling and rotating the logo as necessary.

Note: Images, bar codes and graphics can be included by an IOB only if they are inline objects within the data stream. There is no mechanism in iSeries to store these as external objects on the system.

AFP Data Streams

Two strategic AFP presentation data streams are key components of the architected AFP printing solution:

- MO:DCA-P data stream, which was discussed above, is the application data stream through which applications can describe pages composed of text, images, and graphics data. MO:DCA-P is device independent; therefore, applications that produce this data stream can be directed to any of the printers supported by the AFP system or to graphical displays for viewing.
- Intelligent printer data stream (IPDS) is the printer device data stream that contains the information necessary to identify, monitor, and control the printer's functions. IPDS enables a two-way dialog between the printer and the printer driver, such as PSF, to create a cooperative print management system. IPDS is device dependent.

Elements of AFP Output

AFP output can contain any of these elements on each page:

- Text
- Images
- Graphics
- Bar codes
- Overlays

Figure 3 shows a sample of AFP output using PSF. The sample page is used throughout this publication to illustrate the various AFP functions.

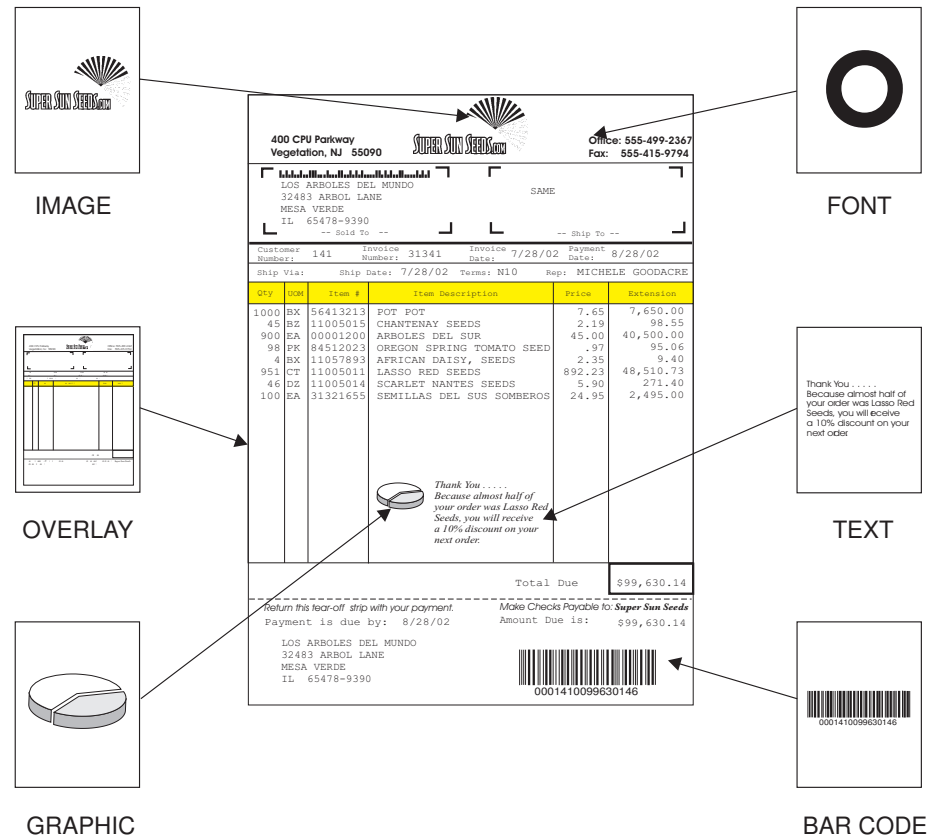


Figure 3. Sample of AFP Output Using PSF

Text and Fonts

Text is character data in the form of letters, numbers, punctuation marks, special characters, and signatures. Character data is printed using fonts, and each page can include several different fonts. You can find more information about fonts in “Fonts” on page 41 and Chapter 7, “Working with Fonts,” on page 79.

Images

Image data is a series of picture elements (pels) arranged in rows and columns.

Image data is created either by a scanning device or a program and is stored in a set of MO:DCA-P structured fields called an image data object. The image data can be included in a print job, in an overlay resource, or in a page segment resource. PSF accesses and sends image resources to the printer with the print job. See “Images and Graphics” on page 61 and Chapter 8, “Working with Images,” on page 87 for more information.

Graphics

Graphics data contains commands to draw lines, arcs, and circles and can be used to represent something as complex as a three-dimensional engineering drawing.

Graphics data is created by a program and stored in a set of MO:DCA-P structured fields called a graphics data object. A graphics data object can be included in a print job, in an overlay resource, or in a page segment

resource. PSF accesses and sends graphics resources to the printer with the print job. See “Images and Graphics” on page 61 and Chapter 8, “Working with Images,” on page 87 for more information.

Bar Codes

Bar codes represent characters two ways: by sets of parallel bars of varying thickness and separation that can be read optically by transverse scanning (linear bar codes) or in a two-dimensional matrix (2D bar codes). Linear bar codes are often used to represent product numbers, part numbers, and zip codes. 2D bar codes can store much larger amounts of information by using extensive data compaction and error correction codes.

Many different types of bar code arrangements, or symbologies, have been developed for specific applications. The zip code at the top of the sample page is printed in POSTNET bar code. The customer number and invoice amount are printed in Interleaved 2 of 5 bar code at the bottom of the page.

For more information about bar codes, see “Bar Codes” on page 67 and Chapter 10, “Working with Bar Codes,” on page 101.

Overlays

The ability to combine data with print resources enables you to use an application to merge data with an overlay, or electronic form. An overlay contains constant information and the print job created by the application contains the variable data that is printed on the overlay. PSF can print both the variable data and the overlay on one sheet, eliminating the need for preprinted forms when printing the variable data. PSF also can print an overlay on a blank page containing no variable data. For more information see “Overlays” on page 65 and Chapter 9, “Working with Overlays,” on page 95.

PSF for OS/400

PSF for OS/400 (hereafter referred to as PSF) is the AFP system software (printer writer) for OS/400 printers that use IPDS. It can be thought of as the glue between the application and the printer. PSF enables OS/400 users and applications to take full advantage of IPDS printer capabilities.

OS/400 supports line printers and a subset of IPDS printers and print functions without PSF, while PSF provides full support for all IPDS printers. Because PSF is integrated with OS/400 print support mechanisms such as printer files and data description specifications (DDS), it is an extension of print processes already familiar to most OS/400 users. Whether PSF is used to process application output is determined by the device description for the target printer. Only printers defined as AFP(*YES) in the printer device description are controlled by PSF.

PSF ensures that applications and their AFP resources print consistently on all of the printers it manages. PSF transforms and prints these data streams:

- XML (requires an XML page definition)
- PostScript (requires Infoprint Server feature 5101)
- PDF (requires Infoprint Server feature 5101)
- PCL
- IBM MO:DCA-P, also called AFPDS, the device-independent AFP page data format
- IPDS, the device-dependent page printer data stream
- SNA Character String (SCS)

- Extended binary-coded decimal interchange code (EBCDIC) line printer data

Print Flow with PSF

The PSF print flow, shown in Figure 4, is similar to the OS/400 print flow, making AFP on the OS/400 easy to use.

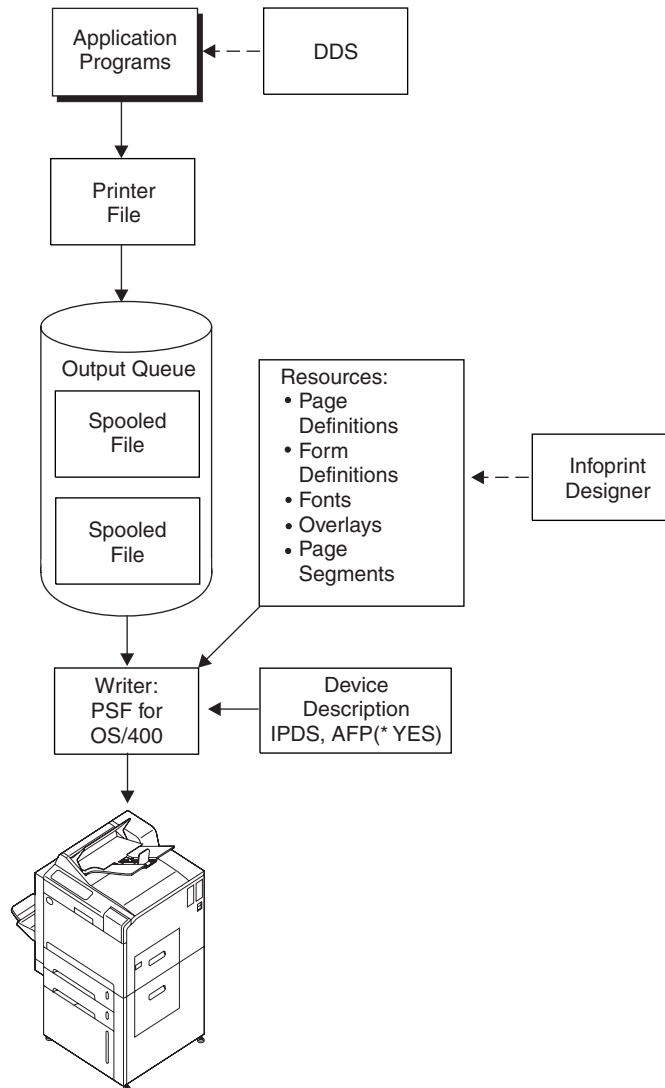


Figure 4. PSF Print Flow

The PSF print flow is described below.

1. **Application programs** generate output that can be either program-described or externally-described. If the output is program-described, then the application program (in RPG or COBOL, for example) formats all of the data on the page one line at a time. The program can produce AFP directly, or it can use an application enabler such as AFP Toolbox to create the output.
If the output is externally-described, the application program simply creates records. These records are processed by DDS, which is separate source and object code from the RPG or COBOL program.
2. **DDS keywords**, if used, determine how each field of every record from the application program is formatted on the page. Fields can be individually placed

on the page in any font or orientation. DDS keywords can be used for many formatting options, such as including bar codes, drawing boxes, and specifying overlays. You can use DDS with or without PSF, but many more formatting capabilities are available with PSF. See Chapter 14, "Using Data Description Specifications," on page 125 for more information.

3. The application program always uses a **printer file**. This is the same printer file used without PSF. Values specified in the printer file determine general characteristics of the entire print job, such as what printer the job is sent to, whether to print simplex or duplex, whether to print multiple pages per side, and whether to add an overlay to every page.
4. A **spooled file** is created on an output queue, just as it is in the OS/400 print flow. The difference is that an AFP spooled file can contain references to resources. These references can result from the printer file parameters and DDS keywords that were used when the file was created. The resources might be downloadable fonts, overlays, and page segments. Other resources that can be referenced by a spooled file are page definitions and form definitions, even though they are less common in the OS/400 environment. See "Page Definitions and Form Definitions" on page 72 for more information.
5. In the PSF print flow, the **device description** of the printer to which the output is sent specifies a device type of IPDS and also specifies AFP(*YES). This means that the printer is an IPDS device capable of handling all AFP functions, such as downloading resources and printing graphical information from DDS.
6. **PSF** is the printer writer
PSF has three primary functions:
 - Take the spooled file off the output queue and perform any necessary data stream transforms on it. The spooled file data type can be SCS, line, IPDS, or AFP, as specified by the printer file. The spooled file must be transformed into the correct printer-specific IPDS data stream.
 - Gather any resources referenced by the spooled file, such as fonts, page segments, and overlays.
 - Send the transformed spooled file and its resources to the printer. Because IPDS is a bidirectional data stream, PSF must also participate in a dialogue with the printer. Through this dialogue, PSF manages error recovery. If printer intervention is required during the job printing, for example, PSF resumes printing at the next page in the job so that no data is lost and the job does not have to be re-sent to the printer.

Printers and PSF

PSF supports the entire family of IBM IPDS printers as well as IPDS printers from other vendors. PSF support may be required or optional, depending upon the particular printer specified.

When PSF Is Required

PSF is required if your printer falls into one of these categories:

- Any TCP/IP attached IPDS printer
- Any Twinax attached IPDS printer that uses PSF functions (specifically, AFP functions such as overlays, fonts, page segments - see Table 3 on page 17 and Table 4 on page 17)

PSF is also required when:

- OS/400 is used to print any AFP, line, or mixed-mode data.

- You use any of the parameters listed in Table 3 and are printing to an IPDS printer.
- You use any of the keywords listed in Table 4 and are printing to an IPDS printer.

Table 3. Printer File Parameters that Require PSF

AFP characters (AFPCHARS)	Front margin (FRONTMGN)
Back margin (BACKMGN)	Front overlay (FRONTOVL)
Back overlay (BACKOVL)	IGC coded font (IGCCDEFNT)
Coded font (CDEFNT)	IPDS pass through (IPDSPASTHR)
Corner staple (CORNERSTPL)	Output bin (OUTBIN)
Device type (DEVTYPE=*AFPDS, *LINE, or *AFPDS LINE)	Page definition (PAGEDEF)
Edge stitch (EDGESTITCH)	Saddle stitch (SADLSTITCH)
Font character set (FNTCHRSET)	Table reference characters (TBLREFCHR)
Font resolution for formatting (FNTRSL)	User resource library list (USRRSCLIBL)
Form definition (FORMDEF)	

Table 4. DDS Keywords that Require PSF

Box (BOX)	Line (LINE)
Coded font (CDEFNT)	Output bin (OUTBIN)
Duplex (DUPLEX)	Overlay (OVERLAY)
End page (ENDPAGE)	Page segment (PAGSEG)
Font character set (FNTCHRSET)	Position (POSITION)
Force (FORCE)	Staple (STAPLE)
Graphic data file (GDF)	Text rotate (TXTRTT)
Invoke medium map (INVMMAP)	Z-fold (ZFOLD)
Invoke data map (INVDTAMAP)	

Chapter 3. AFP Applications

Because AFP is integrated into the iSeries, it is natural that there are many OS/400 applications that work with it. These are the types of applications that you can use to generate and work with AFP:

enabling applications

Applications that help you generate AFP

complimentary applications

Applications that work with AFP output or AFP data streams

Enabling Applications

The following list describes the enabling applications and their components.

- **AFP PrintSuite**

AFP PrintSuite for iSeries is a family of different solutions for building AFP applications. The following tools (separately orderable) are generally application-independent approaches to formatting electronic pages.

- **Advanced Print Utility (APU)**

APU is an interactive, end-user system for transforming existing application output to advanced electronic output. Current line-mode output can be migrated to electronic business documents utilizing electronic forms, image, graphics, fonts, and bar codes. APU will create complex output, including multiple page formats and conditional logic, all without changes to the existing application. For more information about APU, refer to this Web page:

<http://www.ibm.com/printers/R5PSC.NSF/Web/APU400Home>

- **Page Printer Formatting Aid (PPFA)**

PPFA provides the ability to create AFP page definitions and form definitions on iSeries. Page definitions and form definitions are print format resources that control how application output is to be presented on the printed page. They provide document formatting and data mapping capabilities, such as conditional processing, electronic overlays, and multiple copies. Page definitions and form definitions support the full range of print application requirements. They provide an application-independent, high-performance method of transforming existing print applications to electronic output. For more information about PPFA, refer to this Web page:

<http://www.printers.ibm.com/R5PSC.NSF/Web/PPFAHome>

- **AFP Toolbox**

AFP Toolbox enables you to take full control over the power of the AFP data stream, but do it with an easy, object programming interface. AFP Toolbox, through a set of C and RPG objects, functions, and callable APIs, enables you to update your print applications to take full advantage of printer technologies such as color, image, and overlays. AFP Toolbox should be used where you have a requirement for a dynamic program interface to the printed page. For more information about AFP Toolbox, refer to this Web page:

<http://www.ibm.com/printers/R5PSC.NSF/Web/AFPToolHome>

- **SAP R/3 AFP Print**

SAP R/3 AFP Print converts SAP's R/3 data streams into IBM's AFP data stream or line data. This support includes the capability to customize output with signatures, logos, and bar codes through inclusion of AFP resources. This output can then be printed on AFP printers using PSF.

In addition, extended electronic formatting can be added to the R/3 output through the use of page definitions and form definitions. This enables the migration of R/3 line-mode output to electronic documents, with overlays, images, and bar codes.

For more information about PrintSuite and its features, refer to the PrintSuite Web page at <http://www.printers.ibm.com/R5PSC.NSF/Web/AFPPrintsuiteHome>

- **IBM Advanced Function Printing Utilities for iSeries (AFP Utilities)**

AFP Utilities are a set of three utilities that support the creation of AFP applications. AFP Utilities include:

- **Overlay Utility**

With Overlay Utility, you can create electronic overlays on the iSeries to be used in AFP applications. Overlay Utility works on any iSeries terminal, and provides an integrated platform to design, create, and print overlays.

- **Print Format Utility**

Print Format Utility is an interactive tool for building AFP output directly from the iSeries database, without writing any application programs. With Print Format Utility, you can quickly create complex electronic output combining variable data, overlays, bar coding, images, and other document elements.

- **Resource Management Utility**

Resource Management Utility is a tool to manage the overlay and page segment resources used in AFP applications. With Resource Management Utility, you can create, copy, convert, and print these resources.

For more information about AFP Utilities, refer to this Web page:
<http://www.ibm.com/printers/R5PSC.NSF/Web/AFPUilitiesHome>

- **AFP Font Collection for Workstations and OS/400**

AFP Font Collection for Workstations and OS/400 (Program Number 5648–B45) provides font resources for the OS/400 operating environment. In addition to the font resources, an orderable feature of this product (Type Transformer and Utilities for Windows) provides font utilities that allow you to edit outline fonts, create AFP font resources from Type 1 outline fonts, create and modify code pages, and create and modify coded fonts. DBCS CID-keyed outline fonts are also provided for the Chinese, Japanese, and Korean languages. The Chinese, Japanese, and Korean fonts can be used with Type Transformer to create 240-pel and AFP Outline Font resources. AFP Font Collection comes free with PSF, and the Font Collection fonts are included with Infoprint Designer for iSeries.

You do not typically need every font library installed on your iSeries. See “Which Font Libraries Should I Install?” on page 49 for suggestions about which font libraries to install. “Steps to Restore Font Libraries From the Font Collection CD to the iSeries” on page 84 has instructions for restoring AFP Font Collection fonts to your iSeries.

There are two commands that you can download and use to load AFP Font Collection fonts and print a catalog of font samples. For more information about these commands and to download them, refer to <http://www.printers.ibm.com/r5psc.nsf/web/rdfont01>

For more information about AFP Font Collection for Workstations and OS/400, refer to this Web page:

<http://www.ibm.com/printers/R5PSC.NSF/Web/AFPSFontHome>

- **IBM Infoprint Fonts for Multiplatforms**

IBM Infoprint Fonts for Multiplatforms (Program Number 5648-E77) includes the outline fonts available with AFP Font Collection for Workstations and OS/400 and the Type Transformer and Utilities optional feature, which enables you to edit and manage fonts. This product contains enhancements to the GUIs available with AFP Font Collection and Type Transformer, as well as enhancements for Eastern European and Asia Pacific languages and more support for China and Japan.

The Font Installer for AFP Systems feature of Infoprint Fonts for Multiplatforms is required to load TrueType and OpenType fonts on your system. For more information about Infoprint Fonts for Multiplatforms, refer to this Web page:

<http://www.ibm.com/printers/R5PSC.NSF/Web/ipfontshome>

- **IBM AFP Printer Driver for Windows**

The IBM AFP Printer Driver for Windows creates AFP documents, overlays, and page segments from any Windows application. The AFP data stream can be uploaded for printing on the OS/400, or sent to a file for viewing by the AFP Viewer. You can also use the IBM AFP Printer Driver for Windows to create electronic overlays and page segments for use in your OS/400 AFP documents.

For more information about the AFP Printer Driver for Windows or to download the driver, refer to this Web page:

http://www.printers.ibm.com/R5PSC.NSF/Web/afp_printer_driver

- **Business Graphics Utility (BGU)**

BGU is an interactive utility for developing graphical data such as graphs and pie charts. The output of BGU, a graphic data file, can be used within an AFP document.

- **Graphical Data Display Manager (GDDM)**

GDDM is a set of programming routines that can be used to generate graphical data directly from a high-level program. The output of GDDM, a graphic data file, can be then be used with AFP output.

- **Infoprint Designer for iSeries**

Infoprint Designer for iSeries (Program number 5733-ID1), hereafter referred to as Infoprint Designer, has three parts to help you work with your iSeries data: Image Editor, Overlay Editor, and Layout Editor. The Image Editor and Overlay Editor are just that – utilities for editing (and creating) images and overlays. The Layout Editor helps you put it all together. You can use it to seamlessly download spooled files that you have retained from an application run and display them in the Layout Editor for drag-and-drop placement on your electronic forms. If you are migrating from continuous pre-printed forms with each page identically embossed, you only need to do minimal reformatting to fine tune the layout. With Infoprint Designer's graphical design capability, it makes the application-independent formatting easier, and you can produce layout projects with or without electronic forms in the background.

Starting in version 1.11 (provided with PTFs SF67603 and SF67604), Infoprint Designer comes with a layout wizard, which makes it easier for you to create complex layouts in the Layout Editor. Use the Layout wizard to help you create layouts with modified copies, with different formats on the front and back of a duplexed sheet, or that use conditional processing to change layouts from page to page in a print job.

For information about Infoprint Designer, refer to this Web page:

http://www.printers.ibm.com/internet/wwwsites.nsf/vwebpublished/ipdesignerhome_i_ww

Complementary Applications

- **Infoprint Server for iSeries**

Infoprint Server for iSeries (Program Number 5722-IP1) provides the abilities to e-mail OS/400 spooled files (as PDF files) and consolidate network printing on the iSeries with its PostScript, PCL, and PDF to IPDS print transforms. It also includes image transforms to convert PC-based gif, tiff, and jpeg images to AFP page segments, overlays, or documents. With its Create AFP Data (CRTAFPDTA) command, you can add index tags to line or AFP data. The PDF subsystem lets you convert a spooled file to PDF then store it as a stream file, put it on an output queue, or e-mail it.

For more information about Infoprint Server for iSeries, refer to this Web page:
<http://www.printers.ibm.com/R5PSC.NSF/Web/ipserv400Home>

- **@server iSeries Access for Windows**

@server iSeries Access for Windows, hereafter referred to as iSeries Access, lets you connect OS/400 systems and personal computers running a variety of Microsoft® Windows operating systems. iSeries Access offers an all-inclusive client solution for accessing and using resources from your Windows desktop. It includes 5250 emulation, access to DB2 Universal Database™ (UDB) for iSeries through its Data Transfer, and utilizes AS/400 NetServer for working with the OS/400 Integrated File System and printers. It has several components, including iSeries Navigator and AFP Workbench Viewer.

- **AFP Workbench Viewer**

- Provides a high-function client interface to iSeries output. You can view any spooled file fully and graphically, including AFP output. Once viewed, output can be searched, faxed, and printed.

- **iSeries Navigator**

- IBM iSeries Navigator is a graphical interface for managing your iSeries servers. With V5R2, iSeries Navigator implements an interactive interface to PDF and e-mail functions, enabling you to select one or more print files and define immediate PDF and e-mail options. iSeries Navigator also includes AFP Manager, which helps you create and manage AFP objects.

For more information about iSeries Access, refer to this Web page:
<http://www-1.ibm.com/servers/eserver/iseries/access/>

- **iSeries Access for Web**

iSeries Access for Web is the Web browser version of iSeries Access. It provides many of the functions supported by iSeries Access but is installed on the iSeries server and requires only a browser on the client machine. It also integrates with Infoprint Server's PDF support to display and print iSeries spooled files in PDF format.

For more information about iSeries Access for Web, refer to this Web page:
<http://www-1.ibm.com/servers/eserver/iseries/access/web/>

- **IBM Toolbox for Java**

The IBM Toolbox for Java™ is a library of Java classes supporting the client/server and internet programming model to an iSeries or AS/400 server. The classes can be used by Java applets, servlets, and applications to easily access server data and resources. IBM Toolbox for Java provides support similar to functions available when using the iSeries Access APIs. With V5R2, the Toolbox includes an access method for duplicating spooled files. This enables dynamic interfaces to PDF functions. For more information about this method, see "Accessing the PDF Subsystem" on page 255.

For more information about IBM iSeries Toolbox for Java, refer to this Web page:
<http://www-1.ibm.com/servers/eserver/iseries/toolbox/>

- **AFP Viewer Plug-in**

The AFP Viewer Plug-in is a downloadable plug-in to Microsoft Internet Explorer or Netscape Navigator. It lets you view any spooled file fully and graphically, including AFP output. Once viewed, output can be searched, faxed, and printed.

You can download the AFP Viewer Plug-in free from IBM at:

<http://www.printers.ibm.com/R5PSC.NSF/web/afpviewer>

- **Facsimile Support for iSeries**

Facsimile Support for iSeries enables you to use facsimile, or fax, directly from an iSeries system. It is a generalized facsimile enabler for both inbound and outbound faxes that lets you integrate your facsimile transmissions directly into your applications.

Incoming faxes can be viewed using iSeries Access or printed on IPDS printers. Any *SCS or *AFPDS iSeries spooled file can be faxed outbound. Facsimile Support also supports AFP overlays as fax cover sheets (for *AFPDS spooled files only).

For more information, refer to this Web page:

<http://www-1.ibm.com/servers/eserver/iseries/fax400/>

- **Content Manager ImagePlus**

Content Manager ImagePlus[®] enables you to turn paper documents into electronic images that then can be stored, viewed and printed. ImagePlus provides the ability to electronically capture, import, index, store, retrieve, display, process, distribute, and print image information.

- **ImagePlus Workfolder Application Facility (WAF/400)**

ImagePlus Workfolder Application Facility (WAF) provides an application for document-driven workflow and folder management. WAF also provides a set of APIs for the image enablement of business applications. WAF documents, being image-coded data, can be viewed with the iSeries AccessViewer and printed on IPDS printers.

- **Content Manager OnDemand for iSeries (OnDemand)**

OnDemand offers online document capture, indexing, storage, and retrieval of large volumes of electronic data, such as AFP spooled files. This data can be stored on disk, optical, or tape storage media. Once stored, you can view, print or fax individual, selectively retrieved segments of the spooled file, such as an individual invoice or department report.

For more information, refer to this Web page:

<http://www.ibm.com/software/data/ondemand/400>

- **Business Partner Applications**

The second category includes applications that work with or transform AFP data. These include report archive and retrieval systems that index, store, manage, and retrieve output files. These also include various print serving systems that route, transform and print OS/400 output.

- **Backup Recovery and Media Services (BRMS)**

Backup Recovery and Media Services (BRMS) for iSeries provides backup, recovery, and archive services for tape devices. iSeries objects, libraries, folders, spooled files, configurations, and Integrated File System data can be saved, restored, and archived to tape. Output queues can be backed up, including external print resources (overlays, page segments) referenced in AFP spooled files.

For more information about BRMS, refer to the iSeries Information Center.

- **Third Party Design Tools**

There are a number of tools that enable you to design and work with AFP resources, such as overlays, fonts, page segments, page definitions, and form

definitions. Some of these products are listed below. For more information about these products, refer to the appropriate Web site.

DOC1 DOC1 is a WYSIWYG document composition system from Group 1 Software. DOC1 designs individualized documents, such as statements, directives, bills, and other communications, that can be processed and managed across multiple platforms. For more information about DOC1, refer to this Web site: <http://www.g1.com/doc1/>.

Opus Opus is a Windows-based integrated document composition and production system from Elixir Technologies Corporation. Opus is a product that develops, prints, and presents documents that are tailored to meet individual customer needs. For more information about Opus, refer to this Web site: <http://www.elixir.com/products/opus.asp>.

Papyrus

Papyrus is a WYSIWYG document automation solution from ISIS Papyrus. Papyrus enables development, production, and management of business documents. For more information about Papyrus, refer to this Web site: <http://www.isis-papyrus.com>.

StreamServe

The StreamServe Business Communication Platform provides advanced Distributed Output Management capabilities to help reduce the cost of print and fax. For more information about StreamServe, refer to this Web site: <http://www.streamserve.com/>

Chapter 4. Introduction to Printer Files

Printer files, both system provided and user created, help you specify how your output appears on the page. In a printer file you can specify such options as overlays to be used, whether the document is printed duplex, number of lines per inch, and to which output queue the spooled file is sent.

Printer files can contain two different types of information: parameters that apply to the entire spooled file and control how spooled file data is processed; and printer DDS, which defines how application data is positioned on the output page. The printer file parameters are part of any printer file on the system. Printer DDS is optional and exists only if the application program has externally-described output. For information about using DDS in printer files, see “Printer File DDS” on page 125.

Every job on the OS/400 has an associated printer file, whether or not you create one.

Note: Jobs that do not have an option to generate spooled output do not require printer files. However, every application that generates spooled output must use a printer file to do so.

Refer to the iSeries Information Center or *Printer Device Programming* for a list of printer file parameters.

Printer File Flow

Because most application output on OS/400 is spooled, you have two opportunities to affect how your output looks:

1. Before the application builds the spooled file
2. While the spooled file is on the output queue waiting to be printed

To affect how your output is handled before from the application program creates the spooled file, use one of these commands prior to processing the application:

Create Printer File (CRTPRTF)

Creates a new printer file.

Change Printer File (CHGPRTF)

Permanently changes the printer file.

Override with Printer File (OVRPRTF)

Temporarily changes the print parameters to be used by the application program for your session. This is the more commonly used of the three commands because the changes are temporary and do not affect other users. Figure 5 on page 26 shows the opening screen for the OVRPRTF command.

```

Override with Printer File (OVRPRTF)

Type choices, press Enter.

File being overridden . . . . . Name, *PRTF
Overriding to printer file . . . *FILE Name, *FILE
Library . . . . . Name, *LIBL, *CURLIB
Device:
Printer . . . . . Name, *SYSVAL, *JOB
Printer device type . . . . . *SCS, *IPDS, *USERASCII...
Page size:
Page length . . . . . .001-255.000
Page width . . . . . .001-378.000
Measurement method . . . . . *ROWCOL *ROWCOL, *UOM
Lines per inch . . . . . 3, 4, 6, 7.5, 7,5, 8, 9, 12
Characters per inch . . . . . 5, 10, 12, 13.3, 13,3, 15...
Front margin:
Offset down . . . . . 0-57.790, *DEV
Offset across . . . . . 0-57.790

More...
F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys

```

Figure 5. Override with Printer File (OVRPRTF) Command

To change spooled file attributes after the file has been spooled, issue Change Spooled File Attributes (CHGSPLFA). This lets you change such attributes as which output queue the spooled file is stored on and whether the spooled file is saved.

Figure 6 shows the display that opens when you use the CHGSPLFA command.

```

Change Spooled File Attributes (CHGSPLFA)

Type choices, press Enter.

Spooled file . . . . . Name, *SELECT
Job name . . . . . * Name, *
User . . . . . Name
Number . . . . . 000000-999999
Spooled file number . . . . . *ONLY 1-999999, *ONLY, *LAST, *ANY
Job system name . . . . . *ONLY Name, *ONLY, *CURRENT, *ANY
Spooled file created:
Creation date . . . . . *ONLY Date, *ONLY, *LAST
Creation time . . . . . Time, *ONLY, *LAST
Select files for:
User . . . . . *CURRENT Name, *CURRENT, *ALL
Print device . . . . . *ALL Name, *ALL, *OUTQ
Form type . . . . . *ALL Form type, *ALL, *STD
User data . . . . . *ALL User data, *ALL
Printer . . . . . *SAME Name, *SAME, *OUTQ
Print sequence . . . . . *SAME *SAME, *NEXT

More...
F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys

```

Figure 6. Change Spooled File Attributes (CHGSPLFA) Command

Printer File Parameters

Printer file parameters provide a wide range of printing functions, from the basics of managing the spooled file to options that will transform simple line output to advanced electronic output.

With spooled file-level parameters you can specify the following and more:

- The output queue to use.
- Whether a spooled file is saved or held.

- The printer device or device type used, including changing from one type of printer to another (for example, SCS to IPDS).
- The input drawer and output bin, for page printers.
- Simplex or duplex.
- Whether to automatically convert line data to AFP. The AFP spooled file can be viewed and can be printed on more types of printers.

You can use the parameters affecting printed lines to specify these values and more:

- The default font for printed output.
- Characters per inch (CPI) and lines per inch (LPI).
- Page size and page rotation.
- Front and back margins or offsets for the printed lines.

With AFP, you have options to transform simple line output to advanced, full-page documents. You can do this and more:

- Add front and back overlays to your output, including a constant back overlay.
- Print your output in 2-up, 3-up, or 4-up page formats.
- Reduce the output size so that, for example, it can be printed on 8-1/2 by 11 inch paper.
- Use the line data, page definition, and form definition parameters to completely transform the application output. Chapter 16, “Using Page Printer Formatting Aid,” on page 183 and Chapter 17, “Using Infoprint Designer,” on page 197 describe how to build page definitions and form definitions.
- Specify finishing operations, including stapling and punching.
- Specify an e-mail address to which transformed data is sent. This requires Infoprint Server.

During the printing process, not all of the printer file parameters can be used both before and after print spooling. For more information about printer file parameters, refer to the iSeries Information Center or *Printer Device Programming*.

Using Printer Files to Enhance Your Output

There are several ways to enhance your output with printer files. This section describes how to do these tasks:

- “Specifying Line and Font Changes”
- “Specifying Multi-Up Printing” on page 29
- “Specifying Computer Output Reduction” on page 30
- “Adding Overlays” on page 32
- “Converting Line Data to AFP” on page 34
- “Specifying Finishing Operations” on page 36

Specifying Line and Font Changes

You can modify the appearance and spacing of application output by using the line, character, and font parameters in the printer file. Different combinations of the LPI and CPI parameters let you change the line and character spacing.

You can modify the default font used for spooled file output in these ways:

- By specifying a printer-resident font using the FONT parameter.
- By specifying a host-resident font. You can specify a host-resident font by using the coded font (CDEFNT) or by using the font character set (FNTCHRSET). The

coded font maps to a specific code page and character set. With FNTCHRSET, you specify the code page and character set explicitly.

Use the LPI, CPI, FONT, CDEFNT, and FNTCHRSET parameters in the printer file to control line spacing, character spacing, and fonts. The appropriate parameters are shown in bold in the below displays:

```

Override with Printer File (OVRPRTF)

Type choices, press Enter.

File being overridden . . . . . Name, *PRTF
Overriding to printer file . . . *FILE Name, *FILE
Library . . . . . Name, *LIBL, *CURLIB
Device:
Printer . . . . . Name, *SYSVAL, *JOB
Printer device type . . . . . *SCS, *IPDS, *USERASCII...
Page size:
Page length . . . . . .001-255.000
Page width . . . . . .001-378.000
Measurement method . . . . . *ROWCOL *ROWCOL, *UOM
Lines per inch . . . . . 3, 4, 6, 7.5, 7.5, 8, 9, 12
Characters per inch . . . . . 5, 10, 12, 13.3, 13.3, 15... Front margin:
Offset down . . . . . 0-57.790, *DEV
Offset across . . . . . 0-57.790

More...
F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys

```

Figure 7. Override with Printer File Display - First Screen

```

Override with Printer File (OVRPRTF)

Type choices, press Enter.

Back margin:
Offset down . . . . . 0-57.790, *FRONTMGN, *DEV
Offset across . . . . . 0-57.790
Overflow line number . . . . . 1-255
Fold records . . . . . *NO, *YES
Unprintable character action:
Replace character . . . . . *NO, *YES
Replacement character . . . . ' ' 40-FE
Align page . . . . . *NO, *YES
Source drawer . . . . . 1-255, *E1, *FORMDF
Output bin . . . . . 1-65535, *DEV
Font:
Identifier . . . . . Character value, *CPI...
Point size . . . . . *NONE 0.1-999.9, *NONE
Form feed . . . . . *DEV, *CONT, *CUT, *CONT2...
Print quality . . . . . *STD, *DEV, *DRAFT, *NLQ...

More...
F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys

```

Figure 8. Override with Printer File Display - Second Screen

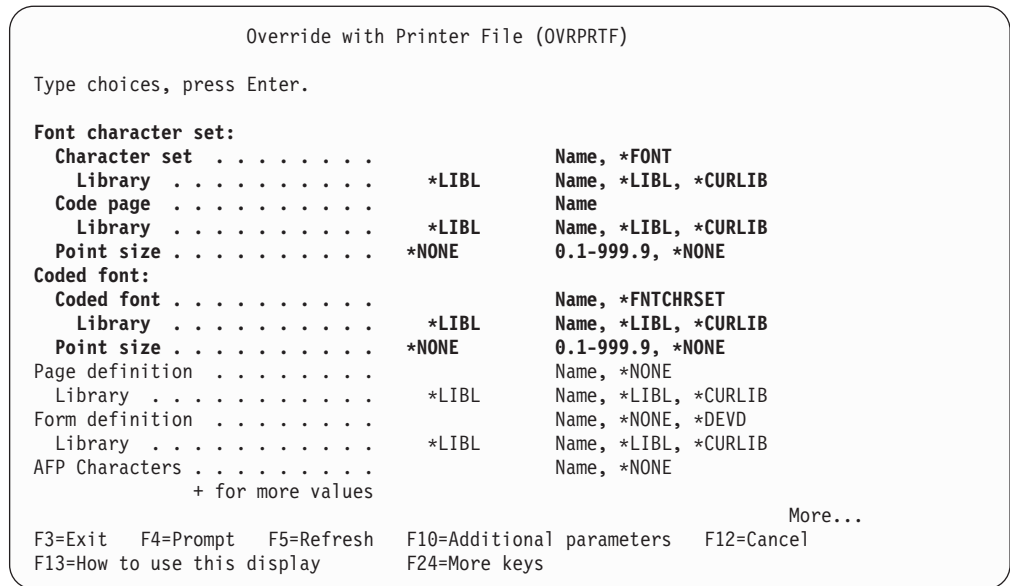


Figure 9. Override with Printer File Display - Third Screen

Specifying Multi-Up Printing

Multiple-up printing lets you print several logical pages on a single physical page. This offers significant advantages in cost and volume of printed reports. Combining this function with duplex printers doubles its value.

The MULTIUP parameter works with the LPI, CPI, PAGRTT, and PAGESIZ parameters of the spooled file and uses the most appropriate printer-resident fonts to fit the logical pages onto an 8-1/2 by 11 inch page with 1/2-inch margins.

Many printers support oversize paper stock such as 11 by 17 inch paper. You can print two full size 8-1/2 by 11 inch pages on 11 by 17 inch paper by specifying MULTIUP (2) and REDUCE (*NO).

Figure 10 on page 30 shows an example of multi-up printing. 4-up printing might push the limits of readability, but it may be the right solution for certain high-volume reports. Adding an overlay also can improve the look and readability of the report.

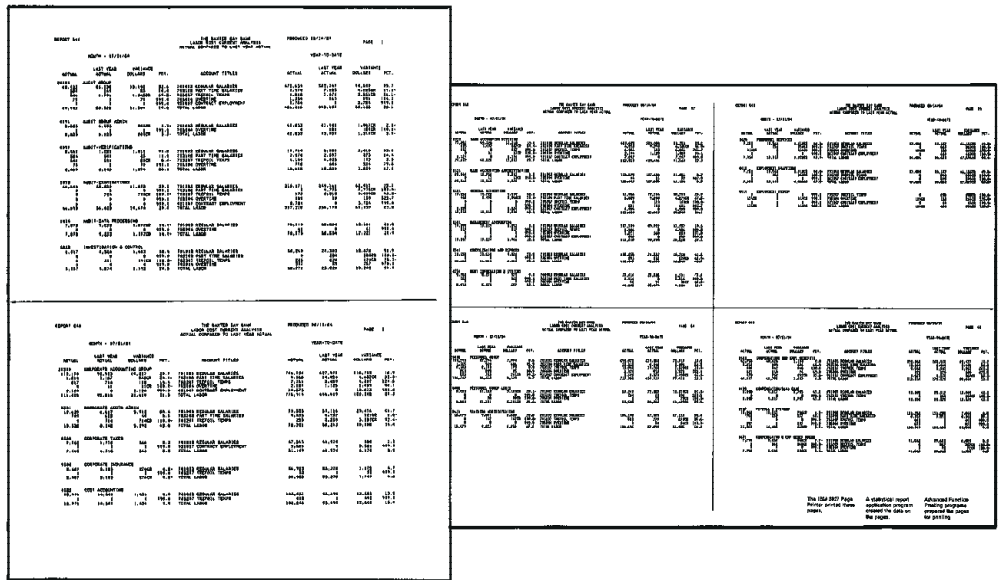


Figure 10. Multi-Up Printing: Example of 2-up and 4-up Printing

Specify a value in the range 1-4 for MULTIUP in the printer file to use this function. The Pages per side parameter is shown in bold in the below display:

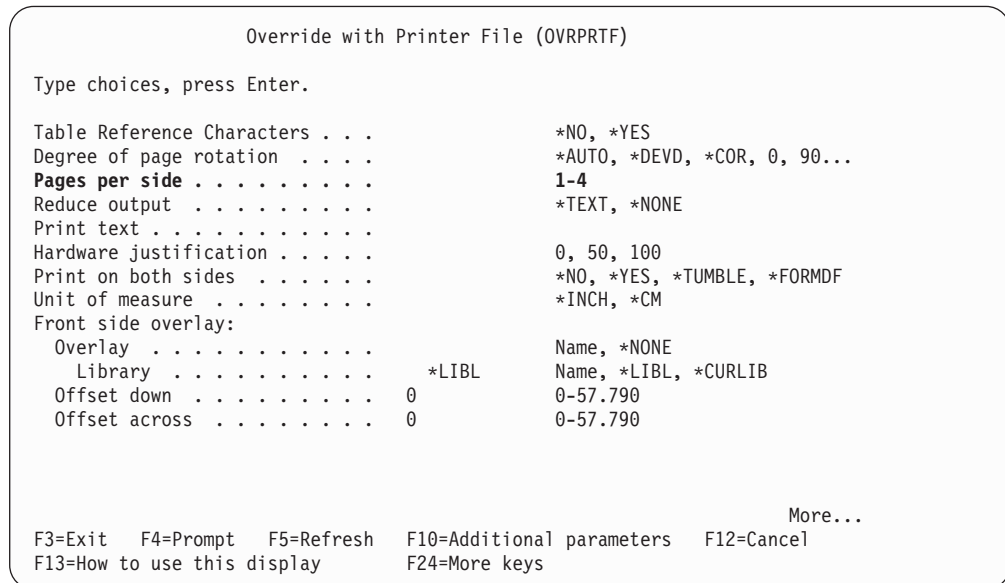


Figure 11. Override with Printer File Display - Pages Per Side

Specifying Computer Output Reduction

Computer output reduction (COR) automatically takes standard reports on computer paper and reduces them to fit on 8-1/2 by 11 inch paper. COR is designed to handle reports created at 66 lines per page and 132 print positions per line.

See Figure 12 on page 31 for an example of the COR function.


```

Override with Printer File (OVRPRTF)

Type choices, press Enter.

Table Reference Characters . . .      *NO, *YES
Degree of page rotation . . .      *AUTO, *DEVD, *COR, 0, 90...
Pages per side . . . . .            1-4
Reduce output . . . . .            *TEXT, *NONE
Print text . . . . .                0, 50, 100
Hardware justification . . . . .    *NO, *YES, *TUMBLE, *FORMDF
Print on both sides . . . . .      *INCH, *CM
Unit of measure . . . . .
Front side overlay:
  Overlay . . . . .                Name, *NONE
  Library . . . . .                Name, *LIBL, *CURLIB
  Offset down . . . . .            0      0-57.790
  Offset across . . . . .          0      0-57.790

More...
F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel
F13=How to use this display  F24=More keys

```

Figure 13. Override with Printer File Display - Reduce Output

Adding Overlays

One of the simplest ways to take advantage of AFP is to add overlays to an application with the printer file. This technique instantly transforms a preprinted form application into a completely electronic application. Specifying overlays with a printer file does not provide you with as many functions as those specified with DDS or with page definitions and form definitions, but for many output applications, this method is adequate.

Figure 14 on page 33 shows a “greenbar” overlay that can be used to improve the look and readability of a standard report. This kind of overlay is easy to create and is easily added using the front and back overlay parameters. Chapter 13, “Using Printer Files,” on page 121 shows a printer file being used to add an overlay to enhance the Super Sun Seeds output.

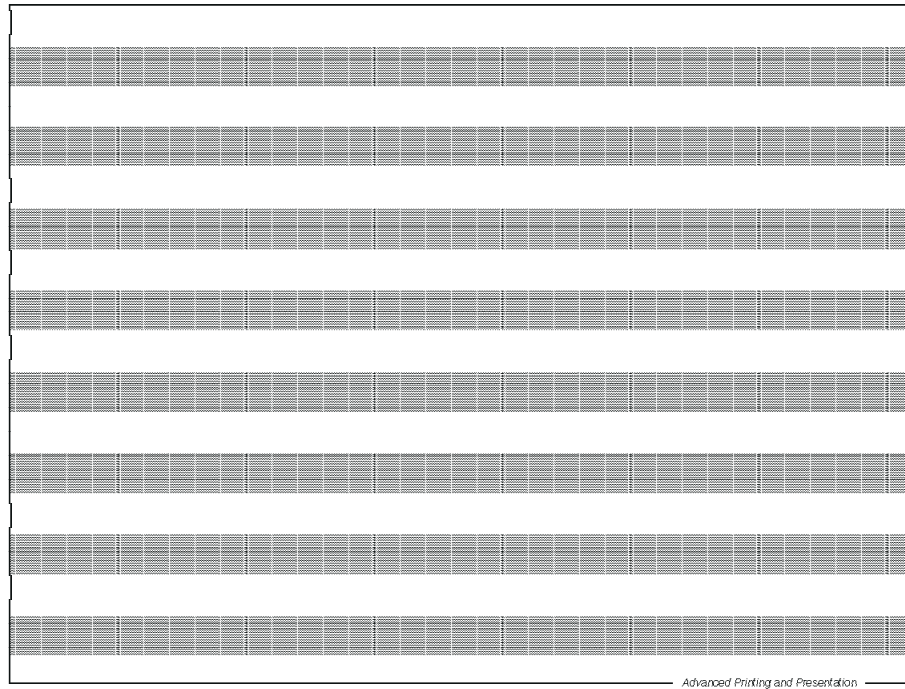


Figure 14. Greenbar Overlay

Use the **FRONTOVL** and **BACKOVL** parameters to add an overlay to every page of your output. The overlay parameters are shown in bold in the below displays:

Override with Printer File (OVRPRTF)

Type choices, press Enter.

Table Reference Characters . . .		*NO, *YES
Degree of page rotation		*AUTO, *DEVD, *COR, 0, 90...
Pages per side		1-4
Reduce output		*TEXT, *NONE
Print text		
Hardware justification		0, 50, 100
Print on both sides		*NO, *YES, *TUMBLE, *FORMDF
Unit of measure		*INCH, *CM
Front side overlay:		
Overlay		Name, *NONE
Library	*LIBL	Name, *LIBL, *CURLIB
Offset down	0	0-57.790
Offset across	0	0-57.790

More...

F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys

Override with Printer File (OVRPRTF)

Type choices, press Enter.

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

More...

F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys

Figure 15. Override with Printer File Display - Overlay Parameters

Converting Line Data to AFP

The Convert line data (CVTLINDTA) parameter lets you specify that spooled line data coming from an application (for example, reengineered applications using Infoprint Designer) is automatically written as an AFP spooled file. This lets you take advantage of key functions that are not supported with line data, such as viewing with the AFP Viewer or printing through Host Print Transform to PCL printers. The Convert Line Data keyword is new with Version 5.

To use this function, in the printer file specify CVTLINDTA(*YES), DEVTYPE(*LINE) or DEVTYPE(*AFPDSLINE), and you must specify a page definition with the PAGDFN parameter. The parameters are shown in bold in the below displays:

Override with Printer File (OVRPRTF)

Type choices, press Enter.

File being overridden		Name, *PRTF
Overriding to printer file . . .	*FILE	Name, *FILE
Library		Name, *LIBL, *CURLIB
Device:		
Printer		Name, *SYSVAL, *JOB
Printer device type		*SCS, *IPDS, *USERASCII...
Page size:		
Page length001-255.000
Page width001-378.000
Measurement method	*ROWCOL	*ROWCOL, *UOM
Lines per inch		3, 4, 6, 7.5, 7,5, 8, 9, 12
Characters per inch		5, 10, 12, 13.3, 13,3, 15...
Front margin:		
Offset down		0-57.790, *DEV D
Offset across		0-57.790

More...

F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys

Override with Printer File (OVRPRTF)

Type choices, press Enter.

Font character set:		
Character set		Name, *FONT
Library	*LIBL	Name, *LIBL, *CURLIB
Code page		Name
Library	*LIBL	Name, *LIBL, *CURLIB
Point size	*NONE	0.1-999.9, *NONE
Coded font:		
Coded font		Name, *FNTCHRSET
Library	*LIBL	Name, *LIBL, *CURLIB
Point size	*NONE	0.1-999.9, *NONE
Page definition		Name, *NONE
Library	*LIBL	Name, *LIBL, *CURLIB
Form definition		Name, *NONE, *DEV D
Library	*LIBL	Name, *LIBL, *CURLIB
AFP Characters		Name, *NONE
		+ for more values

More...

F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys

Override with Printer File (OVRPRTF)

Type choices, press Enter.

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

More...

F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys

Specifying Finishing Operations

Finishing operations are operations performed on a document as it is being printed and after it is printed. On printers that provide finishing support, Z-fold and nested finishing (stapling pages 6–10 together, for example) happen as the document is printed, stapling and punching the entire document occur after a document has been printed.

Check your printer's manual to determine the level of support provided for each finishing operation. When stapling, neither job nor file separators are stapled to their respective documents.

There are many ways to specify finishing, such as in a form definition, by using DDS keywords, and by printer file keywords. To specify finishing in the printer file, use the **CORNERSTPL**, **EDGESTITCH**, or **SADLSTITCH** parameters. These parameters are shown in bold in the below displays:

```

Override with Printer File (OVRPRTF)

Type choices, press Enter.

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

More...
F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys

```

```

Override with Printer File (OVRPRTF)

Type choices, press Enter.

Edge stitch:
  Reference edge . . . . . *NONE, *BOT, *RIGHT, *TOP...
  Reference edge offset . . . . . 0.0-57.790, *DEVD
  Number of staples . . . . . 1-122, *DEVD
  Staple offsets . . . . . 0.0-57.790, *DEVD
  + for more values
Saddle stitch:
  Reference edge . . . . . *NONE, *TOP, *LEFT, *DEVD
  Number of staples . . . . . 1-122, *DEVD
  Staple offsets . . . . . 0.0-57.790, *DEVD
  + for more values
  Font resolution for formatting . . . . . *DEVD, *SEARCH, 240, 300
  Defer write . . . . . *YES, *NO
  Spool the data . . . . . *YES, *NO
  Output queue . . . . . Name, *DEV, *JOB
  Library . . . . . *LIBL Name, *LIBL, *CURLIB

More...
F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys

```

Figure 17. Override with Printer File Display - Finishing Parameters

Chapter 5. Introduction to PSF Configuration Objects

A PSF configuration object is an extension of the printer device description. It lets you specify device configuration options that you cannot specify with the device description. For example, you can specify a library list for PSF to use to search for resources, when PSF should release a session with a printer, or whether to automatically take advantage of outline fonts. The PSF configuration object takes on a greatly expanded role with PDF and intelligent routing functions. You can use CL commands or the AFP Manager graphical interface to create and change a PSF configuration object.

Creating and Changing PSF Configuration Objects

The commands Create PSF Configuration (CRTPSFCFG), Change PSF Configuration (CHGPSFCFG), and Work with PSF Configuration (WRKPSFCFG) let you create, change, and work with PSF configuration objects. To create a PSF configuration object, enter the command CRTPSFCFG along with any parameters on the command line.

Example:

In this example we create a PSF configuration object named EMAILCFG that tells the Infoprint Server PDF subsystem to generate PDF and send it as e-mail. It accepts default values for all other parameters.

```
CRTPSFCFG PSFCFG(EMAILCFG) PDFGEN(*MAIL)
```

Alternatively, enter CRTPSFCFG and press F4 (Prompt), which brings up this screen:

To create the PSF configuration object, fill in the values and press Enter. For help

```
                                Create PSF Configuration (CRTPSFCFG)

Type choices, press Enter.

PSF configuration . . . . .
  Library . . . . .          *CURLIB      Name
User resource library list . . . *JOBLIBL  Name, *CURLIB
Device resource library list . . *DFT      *JOBLIBL, *CURLIB, *NONE
                                Name, *DFT
                                + for more values
IPDS pass through . . . . .    *NO       *NO, *YES
Activate release timer . . . . . *NORDYF   *NORDYF, *IMMED...
Release timer . . . . .        *NOMAX    1-1440, *NOMAX, *SEC15...
Restart timer . . . . .        *IMMED    1-1440, *IMMED
APPC and TCP/IP retry count . . 15         1-99, *NOMAX
Delay between APPC retries . . . 90         0-999
Acknowledgment frequency . . . . 100        1-32767
Printer response timer . . . . . *NOMAX    5-3600, *NOMAX
Generate PDF output . . . . .   *NONE     *NONE, *SPLF, *STMF, *MAIL
                                + for more values
```

Figure 18. Create PSF Configuration Screen

on any field, press F1.

For instructions about using AFP Manager to create or change a PSF configuration object, see “Creating a PSF Configuration Object” on page 271 and “Changing a PSF Configuration Object” on page 272.

Creating a PSF Configuration Object for the PDF Subsystem

Infoprint Server has a PDF subsystem that lets you transform any iSeries spooled file that PSF/400 can print to PDF. The PDF file can then be sent in an e-mail, put on an output queue, stored in the integrated file system, or any combination of those. To use the Infoprint Server PDF subsystem, you must specify something other than *NO for the PSF configuration object PDFGEN parameter, and then specify the PSF configuration object on the device that will be used for the transform, as described in “Using a PSF Configuration Object.”

You can also use a PSF configuration object to customize the PDF transform. For more information about using the PDF subsystem, including instructions about setting up the device, information about PSF configuration object parameters, and the steps required to use the PDF subsystem, refer to *Infoprint Server for iSeries: User's Guide*.

Using a PSF Configuration Object

To use a PSF configuration object, specify its name and location on the User Defined Object (USRDFNOBJ) parameter on the device description command (CRTDEVPRT or CHGDEVPRT) for an IPDS printer specified as AFP(*YES).

Example:

In this example, we configure the device named MYPRINTER to reference the PSF configuration object MYPSFCFG. MYPSFCFG is in the library MYLIB.

```
CHGDEVPRT DEVD(MYPRINTER) USRDFNOBJ(MYLIB/MYPSFCFG *PSFCFG)
```

Any device description for an IPDS printer specified as AFP(*YES) can use any PSF configuration object, one that is used by multiple devices or one that is unique to the device. PSF checks for a PSF configuration object when a writer is started.

Chapter 6. Introduction to AFP Resources

AFP documents consist of objects such as fonts, images in the form of page segments or overlays, and bar codes, illustrated in Figure 3 on page 13. Other objects, called page definitions and form definitions, are used by AFP documents to determine the layout of the output. All of these objects are called AFP resources. This chapter introduces these types of resources:

- “Fonts”
- “Images and Graphics” on page 61 (page segment resources)
- “Overlays” on page 65
- “Bar Codes” on page 67
- “Page Definitions and Form Definitions” on page 72

Fonts

To represent text, you need fonts. This section contains this information to help you better understand fonts and how they are used on an iSeries system:

- “What is a Font?” gives a general overview of fonts.
- “Font Architecture” on page 45 describes the AFP font objects in more detail.
- “Where Fonts are Stored” on page 49 describes where the font objects on your iSeries reside. It also suggests which font libraries to install.
- “How to Find Fonts on Your iSeries” on page 50 lists font object types and naming conventions to help you find fonts on your system.
- “How PSF Finds an AFP Font” on page 52 describes how PSF searches for a font when you request it in your print job.
- “Where to Get Fonts” on page 56 describes the iSeries font products and what is contained in each.

Much of the detail covered in this section relates to specifying fonts in a programming context (for example, within DDS). If you are using a graphical interface such as Infoprint Designer, then you might not have to directly work with font names and structures. If you need more information about fonts and font concepts, refer to *Printer Device Programming* and *IBM Infoprint Fonts: Font Summary*.

For examples of how to use fonts, see Chapter 7, “Working with Fonts,” on page 79.

What is a Font?

A *font* is a combination of these:

- A typeface or style of printable characters
- A group of characters for selection (described in “Font Architecture” on page 45)
- A map that determines which of the selected characters is to be printed or displayed (described in “Font Architecture” on page 45)

A font can include letters, numbers, punctuation marks and special characters. To add variety to your documents, you can include fonts of various sizes, styles, weights, and orientations. Figure 19 on page 42 shows different fonts used on the sample page.

400 CPU Parkway
Vegetation, NJ 55090

Office: 555-499-2367
Fax: 555-415-9794

LOS ARBOLES DEL MUNDO
32483 ARBOL LANE
MESA VERDE
IL 65478-9390

Customer Number: 141 Invoice Number: 31341 Invoice Date: 7/28/02 Payment Date: 8/28/02

Qty	UOM	Item #	Item Description	Price	Extension
1000	BX	56413213	POT POT	7.65	7,650.00
45	BZ	11005015	CHANTENAY SEEDS	2.19	98.55
900	EA	00001200	ARBOLES DEL SUR	45.00	40,500.00
98	PK	84512023	OREGON SPRING TOMATO SEED	.97	95.06
4	BX	11057893	AFRICAN DAISY, SEEDS	2.35	9.40
951	CT	11005011	LASSO RED SEEDS	892.23	48,510.73
46	DZ	11005014	SCARLET NANTES SEEDS	5.90	271.40
100	EA	31321655	SEMILLAS DEL SUS SOMBEROS	24.95	2,495.00

Total Due \$99,630.14

Return this tear-off strip with your payment. Payment is due by: 8/28/02

Make Checks Payable to: Super Sun Seeds
Amount Due is: \$99,630.14

LOS ARBOLES DEL MUNDO
32483 ARBOL LANE
MESA VERDE
IL 65478-9390

0001410099630146

Figure 19. Fonts on a Sample Page

There are many different ways to categorize a font, including how the font looks, its structure, and its location in the system. “Create a Document with Several Fonts” on page 79 gives an example of specifying different types of fonts with DDS.

These determine how a font looks:

Font Family

Defines a font’s overall “look”. Courier, Helvetica, Times Roman, and Sonoran Serif are examples of font families.

Typeface

A typeface defines the shape of the characters within a font family. It includes their weight, posture, and size or point size.

Weight

Defines the character weight such as medium or bold.

Posture

Defines the character stance, such as normal or italic.

Spacing

There are three different ways a font can be spaced:

Monospaced

Each character in a monospaced font has the same width and occupies the same print space. Monospaced fonts are also called *fixed-pitch* and *uniformly spaced* fonts. Monospaced fonts are typically used for tabular data because they are easily aligned in columns. Courier is a typical typeface that is normally monospaced.

Many printers have resident monospaced fonts. These can be raster or outline fonts.

Proportional

Proportionally spaced fonts contain characters with different widths. Narrow characters such as “i” and “l” print with less space than wider characters such as “m” and “A”. All proportionally spaced fonts use consistent widths for each character so an “i” in one proportional font is the same width as an “i” in another proportional font.

The average spacing of a proportionally spaced font is 12 characters per inch (cpi). Proportionally spaced fonts can be either raster or outline fonts.

Typographic

Typographic fonts have variable character widths, but unlike proportionally spaced fonts, each character in each font is individually sized for readability. Typographic fonts can be either raster or outline fonts.

Generally, typographic fonts are more easy to read. They also allow a greater number of characters to be printed on a page.

Size There are two different ways to specify a font’s size, depending on its spacing. For a monospaced font, it is specified in cpi. For a typographic font, size is measured in *point size*. Point size is a vertical measurement used to define the relative size of a font. This is measured from baseline to baseline for the entire font, thus, a “short” character, such as “a”, might have a smaller point size than that of the font. One point equals 1/72 of an inch.

A font’s **structure** can make a visual difference, but creates differences in many other ways as well. There are two different structures for a font; raster and outline.

Raster font

A raster font defines characters by bit patterns. Each bit pattern has a specific size, weight, style, and resolution (240 or 300 dpi). Figure 20 illustrates two font characters formed by raster patterns. The character on the right contains more pels, meaning that it has a higher resolution than the character on the left.



Figure 20. Raster Font Characters with Different Resolutions

A different font must be created for each character in each resolution, size, and style required. If we assume that each style and weight is available in point sizes 6 through 12, 14, 16, 20, 24, 30, and 36, then the capital letter A in the Helvetica font family, for example, is stored as 52 different bitmaps. The raster version of the Helvetica font family contains more than 400 different characters, each of which is represented by 52 bitmaps.

Outline font

An outline font (or *scalable font*) defines each character in terms of its outline, using mathematical formulas to define each character's shape. A rasterizer interprets the data and fills in the outline with dots. This creates a bitmap of the character for the specified point size and print resolution.

Outline fonts from IBM define characters using a 1000 x 1000 matrix. The outlines are size and resolution independent and thus can be scaled to any resolution and point size. They can also be specified in any orientation. Typically, you want a character's proportions maintained when you scale it. Sometimes, however, you might want to stretch or compress a character without maintaining the font's original proportions. This provides for short, fat characters, or for tall, skinny characters. See "Anamorphic Font Character Scaling" on page 84 for information.

No outline fonts come with the iSeries base operating system. The standard set of iSeries fonts is provided by IBM Infoprint Fonts (IBM product ID 5648-E77). See "AFP Font Collection" on page 57 for information.

Fonts can be presented in several different **orientations**.

Orientation

The orientation is the direction the font is presented on the page. All font character sets support four print orientations, 0, 90, 180, and 270 degrees. Other orientations can be printed if the text is created as image data or if outline fonts are used.

The iSeries supports two **locations for a font** to be stored: in the printer (printer-resident) or on the iSeries server (host-resident). Where your font is stored can determine when it is used and how you call it.

Printer-resident fonts

Printer-resident fonts permanently reside in the printer. These can be raster or outline fonts and vary by printer type and manufacturer. Refer to your printer's documentation (or print the resident fonts from the printer control panel) to determine what resident fonts it supports. You use Font Global Identifiers (FGIDs) to access these fonts. FGIDs are discussed in "Font Architecture" on page 45.

Host-resident fonts

AFP host-resident fonts reside on the OS/400 system in libraries, and they are downloaded to IPDS printers as needed. The font libraries on your system are listed in Appendix A, "iSeries Font Libraries," on page 293. Host-resident fonts are identified by a character set and code page or a coded font name (these objects are discussed in "Font Architecture" on page 45).

When a printer needs a host-resident font, PSF handles the download. Downloaded fonts ensure consistent fidelity of each document, regardless of what printer is used (or in the case of the Infoprint Server PDF subsystem, what viewer or output device is targeted). To avoid unnecessary downloads, you can enable *font capturing*, which lets the printer hold a host-resident font until that space is needed.

When you add a new font to the system (for example, by using the Type Transformer option of AFP Font Collection or Infoprint Fonts), any printer that supports downloadable fonts and matches the resolution of the font (if applicable) can use it immediately if requested by an application. Most current IBM printers support IPDS downloadable outline and raster fonts. Refer to your printer's documentation to determine whether your printer supports downloadable fonts.

Font Architecture

To fully understand fonts and how to use them, you need to understand their underlying architecture. Each font technology has its own architecture. We will introduce two different font technologies, then describe the iSeries font architecture.

Font Technologies

Adobe Type 1 Fonts

Type 1 outline is the font format commonly used with ASCII output data streams (PCL, PostScript, and PDF). It is also used with IBM Type Transformer for SBCS fonts. This format includes outlines of the various type families, which can be transformed for use by advanced function printers in sizes from 1-999 points (AFP outline fonts) or from 1-72 points (raster fonts). All Adobe Type 1 fonts (for single-byte encoding) are outline. Type 1 fonts are available from many font vendors.

AFP Fonts

AFP fonts are the standard fonts used for IPDS output. AFP fonts are made up of a code page and a character set, which are described in greater detail below. All AFP fonts are encoded as either single-byte or double-byte fonts, depending on the language for which they are being used.

Encoding Schemes on the iSeries

There are two main ways characters on the keyboard are mapped to the characters in your output, using a single-byte or double-byte system. Multinational applications need to be consider both the encoding used to create the document and the encoding that will be used to print the document.

SBCS (Single-byte character set)

An encoding scheme where a character is identified with a code page that maps the character on your keyboard to the character that is shown in the output. This encoding is used for languages with alphabetic writing systems, such as English, Greek, and Arabic. Code pages are described below.

DBCS (Double-byte character set)

An encoding scheme where a character is identified with a code page that uses two bytes of data to map the characters on your keyboard to the character that is shown in the output. Languages with non-alphabetic writing systems, such as Chinese, Japanese, and Korean, require DBCS encoding.

AFP Font Architecture

In order to use AFP fonts on the iSeries, you need to understand the AFP font architecture in detail. AFP fonts are encoded in EBCDIC format, and are either SBCS or DBCS. We will first focus on host-resident fonts.

Host-resident fonts are represented by a *character set* and *code page* pair or a *coded font* name. They must be downloaded to the printer device before they can be used. These are the types of font objects on the host:

Character set

The set of all the characters in a given font. An IBM EBCDIC character set contains up to 196 characters or symbols. It is an object on the iSeries and has a name. For example, C0D0GT10 is the character set for Gothic Text 10 pitch. Downloaded character sets and code pages are always used together in pairs. They identify host-resident fonts.

The iSeries object type for a character set is *FNTRSC and the attribute is FNTCHRSET. See “OS/400 Font Products” on page 57 to determine what character sets are available. For information about font naming conventions, refer to *IBM Infoprint Fonts: Font Summary*.

Code page

A code page is a map that equates a code point (the key pressed) to a single character in the character set. It is an object on the iSeries and has a name. Using the name (and not a number) implies the object is host-resident. For example, T1V10037 is code page 37 for USA/Canada English. Downloaded character sets and code pages are always used together in pairs. They identify host-resident fonts.

For the different code pages supported on the iSeries system, refer to *National Language Support Planning Guide*. For more information about code pages and a list of code pages provided with the IBM AFP Font Collection, refer to *IBM AFP Fonts: Font Summary for AFP Font Collection*. The iSeries object type for a code page is *FNTRSC and the attribute is CDEPAG. See “OS/400 Font Products” on page 57 to determine which code pages are available. For information about font naming conventions, refer to *IBM Infoprint Fonts: Font Summary*.

Coded font

A specific code page / character set pair. Coded fonts are host-resident.

To put these concepts together, a code page maps a given code point (or keyboard key) to a specific character within the group of characters available (character set). This is what happens as you enter your text at a workstation:

1. Each keyboard character is translated into a *code point* (in single-byte encoding, a code point is a one-byte hexadecimal value, such as X'C1' to represent the letter A).
2. When the text is printed, each code point is matched to a character ID on the code page you specify.
3. The character ID is then matched to the image of the character in the character set you specify. The image in the character set is the image that is printed in your text.

This structure supports the requirements of different national languages to print different patterns based on the same code point. This is illustrated in Figure 21 on page 47.

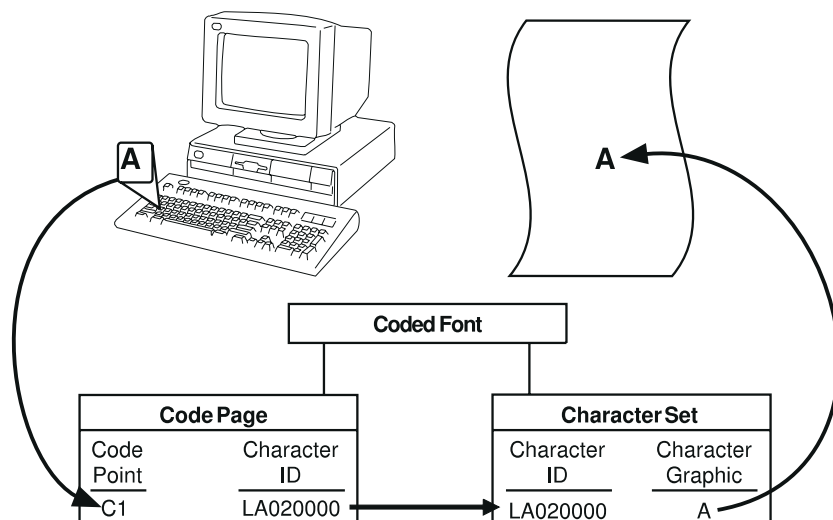


Figure 21. Translation of a Keyboard Character into a Printed Character

The iSeries Font Architecture on International Systems

The same character set (of up to 196 or 256 unique characters) can be used with different code pages to make document creation easier. Those creating documents in Spanish, for example, prefer the 'Ñ' and 'ñ' to be mapped to easily-accessible keys, while those creating documents in another language might want easy access to characters such as 'é', 'ç', 'ç', and others. Use of a code page makes these mapping changes possible. For example, if you press a key and the program you are using translates it to X'4F', you get a different result in your output depending on the code page being used. If you are using EBCDIC code page T1V10037, you get "¡". If you are using EBCDIC code page T1V10500, you get "Ñ", and so on. This is illustrated in Figure 22 on page 48. Table 5 lists some code point translations. "Create a Document with Several Fonts" on page 79 gives an example of specifying different code pages with DDS.

Table 5. Code Point Translations

EBCDIC Code Page ID	Country or Region	Code Point	Result
37	USA/Canada Country extended code page (CECP)	X'43'	ä
		X'4F'	
		X'69'	Ñ
		X'9F'	¤
273	Germany F.R./Austria - CECP	X'43'	{
		X'4F'	!
		X'69'	Ñ
		X'9F'	¤
284	Spain/Latin America - CECP	X'43'	ä
		X'4F'	
		X'69'	#
		X'9F'	¤
500	Multinational #5	X'43'	ä

Table 5. Code Point Translations (continued)

EBCDIC Code Page ID	Country or Region	Code Point	Result
		X'4F'	!
		X'69'	Ñ
		X'9F'	α
1141	Germany/Austria - Euro country extended code page (ECECP)	X'43'	{
		X'4F'	!
		X'69'	Ñ
		X'9F'	€

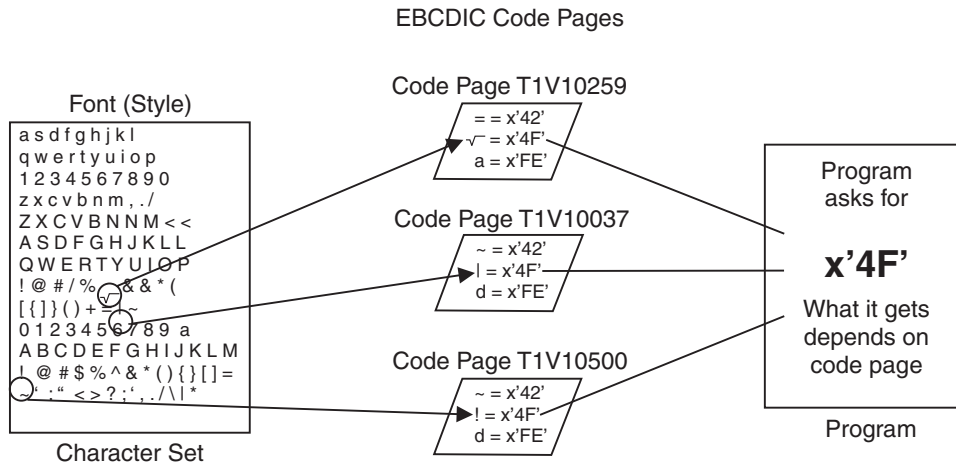


Figure 22. How Code Pages Work

Your system will have a default code page specified, specific to the country or region to which the machine is shipped. To see what your default code page is, enter the command `DSPSYSVAL SYSVAL(QCHRID)`.

In double-byte languages, the map from keyboard to output is somewhat more complicated than with single-byte languages. For outline fonts, a double-byte code page is treated as several single-byte code pages concatenated together and each character has a double-byte code point. For raster fonts, double-byte code pages are treated as a collection of single-byte code pages. The coded font is divided into sections, each with its own single-byte code page. Each character in the section has a single-byte code point.

Printer-resident fonts

Printer-resident AFP fonts are represented by a *Font Global ID (FGID)*. FGIDs are whole numbers from 1 to 65534 registered within IBM and used to identify font resources. FGIDs are assigned to unique type family/typeface combinations, such as Courier Roman Medium (FGID 11). The printer-resident code page used with an FGID is normally defined by the system default. In the USA, code page 37 is the iSeries default. To determine what your default code page is, enter the command `DSPSYSVAL SYSVAL(QCHRID)`. You can find a complete list of family names, typefaces, attributes, and size of each FGID in *Printer Device Programming*. Most laser printers have an option that lets a user print a list of the FGIDs stored in the printer.

In order to ensure document fidelity across various printers, you need to use the same fonts on each printer. If a font is specified in a document but is not resident in the target printer, a substitute font might be used. This can cause output to look different when printed on different printers. See “Font Substitution” on page 53 for more information.

Fonts found in IBM’s AFCCU™ printers are the same as the host-resident fonts provided with AFP Font Collection or Infoprint Fonts. This means that printer-resident fonts and downloaded raster fonts can be interchanged in a print environment where both outline font and bitmap printers exist.

Where Fonts are Stored

Fonts on the iSeries are stored in font libraries that PSF or another printer driver accesses when it needs a font. When you add fonts to your iSeries from Infoprint Fonts or another product, you need to put the fonts in a place where the printer driver can find them.

Which Font Libraries Should I Install?

The Font Collection CD contains a large number of font libraries that include character sets, code pages and coded fonts for a wide variety of languages and applications. For most users, only a few of those font libraries are normally required. You must decide which font libraries best meet your needs, however, you might want to choose libraries from these commonly used font libraries:

Table 6. Commonly Used Font Collection Font Libraries

Font Collection CD File Name	Description	Usual OS/400 Library Name
CPL300	300-peI version of compatibility fonts found in QFNTCPL (which are 240-peI).	QFNT300CPL
LA1300	300-peI version of Latin 1 Fonts.	QFNT300LA1
SYM300	300-peI Symbols character sets. Install this if you have applications that use special symbols.	QFNT300SYM
CDEPAG	Additional code pages not found in QFNTCPL. This library is important if you use the AFP Driver for Windows to create overlays.	QFNTCDEPAG
LA10LN	Outline font version of LA1300.	QFNTOLNLA1
SYM0LN	Outline font version of SYM300.	QFNTOLNSYM

Libraries QFNTCPL and QFNTxx (where xx is 01 to 19 and 61 to 69) are, at print time, automatically included in the active library list. This means that you can store fonts that are needed at print time in the QFNTxx libraries instead of adding their library name to the user or system library list. Because of this, you might want to, for example, change library QFNT300CPL to, QFNT01 (assuming that QFNT01 does not already exist on your system).

See “Steps to Restore Font Libraries From the Font Collection CD to the iSeries” on page 84 for instructions to restore these font libraries to your iSeries.

Helpful Font Tools

There is a free tool available from IBM that you can download to help you load the AFP Font Collection fonts onto your iSeries and print a font catalog. Refer to this Web page for the tool and instructions:

<http://www.printers.ibm.com/r5psc.nsf/web/rdfont01>.

You can look at the fonts supplied in Infoprint Fonts and AFP Font Collection on the Printing Systems Information Center. You can search for a specific font by language family or code page, or you can browse the code pages. The Printing Systems Information Center is at <http://publib.boulder.ibm.com/printer/psindex.htm>.

How to Find Fonts on Your iSeries

Font objects on the iSeries are stored in libraries and have their own object types. Knowing these will help you locate the fonts on your system.

For **code pages**, the object type is *FNTRSC and the attribute is CDEPAG.

For **coded fonts**, the object type is *FNTRSC and the attribute is CDEFNT.

For **font character sets**, the object type is *FNTRSC and the attribute is FNTCHRSET.

Code Page Naming Convention

Like character sets, code pages are named in a number of ways. One way is with a code page name. These code pages are downloaded to the printer. The code page name can be up to 8 characters long. Code page names are used with character set names for printing on printers with host-resident fonts, such as the Infoprint 2000, 2085, 2105, 4000, and 4100.

Another way is with a code page global identifier (CPGID). CPGIDs are printer-resident code pages and have numbers for names (for example, 259 or 500). Generally, printers with resident fonts use CPGIDs to name the printer-resident code pages. CPGIDs are also used within CHRIDs. In this case, the structure is T1Vnmmmm: where:

T1 Designates a code page.

Vn Designates the version of the code page, such as version 1.

mmmm

Designates the 4-character page name, number, or category. For example, 0500.

Coded Font Naming Conventions

Unlike other font components, coded font names are generally shortened by excluding the origin and reserved characters (the first two characters of their name). This is necessary because some AFP licensed programs accept only 6 characters for coded font names. However, some applications can use coded fonts named with 6 or 8 characters.

Coded font names on the iSeries server are 6 or 8 characters long. Each character or group of characters tells something about the coded font. The structure is Xcftpp, where:

Xc Xc means that this object is a coded font. XZ means that this object is an outline coded font. Otherwise, the first two characters are X0.

f Indicates the type family.

t Indicates the typeface.

pp Indicates the characters per inch for uniformly spaced and mixed-pitch fonts.

For example, the coded font name X0GT10, is for a Gothic text style font and the characters are 10 characters per inch.

To find out which font character set and code page make up a coded font name, use the Work with Font Resources (WRKFNTRSC) command. This command allows you to specify the font resource to be worked with, the library it is in, and the attribute (coded font).

Additional naming conventions have been adopted to more explicitly name the code page used with a character set.

Font Character Set Naming Convention

Font character sets are the font resources of which you will most likely need to know the name. Most of the OS/400 font character sets use a common resource naming convention. Understanding these will help you know what fonts to use in your document. For raster font character sets, the format is:

C0fs0cpp where:

C0 Designates a raster character set.

f Designates the font family. For example, N is Times New Roman and H is Helvetica.

s Designates the font style as follows:

- 2: Roman Medium
- 3: Italic Medium
- 4: Roman Bold
- 5: Italic Bold

c Designates the language complement. For example, 0 is Latin 1 and 1 is Symbols.

pp Designates the point size as follows:

60: 6-point	00: 10-point	F0: 16-point	T0: 30-point
70: 7-point	A0: 11-point	H0: 18-point	Z0: 36-point
80: 8-point	B0: 12-point	J0: 20-point	
90: 9-point	D0: 14-point	N0: 24-point	

For outline font character sets, the naming structure is much simpler. The structure is CZfss where:

CZ Designates an outline font.

f Designates the typeface, such as Helvetica

ss Designates the typeface, such as Roman Bold

An example of the name of an outline font character set is CZH200, which is Helvetica Roman Medium.

For more information about font names, refer to *IBM Infoprint Fonts: Font Summary* or the iSeries output Web page: www.ibm.com/eserver/iseries/printing.

How PSF Finds an AFP Font

Because there are different places a font can be stored, PSF needs a method to find and select a font to use. This is how PSF searches for a requested font when it is referenced without a library qualifier:

- If the font requested is printer-resident (the application requests the font by FGID) PSF looks in the printer for the font. If it is not found, PSF might use a different printer-resident font. When PSF uses a different font than requested, it is called *font substitution*.
- If the font requested is host-resident (the application requests the font by character set and code page or coded font), PSF looks on the iSeries for it. If it is found, PSF downloads it if it has not already been downloaded to the printer. While outline fonts have a unique name (CZxxxx), 240- and 300-pel raster fonts have the same name but are stored in different libraries. OS/400 will select the correct font. These are the libraries in which PSF looks for a requested font:

1. Libraries in the user's library list, specified on the PSF configuration object. The libraries searched vary, depending on the value specified:

***JOBLIBL**

The job library list in effect at the time the spooled file was created is searched. The job library list is the System Library list followed by the User Library list.

***CURLIB**

The current library is searched. If no current library has been defined, QGPL is searched.

***NONE**

Only the device resource library list is used.

2. Libraries in the device resource library list, specified on the PSF configuration object. The libraries searched vary, depending on the value specified:

***DFT** The system font libraries, as listed below are searched:

- QFNTCPL: This library is shipped with the operating system. It contains the 240-pel AFP Compatibility Fonts.
- QFNT01-QFNT19
- QFNT61-QFNT69

library-name

Up to 30 library names can be listed.

3. If a PSF configuration object has not been created for the selected device, step 1 is followed, with a value of *JOBLIBL, then step 2, with a value of *DFT.

For information about the library lists and how to specify them, see "How PSF Searches for Resources" on page 357.

- If the font is not found, a substitute font might be used. For a description of the cases when a substitute font is used, see "Font Substitution" on page 53.

It is generally a better idea to use *LIBL when specifying a font library to search than specify one individually, because when you specify an individual library, you limit your fonts to a single resolution.

If you have certain fonts that you want to use for a specific device, for example, if you want to use 300 dpi fonts with one printer and 240 dpi fonts with another, specify the appropriate libraries in a device resource library list. The device resource library list is specified on the PSF configuration object associated with the device. Alternatively, because of the way the system looks for fonts, you can store fonts that are needed at print time in the QFNTxx libraries and they will be found. You could, for example, change library QFNT300CPL to QFNT01, as long as QFNT01 does not already exist on your system. To rename that library, use this command:

```
RNMOBJ OBJ(QFNT300CPL) OBJTYPE(*LIB) NEWOBJ(QFNT01)
```

Note: You must have object management authority to the library you are renaming.

For more information about OS/400 libraries, see “How PSF Searches for Resources” on page 357.

Captured Fonts

A printer that supports font capture can dynamically “capture”, or cache, a host-downloaded font. The captured font then acts like a printer-resident font, even when the printer has been powered off. Network traffic is reduced by eliminating subsequent font downloads to the printer.

Notes:

1. Because improper use of font capturing can cause unpredictable results, only system administrators should handle the font capture feature.
2. Even after a font is captured, the host font must still remain on the system because the printer might discard fonts to free up space, and the font might need to be reloaded. In addition, users must still have authority to the font object on the host in order to use it.

Refer to *Printer Device Programming* for information about the proper use of font capturing. For instructions to enable font capture, see “Capturing Fonts” on page 83.

Font Substitution

OS/400 tries to print your file even if it does not have the font you requested. It substitutes a font that will produce similar, if not exact, results. However, if you specify FIDELITY(*ABSOLUTE) on the printer file, the match for font substitution must be exact or the job will not print. To allow font substitution, specify FIDELITY(*CONTENT).

If the requested font is printer-resident, a substitute font is used in these situations:

- If the printer supports resident fonts but does not contain the requested font, a substitute printer-resident font is used. For information about how the substitute font is selected, see the information about the Printer-resident font to printer-resident font mapping table in “Font Mapping Tables” on page 55.
- If the printer does not support resident fonts, a host-resident font is downloaded and used. For example, if Letter Gothic (FGID 87) is requested, but not resident, a host font from the font substitution tables (such as the Letter Gothic bitmap character set COS0LR12) is selected and downloaded to the printer.

If the requested font is a host font, a substitute font is used in these situations:

- If the printer does not support downloaded fonts, such as impact printers, and a matching printer-resident font exists, the matching printer-resident font is used.

- If the printer does not support downloaded fonts and an exactly matching printer-resident font does not exist, a printer-resident font that matches the requested font as closely as possible is substituted.
- If the printer supports downloaded fonts but the font cannot be found by searching the resource library list or is in the wrong resolution, the printer tries to substitute a printer-resident font. If there is not an exact match, the print job is held.

No substitution of coded fonts takes place on the iSeries server. If the coded font is not available, the document will not print. However, you can use these parameters on the PSF configuration object to specify font substitution in certain circumstances:

MAPIGCFNT(*YES)

Specifies that DBCS simulation fonts are used instead of the DBCS raster fonts specified in the data stream. DBCS simulation fonts are outline fonts that are positioned like raster fonts. This lets you use outline fonts to print applications that use DBCS raster fonts without changing the application or the appearance of the printed output.

USEOUTLFNT(*YES)

Specifies that downloadable AFP raster fonts (named X0nnnnnn) are replaced with equivalent downloadable outline fonts (named XZnnnnnn) if they exist. If the XZnnnnnn coded font is found, it is used, if it is not found the X0nnnnnn coded font is used.

Example 1:

Assume these conditions:

- Your application calls for a font character set (FNTCHRSET specified on the printer file), for example, C0S0CR10 for Courier Roman medium 10 pitch.
- The printer is a 4224 and has resident fonts identified by FGIDs.

FGID 11 will be substituted for C0S0CR10 and sent to the printer. In this example, the system substitutes a font that is resident on that printer.

Example 2:

Assume these conditions:

- Your application calls for a font (specified on the FONT parameter of the printer file). The font specified is font 26 (Gothic Matrix, Roman medium 10 pitch) and the printer is a 3812.
- You decide to print the document on a 4019 printer. Font 26 is not supported on the 4019.

In this example, the system substitutes font 11 (Courier, Roman medium 10 pitch). You can figure out such substitutions by looking at the font substitution table in *Printer Device Programming*.

Example 3:

Assume these conditions:

- Your application uses a font (specified on the FONT parameter of the printer file). The font specified is font 40 (Gothic, Roman medium 10 pitch).
- The printer you are going to print on supports only font character sets (for example, a 3827).

In this example, the system substitutes font character set C0D0GT10 (Gothic Text, Roman medium 10 pitch). You can figure out such substitutions by looking at the font substitution table in *Printer Device Programming*.

You can suppress messages regarding *successful* font substitutions. Messages indicating that a font substitution attempt failed are always issued. To suppress messages regarding successful font substitution, specify FNTSUBMSG(*NO) on the PSF configuration object used by the device to which you are printing. Alternatively, you can change it using the AFP Manager component of iSeries Navigator by de-selecting **Notify after successful font substitution** on the PSF Configuration **Resources** page. For detailed instructions about using iSeries Navigator to work with PSF configuration objects, see “Changing a PSF Configuration Object” on page 272.

Font Mapping Tables: Font mapping tables control how the fonts are substituted, when necessary. PSF comes with system versions of all of the host-to-printer and printer-to-host font mapping tables. However, you can also create your own font mapping tables. If you do create font mapping tables, PSF searches your tables for a font substitution before the system tables. You can create these mapping tables:

- Host-resident font character set to printer-resident font character set (HPFCS)
This table is named QHPFCS and is in library QUSRSYS. This table is used when your application references host-resident fonts (font character sets and code pages) and the printer, such as the 4224, 4234, 4230, 64XX, does not support downloading of host-resident fonts. PSF maps the references from host-resident fonts to printer-resident fonts.
- Host-resident code page to printer-resident code page (HPCP)
This table is named QHPCP and is in library QUSRSYS. This table is like the HPFCS table, in that it is used when the application references host-resident code pages and the printer being used does not support host-resident code pages. PSF maps the host-resident code page to a printer-resident code page and downloads to the printer (if necessary).
- Printer-resident font character set to Host-resident font character set (PHFCS)
This table is named QPHFCS and is in library QUSRSYS. This table is used when your application references printer-resident fonts and the printer, such as the 3827, 3825, 3820, 3900 Model 1, does not support resident fonts. PSF maps the references from printer-resident fonts to host-resident fonts and downloads them (if necessary).
- Printer-resident code page to Host-resident code page (PHCP)
This table is named QPHCP and is created in library QUSRSYS. This table is like the PHFCS table, in that it is used when the application references printer-resident code pages and the printer being used does not support them. PSF maps the printer-resident code page to a host-resident code page and downloads it to the printer (if necessary).
- Printer-resident font to printer-resident font
For the printer-resident to printer-resident font mapping table, the following processing is done by the system:
 - If the printer-resident font specified in the print job is supported by the printer, then it is used. The printer-resident to printer-resident font mapping table is not searched.
 - If the printer-resident font specified in the print job is not supported by the printer, then the printer-resident to printer-resident font substitution table is searched.

- If a matching entry is found in the printer-resident font mapping table and the entry is supported by the printer, then the specified substitute font in the printer-resident font mapping table is used.
- If a matching entry is not found in the printer-resident font substitution table or if the specified substitute font is not supported by the printer, then the system will use its internal font substitution tables to perform the font substitution.

Notes for the printer-resident font mapping table:

1. This font substitution table controls font mapping by printer type group, not by individual printer type.
2. You can only make a substitution for fonts (FGIDs) that are not supported by the printer group. For example, if the printer group has FGID 11, you cannot map it to something else.

Printer Device Programming lists the system font mapping tables. You can also view the system font mapping tables with the OS/400 DSPFNNTBL (Display font table) command or AFP Manager.

“Working with Font Mapping Tables” on page 82 has commands to use with font mapping tables.

Where to Get Fonts

You can get fonts from any of these places:

- Fonts can come with your printer.
- You can get them from a font product and install them on your iSeries server, see “OS/400 Font Products” on page 57.
- You can use a conversion product to convert a font to AFP format, see “Type Transformer and Utilities for Windows” on page 59.

Fonts Resident on Your Printer

To determine what fonts are resident on your printer, refer to your printer’s documentation or print the resident fonts from the printer control panel.

Example:

The IBM Infoprint 2085 lists its resident font character sets and code pages (double- and single-byte) by FGID, GCSGID, and typeface. This printer has Times New Roman, Helvetica, Courier, ITC Boutros, Boutros, Narkissim, Shalom, and other typefaces for various code pages. This is part of the font list:

Table 7. IBM Infoprint 2085 Fonts (Partial List)

FGID	GCSGID	Typefaces - Latin 1/2/3/4/5 with Euro
2308	1355	Times New Roman Medium
2309	1355	Times New Roman Bold
2310	1355	Times New Roman Italic Medium
2311	1355	Times New Roman Italic Bold
2304	1355	Helvetica Roman Medium
2305	1355	Helvetica Roman Bold

OS/400 Font Products

The font products described in this section provide host-resident fonts. On the iSeries, these sets of fonts are included in the base system:

- AFP Compatibility Fonts, see page 60.
- GDDM Fonts, see page 60.

These are the standard set of business fonts for iSeries:

- Infoprint Fonts for Multiplatforms (Program number 5648-E77)
- AFP Font Collection (Program number 5648-B45)

Infoprint Fonts supersedes AFP Font Collection but the sets of fonts are very similar.

Infoprint Fonts for Multiplatforms: This set of standard fonts is very similar to AFP Font Collection. However, it does not contain any raster fonts (240 dpi or 300 dpi). It contains some globalization enhancements that affect double-byte and Euro countries. It contains enhancements for these languages: Arabic, Cyrillic/Greek, Latin, and Thai. Additionally, these DBCS font libraries were enhanced: Japanese, Korean, Simplified Chinese, and Traditional Chinese. In general, if you already have AFP Font Collection, you do not need to replace it with Infoprint Fonts unless you need the globalization features, Font Installer for AFP Print Servers or WorldType Fonts for AFP Print Servers. Infoprint Fonts for Multiplatforms is made up of these font groups and features:

- IBM Expanded Core Fonts (AFP outline fonts)
- Font Installer for AFP Systems – orderable feature required to upload TrueType fonts or OpenType fonts onto your system
- WorldType Fonts for AFP Print Servers – orderable feature that supplies TrueType and OpenType fonts
- DBCS Core Fonts and DBCS Simulation Fonts – orderable features
- Type Transformer and Utilities for Windows – orderable feature

The font groups and Type Transformer and Utilities are described in the following sections. You do not typically need to install every font library on your OS/400. “Which Font Libraries Should I Install?” on page 49 gives suggestions about which font libraries to install. For more information about Infoprint Fonts for Multiplatforms, refer to *IBM Infoprint Fonts: Font Summary*.

AFP Font Collection: IBM AFP Font Collection for Workstations and OS/400, provides font resources and utilities for the OS/400 operating environment. The Font Collection is made up of these font groups and features:

- IBM Expanded Core Fonts (240-pel, 300-pel, and AFP outline fonts)
- AFP Compatibility Fonts (240-pel and 300-pel)
- DBCS Core Fonts and DBCS Simulation Fonts – orderable features
- Type Transformer and Utilities for Windows – orderable feature

These fonts and features are described in the following sections. You do not typically need to install every font library on your OS/400. “Which Font Libraries Should I Install?” on page 49 gives suggestions about which font libraries to install. For more information about the AFP Font Collection, refer to *IBM AFP Fonts: Font Summary for AFP Font Collection*.

IBM Expanded Core Fonts: The single-byte type families provided in the AFP Font Collection and Infoprint Fonts are called the Expanded Core Fonts and are

available as 240-pel, 300-pel, and AFP outline fonts¹. The IBM Expanded Core Fonts are all derived from Type 1 font technology. These font families and language groups are included:

- Boldface Latin1 (with euro)
- BookMaster[®] Latin1 and Specials (with euro)
- Courier (with euro)
- Courier APL2[®]
- Gothic Katakana
- Gothic Text Latin1 (with euro)
- Helvetica (with euro)
- IBM Logo (for BookMaster)
- Letter Gothic Latin1 (with euro)
- OCR-A
- OCR-B
- Prestige Latin1 (with euro)
- Times New Roman (with euro)

DBCS Core Fonts and DBCS Simulation Fonts: The DBCS Core fonts and the DBCS Simulation Fonts are provided as a feature of Infoprint Fonts and AFP Font Collection. DBCS Simulation fonts are outline fonts that look like a raster font product (listed in Table 9 on page 59. For more information about these DBCS fonts, see the appropriate technical reference for the DBCS fonts (for example, *IBM Infoprint Fonts: Japanese Font Library Technical Reference*, S544-5849-01).

Each of these DBCS type families and languages are provided as a separate feature. When you buy a language feature you get the DBCS core fonts and the DBCS simulation fonts for that language.

Table 8. DBCS Font Families Available

Language	Font families
Japanese	Heisei Kaku Gothic
	Heisei Maru Gothic
	Heisei Mincho
Korean	Gothic
	Myengjo
Simplified Chinese	Fang Song (GB)
	Hei (GB 108030)
	Kai (GB)
	Song (GB 108030)
Traditional Chinese	Kai
	Sung

1. Infoprint Fonts does not contain raster (240-pel or 300-pel) fonts.

These are the raster font products simulated by the DBCS Simulation Fonts

Table 9. Font Products Simulated by DBCS Simulation Fonts

Language	Product simulated
Japanese	AFP Japanese Font V2 (5771-AGB)
	AFP Japanese Heisei Font (5648-104)
Korean	AFP Korean Font (5771-AFW)
Simplified Chinese	AFP Simplified Chinese Font (5771-AEK)
Traditional Chinese	AFP Traditional Chinese Font (5771-AFZ)

Type Transformer and Utilities for Windows: This optional feature of the Infoprint Fonts and AFP Font Collection lets you convert any TrueType, Type 1 or CID font typeface into an AFP font. It also lets you edit outline fonts, create and modify code pages, and create and modify coded fonts.

Specifically, you can use the Type Transformer component of AFP Font Collection for Workstations and OS/400 to do these conversions:

- Adobe Type 1 fonts to AFP outline
- Adobe Type 1 fonts to 240, 300, and 600 pel AFP raster
- Adobe TrueType Fonts to 240 pel AFP raster
- Adobe TrueType fonts to Adobe Type 1

To support these capabilities in the Windows environment, Infoprint Fonts and AFP Font Collection include the same AFP fonts that are available for IBM operating systems in a Windows NT format. The transformers, utilities, and tools are summarized in Table 10. For information about using this feature, refer to *IBM Infoprint Fonts: Introduction to Type Transformer and Utilities for Windows* or *AFP Font Collection: Type Transformer for Windows User's Guide*.

Table 10. Transformers, Utilities, and Tools

Product Name	Description
AFP Code Page Editor	The Code Page Editor has a graphical interface that lets you convert an AFP code page resource into a text-readable file. You can then use any text editor to add, change, or delete code point assignments, character identifier assignments, and global code point control information. The edited file is then rebuilt into a code page resource.
AFP Coded Font Editor	The Coded Font Editor has a graphical interface that lets you convert an AFP coded font resource into a text readable file. You can then use any text editor to add, change, or delete character set and code page assignments. For coded fonts that name AFP outline character sets, you can also change the point size. The edited file is then rebuilt into an AFP coded font resource.
AFP Font to Flat File	Lets you build a flat file from a font object file.
AFP2FON	Command-driven program that displays double byte characters.
ATMSHOW	Converts AFP raster format UDC to Windows 3.1 UDC format. By importing the output font from a Windows EUDC editor, you can use it as Windows EUDC (end user defined characters).
CID2EPS	Generates an encapsulated PostScript (EPS) file of a character from a selected CID-keyed font. Use it to extract characters from a base font to use in creating UDCs.

Table 10. Transformers, Utilities, and Tools (continued)

Product Name	Description
DBCS Type Transformer	Transforms CID-keyed outline fonts into AFP 240-dpi raster and AFP outline font formats. It has a graphical interface to help build DBCS fonts, and a Job Manager to control job processing.
DUVRMARK	Changes the date and time stamp or the capture setting of a font object.
FontLab	A digital typeface editor with a graphical interface that lets you create or edit Type 1 and CID outline fonts. It includes native editing tools for Multiple Master and TrueType fonts plus extensive font-manipulation features.
FontLab Composer*	An editor for fonts with very large character sets. It can edit CID fonts, convert double-byte TrueType fonts into CID font format and includes extensive support for multiple encodings.
OS/400 Uploader	Enables you to store transformed SBCS fonts on the OS/400. Other font objects, like DBCS fonts, code pages, or coded fonts, must be stored in different ways.
SBCS Type Transformer	Transforms Adobe TrueType and Type 1 fonts to AFP raster and outline fonts. As part of the transform process, you can be selective in choosing which characters are to be included in the output fonts so that font storage requirements can be reduced. For specific information about the transforms, see "Type Transformer and Utilities for Windows" on page 59.
ScanFont	A paper-to-font application. It can be used to transform printed or painted images of characters into Type 1 or TrueType fonts. It includes outline editing tools, kerning and hinting editors.
SigMaker*	A program that lets you insert your signature, company logo or special symbol into any TrueType font.
TypeTool*	An entry-level font editor. It includes outline font editing tools and a kerning module. It can import and export fonts in Type 1 or TrueType format.
* Not available with Infoprint Fonts.	

AFP Compatibility Fonts: OS/400 - AFP Compatibility Fonts is included free of charge in the iSeries base operating system. It can be installed as an option on any iSeries system. The fonts are downloadable 240-pel raster fonts and are installed in font library QFNTCPL.

The AFP Compatibility Fonts include both uniformly spaced and mixed-pitch type families at 240-pel resolution. These font families are included: APL, Boldface, Courier, Document, Essay, Format, Gothic, Letter Gothic, Orator, Prestige, Roman, Script, Serif, Symbols, and Text.

Both the 240- and 300-pel versions of the AFP Compatibility fonts are provided in the AFP Font Collection.

GDDM Fonts: An option called OS/400 - GDDM is included free of charge in the iSeries base operating system and can be optionally installed on any iSeries system. If the option is installed, you can find a list of graphic symbol sets in the library QGDDM (look for object type *GSS). You can use these fonts in printer files, in DDS, and with the optional licensed program Business Graphic Utility (5763-BG1). They work on any IPDS printer that supports GOCA.

GDDM (Graphical Data Display Manager) has two types of graphic fonts:

Vector symbol sets

These are identified by an object attribute *VSS. Each character is built with a set of straight or curved lines.

Image symbol sets

These are identified by an object attribute *ISS. Each character is a pattern of dots quite similar to FGID, except that some graphic characteristics apply.

Note: Image symbols are not supported in DDS.

GDDM fonts are outline fonts, and can best be used for printing larger characters. For more information about GDDM, or for the font naming conventions, refer to *Printer Device Programming*.

AFP MICR Fonts, RPQ 8A8083: This font offering provides MICR fonts necessary for printing on IBM MICR printers. These fonts are also incorporated with Infoprint Designer.

Images and Graphics


Images and graphics are key components in creating advanced electronic print and presentation applications. In some cases, such as with accent images, the purpose of the image or graphic is to improve the look and effectiveness of output. In basic documents, images, such as an enterprise logo, are an essential part of the document. In other image applications, such as an integrated check image on bank statements, the image itself is part of the application. Images and graphics are usually included in a document in a page segment resource.

Some common examples of putting images and graphics to work in OS/400 applications include:

- Document logos
- Electronic signatures
- Accent images, such as the variable image used in the Super Sun Seeds sample application (see Figure 55 on page 138)
- Document images, such as check images and freight bill images
- Pictorial graphics, such as pie and bar charts

Figure 23 on page 62 shows a Super Sun Seeds invoice case study with an example of an image (the Super Sun Seeds logo) and a graphic (the pie chart). In fact, there are more graphics on the invoice than just the pie chart. The line and box elements on the invoice form are also graphics.

400 CPU Parkway
Vegetation, NJ 55090



Office: 555-499-2367
Fax: 555-415-9794

LOS ARBOLES DEL MUNDO
32483 ARBOL LANE
MESA VERDE
IL 65478-9390
SAME

Customer Number: 141 Invoice Number: 31341 Invoice Date: 7/28/02 Payment Date: 8/28/02

Ship Via: Ship Date: 7/28/02 Terms: N10 Rep: MICHELE GOODACRE


Qty	UOM	Item #	Item Description	Price	Extension
1000	BX	56413213	POT POT	7.65	7,650.00
45	BZ	11005015	CHANTENAY SEEDS	2.19	98.55
900	EA	00001200	ARBOLES DEL SUR	45.00	40,500.00
98	PK	84512023	OREGON SPRING TOMATO SEED	.97	95.06
4	BX	11057893	AFRICAN DAISY, SEEDS	2.35	9.40
951	CT	11005011	LASSO RED SEEDS	892.23	48,510.73
46	DZ	11005014	SCARLET NANTES SEEDS	5.90	271.40
100	EA	31321655	SEMILLAS DEL SUS SOMBEROS	24.95	2,495.00

Total Due \$99,630.14

Return this tear-off strip with your payment.
Make Checks Payable to: Super Sun Seeds

Payment is due by: 8/28/02
Amount Due is: \$99,630.14

LOS ARBOLES DEL MUNDO
 32483 ARBOL LANE
 MESA VERDE
 IL 65478-9390


 0001410099630146

Image

Graphic

Figure 23. Image and Graphic on the Sample Page

Images and Graphics in AFP Output

AFP output can contain both graphics and images. Images are captured while graphics are created. In many instances, these terms are used interchangeably, but for data processing applications, they are used very differently.

The term *image* is usually applied to a captured replica of an external source document. The source document can be a photograph, drawing, signature, or a printed or handwritten page. The source document is usually captured by a scanner. An image has to be in the appropriate format to be used with AFP output. There are several tools you can use to generate AFP images from other data types, such as the image (gif, tiff, and jpeg) to AFP transforms in Infoprint Server, and the Image Editor component of Infoprint Designer.

The term *graphic* is normally applied to pictorial data that is created internally. OS/400 applications that create graphical data include:

- Business Graphic Utility
- GDDM/400
- AFP Utilities

Graphics applications let you create shapes with lines or generate charts based on numerical input. The resulting graphic can be as simple as a box on an electronic form, or as complex as an engineering drawing.

Images and graphics are also distinguished by how the object is stored. Images are stored in digital format. Regardless of image file format, the image is represented by a string of bits that correspond to the exact layout of the dots on the page required to reproduce the image.

If the image is stored as a complete string of bits, it is called a raster, or bitmapped, image. There are data-reduction techniques that can compress the image, reducing amount of storage required to record the image.

An image is closely tied to the resolution at which it was created. An image that is scanned at one resolution and printed at another will change in size, and possibly proportion, unless some compensating manipulation is done.

Graphics are stored in vector representation and are similar to mathematical equations. A graphic is made up of commands to draw lines, arcs, and circles. The size of the final output can be varied by modifying the vector equations. Thus, graphics can easily be scaled or sized. They also take up relatively little storage space. In AFP, graphic elements are in the lines and boxes that comprise electronic forms, in the drawing orders for scalable or outline fonts, and in the bar and pie charts created with Business Graphic Utility and GDDM/400.

Image and Graphic Terminology

These are common terms used in working with images and graphics:

Table 11. Image and Graphic Terminology

Term	Definition
Bilevel	Image composed of only black and white pels. A bilevel pel is represented by a single bit. Most monochrome printers are bilevel image printers.
Clip	To truncate image information that lies outside a given boundary.
Compression	Method of making image data files smaller so that less disk space is used to store them and less system and printer resources are used to send and print them.
Crop	In image applications, to cut off or trim.
File Format	How image data is physically stored in a file. For example, how the image header is structured, and how the actual image data is stored. Each image file format has a unique structure that enables programs that work with images to identify which format is being used.
Page Segment	General term for an AFP image object. A page segment object is an IOCA image file with iSeries object wrappers.
Pel	Abbreviation for picture element. It is the smallest addressable element of an image. The term dots per inch (DPI) is also used.
Pixel	Early nickname for picture element. Pel is the more common term.
Raster	In computer graphics, a predetermined pattern of lines that provides uniform coverage of a display space. In other words, the entire image is represented by data bits. Contrast to vector where the image is represented by mathematical formulas.
Resolution	Number of pels per unit length on the image device, whether that device is a scanner, display, or printer. Common image resolutions are 144 pel (also called dots per inch, or dpi) for impact printers, and 240 pel, 300 pel, and 600 pel for printers and scanners.
Scale	In computer graphics, to enlarge or reduce all or part of a display image.
Scanner	Device to convert a physical image into digital form.

Table 11. Image and Graphic Terminology (continued)

Term	Definition
Wrappers	Header and trailer information “wrapped” around an image object enabling it to conform to a particular document architecture.

Image File Formats

These are the most widely used file formats for storing bilevel image data:

Table 12. Commonly Used Image File Formats

File type	Description
BMP	Bitmap, a simple format generally used for display, clip art, and so on. BMP is seldom compressed. Signature is X'4D42' in the first two bytes.
CGM	Computer Graphics Metafile. Industry standard specification for graphics creation.
EPS	Encapsulated PostScript. Subset of the PostScript page description language from Adobe. EPS supports both raster and graphic formats.
GIF	Graphical image format. A digital format that is used to compress and transfer graphical information over computer networks. For example, GIF is a common format for displaying graphical information on the Internet.
IM1	IPDS format for uncompressed raster image.
IMDS	Image Data Stream. Architected image format, more widely known as IOCA.
IOCA	Image Object Content Architecture. Can be compressed or uncompressed. Also known as IO1. IOCA defines five function sets: FS10, FS11, FS20, FS42, and FS45. FS10 is intended for bilevel images. FS11 and FS20 cover bilevel, grayscale, and color images. FS42 covers tiled bilevel images and tiled CMYK images with 1 bit per spot. FS45 carries tiled bilevel and CMYK images. CMYK images can be either 1 or 8 bits per spot. FS11 is a superset of FS10. FS45 is a superset of FS42. There are no other relationships among the functions sets.
PCX	Simple format, generally used for display, clip art, and freehand graphics. It has X'0A' in the first byte.
TIFF	Tagged Image File Format. Most common file format in document imaging and scanning. It has “II” or “MM” in the first two bytes and is compatible with a number of personal computing platforms.
JPEG	Joint Photographic Experts Group. A standard format for storing compressed true-color images commonly used for displaying documents on the Internet.

Images are generally stored on the OS/400 in two formats, IOCA and IM1. Images for AFP printing applications are stored as page segment resources. Images for ImagePlus and Workfolder Application Facility are IOCA and are stored in folders.

All IBM printers support IM1, and nearly all IPDS printers support IOCA image, which has become the standard. IOCA images:

- Can be compressed
- Can be placed in any position on the page
- Are resolution independent
- Can be scaled by the printer to any size
- Can be rotated by the printer
- Can be clipped by the printer so that only a portion of the image is printed

Image Compression

Storing an 8-1/2 by 11 inch page at 300 pel density requires over 1 million bytes for simple bilevel images. Grayscale and color images take up significantly more storage space. Uncompressed images have to be laboriously moved around the system to be displayed or printed.

Image compression can significantly reduce the storage size of an image along with improving the efficiency with which images are sent to the device. IPDS printers are designed to efficiently decompress compressed images.

IPDS printers support several different image compression techniques. These are some of the most common:

- ITU-TSS Group 3, the industry standard for facsimile (FAX) machines
- ITU-TSS Group 4, the industry standard for scanners
- MMR, used by IBM scanners
- ABIC, a compression technique used to handle checks and financial documents
- Solid rectangle fill, which is used by Infoprint Color xxx printers.

Compression techniques work most effectively with high contrast images, images that have a lot of black and white space. They are less effective with photographs. Referring again to an example of an 8-1/2 by 11 inch scanned image, this image could easily be compressed to 1/10 its size (100,000 bytes, or less) if it was a high contrast image.

Overlays

An overlay, or electronic form, is a collection of constant data stored as an AFP resource and primarily used in place of preprinted forms. It can include some or all of these elements:

- Vertical, horizontal, and diagonal rules
- Rules with different weights and thickness
- Boxes with and without shading
- Grids, arcs, and polygons
- Graphics or images, such as company logos
- Bar codes
- Text
 - In different inline directions and character rotations
 - In different fonts, including fonts not used in the input file

Figure 24 on page 66 shows an example of an overlay.

- Infoprint Designer. For information about using Infoprint Designer, see Chapter 17, “Using Infoprint Designer,” on page 197.
- IBM AFP Printer Driver for Windows. For information about using the AFP Printer Driver for Windows, see “Creating an Image Resource with the IBM AFP Printer Driver for Windows” on page 91.
- DDS (overlay equivalent using text, line, box, and paged keywords). For information about using DDS, see Chapter 14, “Using Data Description Specifications,” on page 125.
- Infoprint Server’s Windows-based image transforms. Infoprint Server can transform a GIF, TIFF, or JPEG file on your PC to an AFP overlay. See “Creating an Image Resource with the Infoprint Server Image Transforms” on page 91 for instructions.
- Forms transferred from other systems
- Services from IBM and other companies


Bar Codes

Bar codes represent characters, typically with sets of parallel bars of varying width and separation or varying heights. Combinations of bars and spaces form individual characters, which in turn represent a numeric or alphanumeric symbol that might be a product, part, or publication number. These are called *linear* bar codes. Bar codes can also be two-dimensional (2D). 2D bar codes allow large amounts of information to be encoded in a two-dimensional matrix. They are usually rectangular and use extensive data impaction and error correction codes.

Bar codes are designed to be read by a device called a bar code reader or scanner. The scanner must be compatible with the printed bar code symbology.

Bar codes can be produced using either AFP data stream presentation text data or an AFP data stream bar code data object. A bar code data object, which is a set of structured fields, specifies the type of bar code (the symbology), its size, and positioning information. Many different bar code symbologies can be produced using bar code objects.

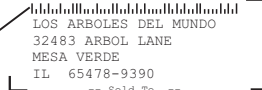
Figure 25 on page 68 shows bar codes printed on the sample page. The zip code at the top of the sample page is printed in POSTNET bar code. The customer number and invoice amount are printed in Interleaved 2 of 5 bar code at the bottom of the page.



400 CPU Parkway
Vegetation, NJ 55090

Office: 555-499-2367
Fax: 555-415-9794

Bar Code



LOS ARBOLES DEL MUNDO
32483 ARBOL LANE
MESA VERDE
IL 65478-9390

SAME

-- Sold To -- -- Ship To --

Customer Number: 141

Invoice Number: 31341

Invoice Date: 7/28/02

Payment Date: 8/28/02


Ship Via:

Ship Date: 7/28/02

Terms: N10

Rep: MICHELE GOODACRE

UOM	Item #	Item Description	Price	Extension
1000 BX	56413213	POT POT	7.65	7,650.00
45 BZ	11005015	CHANTENAY SEEDS	2.19	98.55
900 EA	00001200	ARBOLES DEL SUR	45.00	40,500.00
98 PK	84512023	OREGON SPRING TOMATO SEED	.97	95.06
4 BX	11057893	AFRICAN DAISY, SEEDS	2.35	9.40
951 CT	11005011	LASSO RED SEEDS	892.23	48,510.73
46 DZ	11005014	SCARLET NANTES SEEDS	5.90	271.40
100 EA	31321655	SEMILLAS DEL SUS SOMBEROS	24.95	2,495.00



Thank You
Because almost half of
your order was Lasso Red
Seeds, you will receive
a 10% discount on your
next order.

Total Due


\$99,630.14

Return this tear-off strip with your payment.
Payment is due by: 8/28/02

Make Checks Payable to: **Super Sun Seeds**

Amount Due is: \$99,630.14

LOS ARBOLES DEL MUNDO
32483 ARBOL LANE
MESA VERDE
IL 65478-9390



0001410099630146

Bar Code

Figure 25. Bar Codes on the Sample Page

Bar Code Terminology

Table 13 explains some of the basic language used in bar code applications.

Table 13. Bar Code Terminology

Term	Definition
Aspect Ratio	The ratio of the length (or height) of a bar to the length (or width) of the linear bar code symbol.
Bearer Bars	A bar across the top and bottom edge of a linear bar code. Partial scans of bar code symbologies, such as Interleaved 2 of 5, can produce valid, but incorrect, reads by self-discriminating scanners. Bearer bars help prevent such errors and increase reliability. Note: A self-discriminating scanner is one that automatically determines which bar code symbology it is reading.
Check Character	The result of some mathematical combination of the characters in the field being bar coded. Used as a check of the accuracy of both the input of the data field and the scanning of the bar code. A bar code can have 0, 1, 2 or sometimes more check characters.
Check Digit	Same as a check character, but limited to decimal digits only.
Continuous Code	A linear bar code in which each character starts immediately after the preceding character. There is no space or gap between characters. Interleaved 2 of 5 is an example of a continuous bar code.

Table 13. Bar Code Terminology (continued)

Term	Definition
Density	A measure of the number of characters per inch or per millimeter represented by the bar code. A high-density bar code represents more characters per inch than a low-density bar code. The bar code symbology helps determine the density. Within a given symbology, factors that affect the density of a bar code are: the x-dimension (width of the narrow bar) and the wide-to-narrow ratio (width of a wide bar relative to the narrow bar).
Discrete Code	A linear bar code constructed with groups of bars and spaces representing individual characters and having a space or intercharacter gap between each group. This gap is used solely to separate characters and contains no data. Code 3 of 9 is a discrete bar code.
First Read Rate	The percentage of the bar code scans that read correctly on the first scan of the bar code. A 99% or higher first read is desirable. Anything below 85% is normally not acceptable.
Human Readable Interpretation (HRI)	The translated characters of a linear bar code that can be read by a human.
Intercharacter Gap	The space between characters in a discrete bar code symbology.
Ladder Orientation	Linear bar code orientation where the bars are parallel to the base of the document (like the rungs of a ladder). It is sometimes called vertical orientation because that is the direction of the scan.
Picket Fence Orientation	Linear bar code orientation where the bars are perpendicular to the base of the document (like the pickets in a picket fence). Sometimes called horizontal orientation (because that is the direction of the scan).
Quiet Zone	A blank area prior to and following a bar code. This required space enables the scanner to differentiate the start and stop of a bar code. The size of the quiet zone is usually 10 times the x-dimension or 1/4 inch, whichever is larger.
Stop Pattern	A special bar code character that provides the scanner with stop reading instructions. The stop character is normally at the right end of a horizontally oriented symbol.
Symbology	The name of the specification used to define the combinations of bars and spaces for encoding data.
Wide-to-Narrow Ratio	The ratio of the width of the wide bar or space to the narrow bar (x-dimension) or space in a two-width symbology. This ratio is usually in the range of 2:1 to 3:1.
X-Dimension	The width (usually in thousandths of an inch) of the narrow bar or space of the bar code symbology.

Bar Code Basics

All linear bar codes, regardless of the symbology used, are created using the same, basic elements. Figure 26 on page 70 shows the common elements of all bar codes.

Bar Code Basics

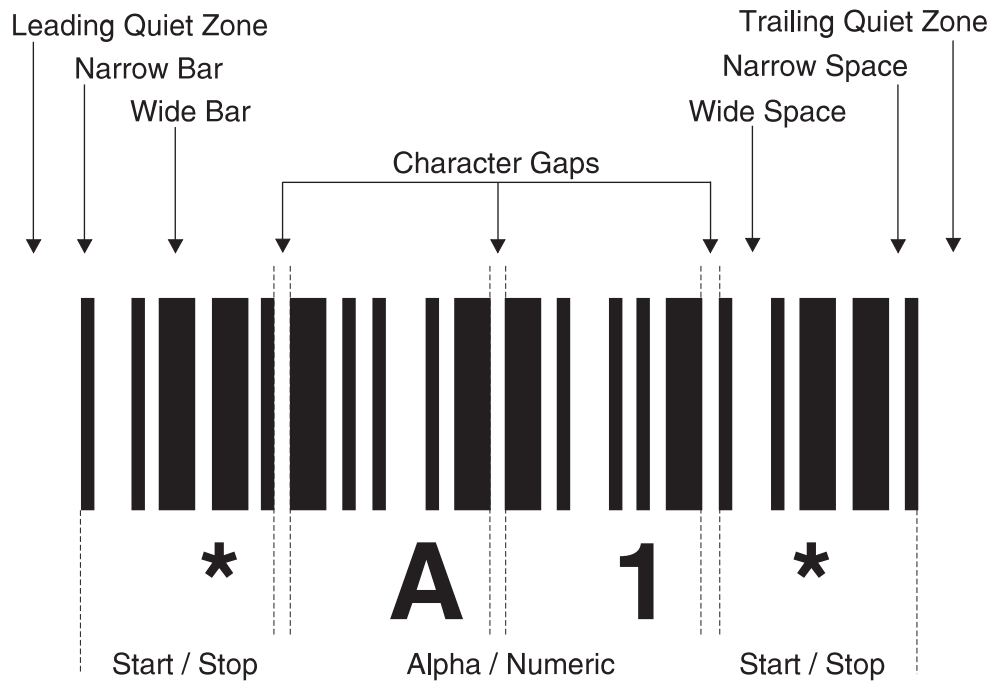


Figure 26. Linear Bar Code Elements

Bar Code Symbolologies

Many different types of bar code symbolologies, or coding arrangements, have been developed. Among the more widely used bar code symbolologies are:

- Code 3 of 9
- Codabar
- Interleaved 2 of 5
- Code 128
- Universal Product Code (UPC)
- European Article Numbering (EAN)
- Postnet (Postal Bar Code)
- Royal Mail (Postal Bar Code)
- Japan Postal (Postal Bar Code)
- Australian Postal (Postal Bar Code)
- PDF417 (2D)
- Maxicode (2D)
- Data Matrix (2D)

Examples

This section contains examples of these bar codes:

- “Universal Product Code Example”
- “Postnet (Postal Bar Code) Bar Code Example” on page 71
- “Portable Data File 417 (PDF417) Bar Code Example” on page 71

Universal Product Code Example: Universal Product Code (UPC) is a continuous bar code symbology that also uses the module approach to construct digits. UPC uses two bars and two spaces and there is no wide-to-narrow ratio.

This type of bar code is divided into two halves, a left side and a right side. Digits on the left side are coded in odd parity (bar widths total up to an odd number of modules) while digits on the right side are coded in even parity. This enables either side of the symbol to be decoded separately and provides the advantage of being able to scan the bar code from almost any angle.

Figure 27 shows an example of the UPC bar code symbology.

UPC-A



Figure 27. UPC Bar Code Example

Postnet (Postal Bar Code) Bar Code Example: The Postnet bar code uses five bars to represent a digit. The data is based on the height of the bars instead of on their width. This symbology requires a check or correction digit. Postnet is a numeric bar code only with each digit having two tall and three short bars.

Figure 28 shows an example of the Postnet bar code symbology.

POSTNET ZIP



Figure 28. Postnet Bar Code Example

Portable Data File 417 (PDF417) Bar Code Example: PDF417 is a two-dimensional (2D) stacked bar code symbology. The bar code can be full or truncated, as shown in Figure 29. The truncated PDF417 bar code symbol is for use in a relatively clean environment in which damage to the symbol is unlikely. This version omits the right row indicator and simplifies the stop pattern into a single module width bar.

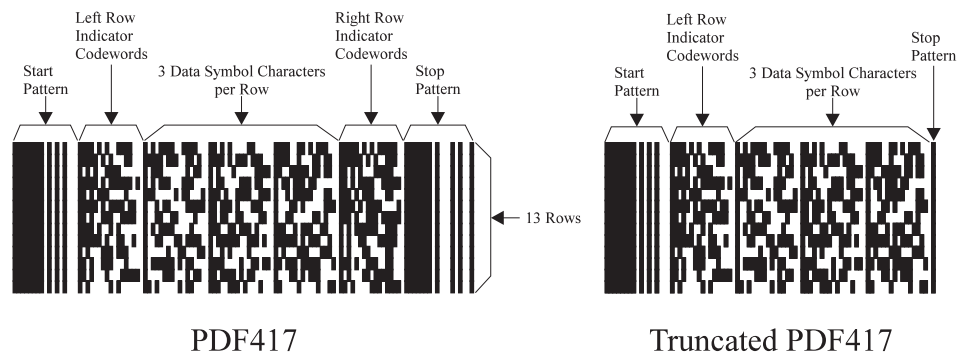


Figure 29. PDF417 Bar Code Example

Page Definitions and Form Definitions

AFP supports two external resources for formatting pages – page definitions and form definitions. These resources contain rules for mapping application output into fully composed pages. Page definitions and form definitions are external to the application program and thus separate the final document formatting from the application.

Figure 30 shows how page definitions and form definitions change the print flow. In the figure, a print application sends lines of output to the output queue. By specifying the page definition and form definition objects in the printer file, the line data output produced in the output queue is transformed by PSF to AFP.

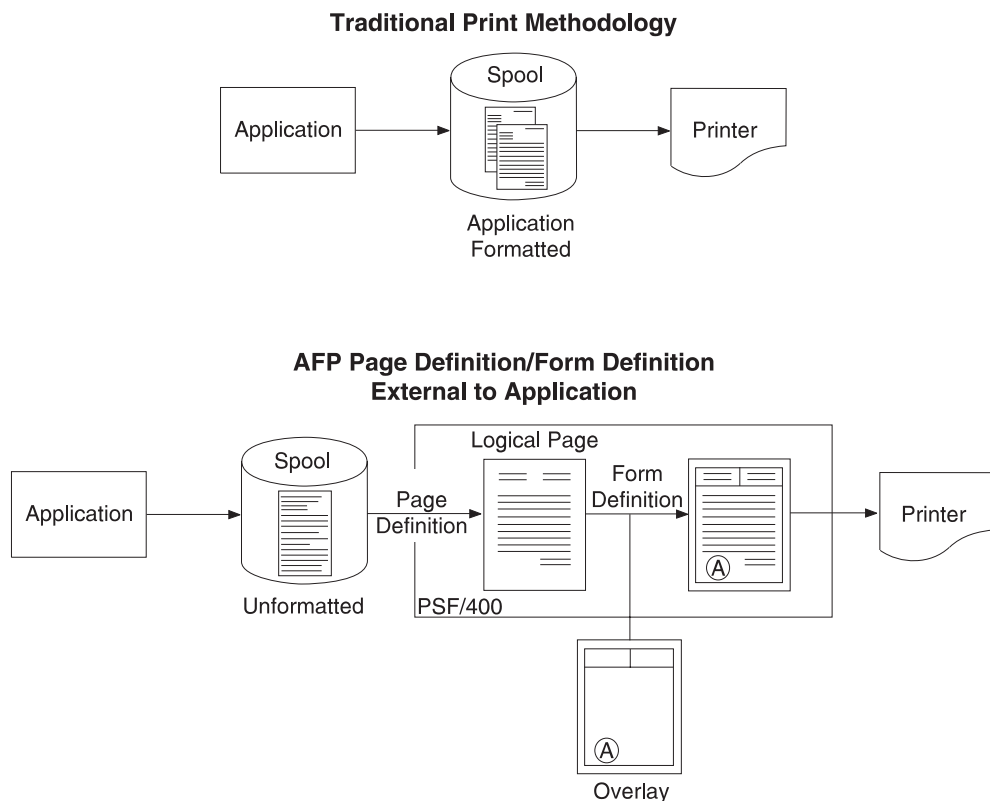


Figure 30. Traditional Printing Compared to Printing with Page Definitions and Form Definitions

Page Definitions

A page definition defines how line data (traditional and record format) and XML data is placed on a logical page by PSF (page definitions are not used for AFP data because it is already composed into pages). Input print lines are read in, optionally parsed into individual fields, and placed on the page. Similar in structure to DDS, the page definition language lets you place print lines or print fields anywhere on the page while controlling fonts, orientation, color, and more. A page definition performs many of the same functions that DDS does, but page definitions are independent of the application program. Figure 31 on page 73 shows how the page definition works.

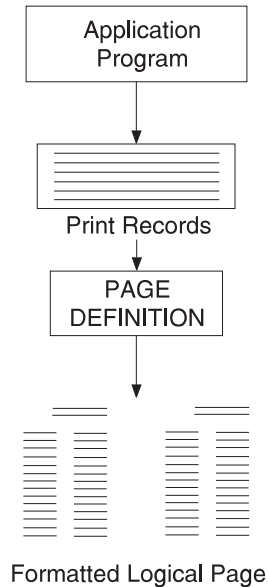


Figure 31. How a Page Definition is Used

Specifying a page definition is optional. If you do not specify a page definition, media handling characteristics are taken from existing keywords. If you do specify a page definition, its media handling instructions override those specified in existing keywords. The form definition is normally used with the page definition. It can, however, be used by itself.

Page definitions can contain conditional logic. This means that the formatting rules can change based on the contents of an input field. A “trigger” field (for example, company number) can be used to select a whole series of formatting commands.

These are some of the options you can specify with page definitions:

- Dimensions of the logical page
- Print direction of the logical page (landscape or portrait)
- Print direction of lines and fields relative to the logical page
- Conditional processing (enabling the content of the print data to control the layout and media handling of the print job)
- Line spacing (number of lines per inch)
- Location of individual lines and fields
- Number of lines per page
- Page segments for inclusion in printed output
- Overlays for inclusion in printed output (positioned anywhere on the page)
- Page-ejection points
- Fonts and font rotation used on a page
- Multiple-up printing (placing more than one logical page on one side of a single sheet)
- Colors to be used (on printers that support this function or for viewing)
- Print position for carriage controls or channel codes
- Constant data to be printed
- Bar codes to be printed
- XML formatting
- Record format line data formatting

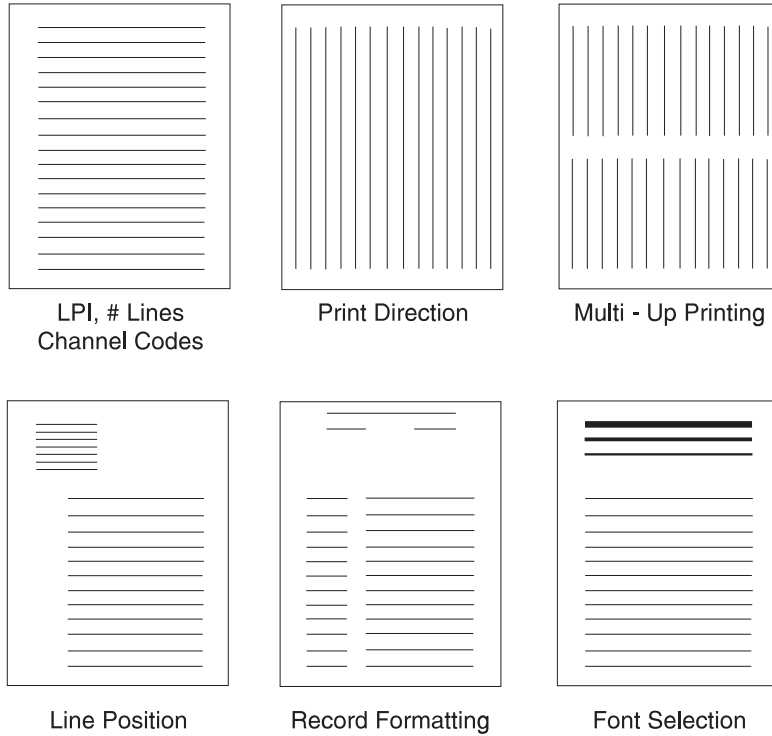


Figure 32. Page Definition Functions

See Chapter 16, “Using Page Printer Formatting Aid,” on page 183 and Chapter 17, “Using Infoprint Designer,” on page 197 for information about creating page definitions.

Form Definitions

The form definition controls how the logical page (defined with the page definition) is placed on the physical medium - the sheet of paper. Figure 33 on page 75 illustrates how a form definition works.

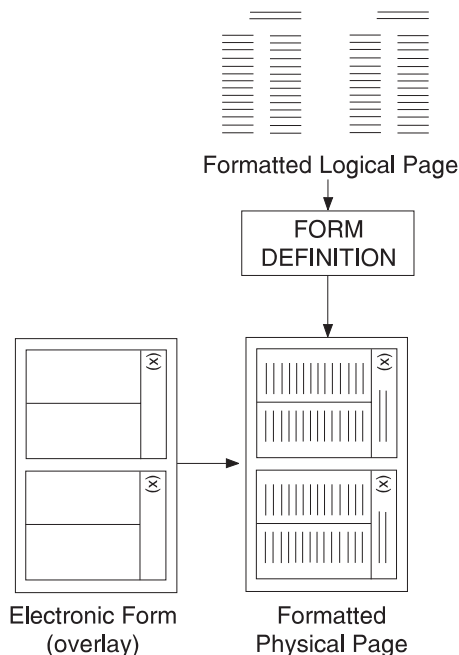


Figure 33. How a Form Definition is Used

A form definition performs many of the same functions as a printer file on the OS/400, but a form definition is optional. If you do not specify a form definition, media handling characteristics are taken from other existing keywords. If you do specify a form definition, its media handling instructions override those specified in existing keywords.

These are some of the options you can specify with form definitions, within the limits of your printer:

- Position of a logical page on a physical page (page origin).
- Duplex or simplex printing.
- Overlays to use.
- Number of copies of any page.
- Suppression of selected fields on a copy.
- Jog (the offset stacking of cut-sheet output or copy marking on continuous-forms output).
- Paper source for a cut-sheet printer (input bin).
- Print quality.
- Constant back (enables printing of a page without variable data).
- N_UP printing – printing up to eight logical pages on either side of a sheet. For example, you could have two logical pages on one side of a sheet and six logical pages on the other side of the same sheet.
- Page orientation (portrait or landscape).
- Postprocessing controls, such as:
 - selecting functions
 - perforating
 - cutting
 - stapling
 - folding
 - hole punching
 - nested finishing

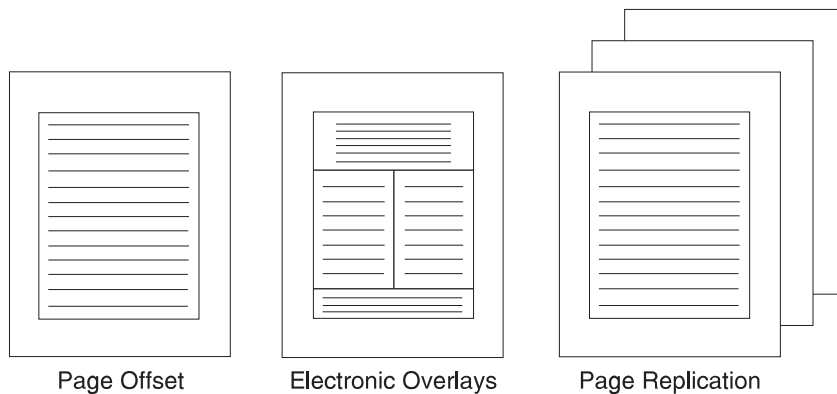


Figure 34. Form Definition Functions

See Chapter 16, “Using Page Printer Formatting Aid,” on page 183 and Chapter 17, “Using Infoprint Designer,” on page 197 for information about creating form definitions.

How Page Definitions and Form Definitions are Used

Page definitions and form definitions, while frequently used together, can also be used independently. They even support different types of data. You can use a form definition with *SCS, *IPDS, *AFPDS, *AFPDSLNE, and *LINE data, but you can only use a page definition with *USERASCII for XML data, *LINE data, and *AFPDSLNE data. When specified, page definitions and form definitions are used throughout the print process:

1. Preparation stage
 - a. The application generates line data.
 - b. The person responsible for designing the output layout creates a page definition and form definition to format the line data. This step might require detailed knowledge of the application-generated line data.
2. Production stage
 - a. The application generates line data specifying the corresponding page definition and form definition resources through printer file parameters.
 - b. The line data is sent to a printer defined as AFP(*YES), where PSF uses the page definition and form definition to create a formatted data stream which is sent to the printer.

You can use page definitions and form definitions to specify options that are also specified on the printer file. Because of this, you should be aware of how the printer file interacts with page definitions and form definitions.

Printer File Keyword Support when Using Page Definitions and Form Definitions

With line data, the OS/400 can receive layout and media handling instructions through existing printer file command keywords and from page definitions and form definitions. Some printer file keywords are ignored when page definitions and form definitions are used.

Printer file keywords that are ignored when a page definition is used:

These printer file keywords are ignored when a page definition is used. If you specify line data to PSF and do not specify a page definition, a page definition is built by using the values for these parameters. However, this page definition is not part of the spooled file and cannot be transferred to another system.

CDEFNT	FONT	MULTIUP
CHRID	FNTCHRSET	PAGESIZE
CPI	LPI	PAGRRT
FOLD	LVLCHK	REDUCE

Printer file keywords that are ignored when a form definition is used:

These printer file keywords are ignored when a form definition is used. If you specify line data and do not specify a form definition, a form definition is built inline by using the values for these parameters.

BACKMGN	FOLD	MULTIUP
DRAWER (if *FORMDF is specified)	FORMFEED	PAGRRT
DUPLEX (if *FORMDF is specified)	FRONTMGN	PRTQLTY
FOLD	LVLCHK	REDUCE

Printer file keywords that override form definition values:

These printer file keywords, if used, override the values specified in a form definition:

- DRAWER (if *FORMDF is not specified)
- DUPLEX (if *FORMDF is not specified)

Sources of Page Definitions and Form Definitions

You can use many different methods to obtain page definitions and form definitions. Each option varies in level of skill required and cost. Use the approach that is best for you:

- Infoprint Designer is the standard graphical product for creating page definitions and form definitions interactively on top of an overlay design application. This lets you design the overlay and then format the existing print lines on top of that form. See Chapter 17, "Using Infoprint Designer," on page 197 for more information.
- PSF is shipped with a standard set of pre-compiled page definitions and form definitions to get you started immediately. The names and characteristics of these resources are included in the accompanying documentation.
- PPFA is the AFP PrintSuite feature for creating page definitions and form definitions. It can build page definitions for line data (traditional and record format) and XML. See Chapter 16, "Using Page Printer Formatting Aid," on page 183 and *Page Printer Formatting Aid: User's Guide* for more information.
- You can create page definitions and form definitions on other AFP platforms, such as MVS™, VM, VSE, z/OS and AIX.
- IBM Printing Systems Solutions Service Center.
- Other third party tools can generate these resources.

Object Containers

Object containers are used to carry non-OCA objects, such as Encapsulated PostScript (EPS) and PDF, in an AFP data stream. These non-OCA objects can be *wrapped* or *unwrapped*. Wrapped objects are carried in a MO:DCA envelope called an object container. Unwrapped objects are unaltered from their native form. If the object is to be carried in MO:DCA resource groups and interchanged, it must be wrapped.

Object containers, even though they contain non-OCA data, are similar to other AFP resources in that they:

- Can be mapped. A mapped resource is downloaded once per spooled file, irrespective of the number of times the resource is referenced within the spooled file.
- Can be included on a page.
- Can be resident in the printer.
- Can be captured in the printer.

Using object containers has several benefits:

- You can reference several types of OCA objects (image data, bar code data and graphics data) in a spooled file without having to include them in an overlay or page segment.
- You can also scale and rotate these objects. If they were in a page segment or overlay and needed to be rotated, you would have to create one copy of the object in each orientation.
- You can include images larger than 16 MB in your output. If the image exceeds 16 MB, the image cannot be stored as a *PAGSEG object.
- Sophisticated print applications can specify the use of non-OCA objects which refer to other non-OCA objects, such as color mapping tables (which are printer-resident). These are called *secondary resources*.
- Applications can use color objects with IPDS printers.
- Infoprint Server's PDF subsystem accepts object containers with certain types of images as input. Refer to *Infoprint Server for iSeries: User's Guide* for more information.

See Chapter 11, "Working with Object Containers," on page 109 for information about specifying object containers in your data.

Chapter 7. Working with Fonts

Fonts are used to represent text and sometimes bar codes. “Fonts” on page 41 introduces fonts. This chapter describes how to use fonts on the iSeries. It contains these sections:

- “Create a Document with Several Fonts” gives an example of using fonts, employing several different composition methods.
- “Printing a Font Catalog” on page 82 describes the free AFP Font Collection Tools download. This helps you load AFP Font Collection fonts onto your iSeries and print a font catalog with samples of the available fonts.
- “Using Outline Fonts” on page 82 explains how to use outline fonts instead of raster fonts without changing your application.
- “Working with Font Mapping Tables” on page 82 explains how to change the font substitution tables.
- “Capturing Fonts” on page 83 explains how to mark fonts eligible for capture.
- “Anamorphic Font Character Scaling” on page 84 explains how to scale fonts without maintaining the original proportions.

Create a Document with Several Fonts

In this section, we will specify several different fonts in a document. There are many different tools you can use to do this. Each tool has different capabilities, use the tool that best fits your needs. These products are discussed in greater detail in Chapter 3, “AFP Applications,” on page 19 and throughout the document. Chapter 25, “Deciding on an Output Strategy and Set of Tools,” on page 287 can help you choose the best product for you.

DDS You can use keywords in your DDS application to specify fonts; including how they look and how they are placed. The example in this section uses DDS.

Infoprint Designer

This graphical tool lets you use menus to specify fonts and font properties, such as size and weight. You can also specify placement properties, such as alignment and orientation. Chapter 17, “Using Infoprint Designer,” on page 197 gives an example of using Infoprint Designer.

AFP Printer Driver

This driver can be used to produce AFP documents, overlays, and page segments from any Windows 95/98, Windows NT, or Windows 2000 application that supports printing to AFP printers.

APU Chapter 15, “Using Advanced Print Utility,” on page 153 gives an example of using APU.

AFP Toolbox

With AFP Toolbox you can specify a font, its properties (such as style and weight), and specify its formatting on the page (such as alignment). Chapter 18, “Using IBM AFP Toolbox,” on page 217 gives an example of using AFP Toolbox.

AFP Utilities

You can specify fonts in the components of AFP Utilities. Chapter 19, “Using the Print Format Utility,” on page 223 gives an example of using AFP Utilities.

Third party applications

Several third party applications exist that you can use to work with fonts in a document. Some such products are Papyrus, Elixir, Opus, StreamServe, and DOC1.

Example

In this example, we use RPG and DDS to specify several different fonts in a document. The fonts are different sizes, styles, orientations, and colors, and use different code pages to produce different characters. We also use both printer resident (FGID) and host resident (character set and code page) fonts that are fixed-pitch, proportional or typographic, as well as outline and raster fonts. The sample document is shown in Figure 35 on page 81. The DDS keywords used to create the document follow in Figure 36 on page 82.

Using Fonts With DDS

1

The brown fox jumped ABC123! ☒ #

Courier, Roman, Medium, 20p
 FGID 416 Code page 37 (default)
 Fixed Pitch, Printer-Resident, Outline

The brown fox jumped ABC123! ☒ #

Boldface
 Character set COS0BRTR, Code page T1V10037
 Proportional, Host-Resident, Raster

The brown fox jumped ABC123! ☒ #

Times New Roman, Italic, Bold, 18p
 FGID 2311 Code Page 37 (default)
 Typographic, Printer-Resident, Outline

The brown fox jumped ABC123] ☒ #

Times New Roman, Italic, Bold, 18p
 Character set CON500H0, Code page T1V10500
 Typographic, Host-Resident, Raster

The brown fox jumped ABC123] ☒ N

Times New Roman, Italic, Bold, 18p
 Character set CZN500, Code page T1V10284
 Typographic, Host-Resident, Outline

The brown fox jumped ABC123] € #

Times New Roman, Italic, Bold, 18p
 Coded font XON51EHC
 Character set CON500H0, Code page T1001148
 Typographic, Host-Resident, Outline

The brown fox jumped ABC123! ☒ #

Helvetica, Roman, Medium, 24ph 10pw
 FGID 2304 Code Page 37 (default)
 Typographic, Host-Resident, Outline

The brown fox jumped ABC123] ☒ #

Helvetica, Roman, Medium, 48ph 10pw
 Character set CZH200, Code page T1V10274
 Typographic, Host-Resident, Outline

THE BROWN FOX JUMPED ABC123] ☒ #

OCR-A
 FGID 19 Code Page 37 (default)
 Fixed Pitch, Host-Resident, Raster

2
3
4
5
6
7
8
9
10
11
12

Using Fonts With DDS

Using Fonts With DDS

Figure 35. Document with Several Fonts

The fonts in the example have been specified using these DDS keywords. The numbered keywords below correspond to the numbering in the sample document.

Note:

```

Line 1:  FONT(2305 (*POINTSIZ 18)) COLOR(RED)
Line 2:  FONT(416 (*POINTSIZ 20)) COLOR(GRN)
Line 3:  FNTCHRSET(C0S0BRTR T1V10037) COLOR(BLU)
Line 4:  FONT(2311 (*POINTSIZ 18)) COLOR(PNK)
Line 5:  FNTCHRSET(C0N500H0 T1V10500) COLOR(*RGB 50 50 75)
Line 6:  FNTCHRSET(CZN500 T1V10284 (*POINTSIZ 18)) COLOR(*RGB 75 20 40)
Line 7:  CDEFNT(X0N51EHC) COLOR(*CMYK 0 50 50 0)
Line 8:  FONT(2304 (*POINTSIZ 24 10)) COLOR(*CMYK 50 0 0 50)
Line 9:  FNTCHRSET(CZH200 T1V10274 (*POINTSIZ 48 10)) COLOR(TRQ)
Line 10: FONT(19) COLOR(BRN)
Line 11: FONT(420 (*POINTSIZ 18 50)) COLOR(RED) TXTRTT(270)
Line 12: FONT(420 (*POINTSIZ 18 50)) COLOR(RED) TXTRTT(90)

```

Figure 36. DDS Keywords Used to Create Sample Document

Notes:

1. The TXTRTT keyword only works with fields, not constants.
2. The print data for lines 1 through 10 is constant text and was specified in the DDS source.
3. The print data for lines 11 and 12 is a data field that contains the printed text.

For more information about DDS keywords for printing, refer to the *DDS Reference: Printer Files* topic in the iSeries Information Center. To find the topic, from the Information Center iSeries search for “DDS Reference”.

Printing a Font Catalog

iSeries AFP Font Collection Tools is a free download that contains commands that help you load AFP Font Collection fonts onto your iSeries and print a font catalog with samples of the available fonts. You can use the commands if you meet these criteria:

- You can transfer files from your PC to an iSeries using FTP.
- You have the AFP Font Collection “Fonts for OS/400” on CD or tape media.
- You use PSF for OS/400 to print to an IPDS printer.

You can get the tools from: <http://www.printers.ibm.com/r5psc.nsf/web/rdfont01>. This Web page also has instructions about loading and using the tools.

Using Outline Fonts

If you have an application that currently uses downloadable AFP raster fonts and you want to use equivalent downloadable outline fonts when possible, you can request that substitution without changing the application. To do this, use the USEOUTLFNT parameter on a PSF configuration object, then specify that PSF configuration object on the printer device you are using.

Working with Font Mapping Tables

You can use the OS/400 commands listed below to work with font mapping tables, with these restrictions:

Restrictions:

1. You cannot change system font mapping tables.
2. You can only create one font mapping table of each kind, except printer-resident to printer-resident.

- Host-to-printer and printer-to-host font mapping tables are given one of these names as appropriate: *PHFCS, *HPFCS, *PHCP, *HPCP as appropriate, and are stored in QUSRSYS.

Add font table entry (ADDFNTTBLE)

Adds an entry to a user-defined font mapping table. For example, `ADDFNTTBLE FNTTBL(*PHFCS) PHFCS((NNN *NONE) (CHAR_SET *OUTLINE))` maps the printer font identifier *NNN* to the outline font character set *CHAR_SET*. *NONE is specified for point size because the host character set is an outline font.

Change font table entry (CHGFNTTBLE)

Changes an entry in a user-defined font mapping table. For example, `CHGFNTTBLE FNTTBL(*HPFCS) HPFCS((CHAR_SET) (NNN))` specifies that the host font character set *CHAR_SET* is now mapped to the printer font identifier *NNN*, instead of the previously used printer font identifier.

Create font table (CRTFNTTBL)

Creates a user-defined font mapping table. You can only have one font mapping table of each kind except printer-to-printer. For example, `CRTFNTTBL FNTTBL(*PHFCS)` creates the user-defined PHFCS font mapping table if one does not already exist.

Delete font table (DLTFNTTBL)

Deletes any user-defined font mapping table. For example, `DLTFNTTBL FNTTBL(*PHFCS)` deletes the user-defined PHFCS font mapping table.

Display font table (DSPFNTTBL)

Displays any font mapping table at the requesting work station or prints it with the job's spooled output. For example, `DSPFNTTBL FNTTBL(*SYSPHCP)` displays the system PHCP font mapping table on your screen.

Remove font table entry (RMVFNTTBLE)

Removes an entry from a user-defined font mapping table. For example, `RMVFNTTBLE FNTTBL(*HPCP) HPCP(CDE_PAG)` removes the entry that mapped the host code page *CDE_PAG* to a printer code page.

Note: Because improper use of font substitution can cause unpredictable results, only system administrators should work with the font mapping tables.

Capturing Fonts

Captured fonts are host-resident fonts that are captured by the printer and treated as printer-resident fonts as long as there is room for them. This can improve performance by reducing font downloads to printers that support font capture. You cannot clear a captured font from the printer, so it is possible that an unauthorized person could access the captured font. Therefore, you should not mark sensitive fonts, such as signatures and MICR fonts, as eligible to be captured.

Note: Because improper use of font capturing can cause unpredictable results, only system administrators should handle the font capture feature. Refer to *Printer Device Programming* for information about the proper use of font capturing.

Font capturing is turned on in a PSF configuration object. You can use OS/400 commands or AFP Manager (a feature of iSeries Navigator) to activate font capturing. You can also use iSeries Navigator to enter OS/400 commands. For instructions, refer to *Infoprint Server for iSeries: User's Guide*.

To activate font capturing using OS/400 commands, follow these steps:

1. Mark the font eligible for capture. To do this, specify FNTCAPTURE (*YES) in the appropriate font resource command. You can specify this value on a new font character set or code page by using "Create font resource" (CRTFNTRSC) or change a font character set or code page by using "Change font resource" (CHGFNTRSC). For example, to mark the existing font character set CHR_SET eligible for capture, enter this command:

```
CHGFNTRSC FNTRSC(CHR_SET) FNTCAPTURE(*YES)
```

2. Activate font capturing. To do this, specify FNTCAPTURE(*YES) on a PSF configuration object. You can specify this value on a new PSF configuration object by using "Create PSF configuration object" (CRTPSFCFG) or change a PSF configuration object by using "Change PSF configuration object" (CHGPSFCFG). For example, to activate font capture in the PSF configuration object named MY_CFG, enter this command:

```
CHGPSFCFG PSFCFG(MY_CFG) FNTCAPTURE(*YES)
```

To use AFP Manager to activate font capture, follow the steps in "Activating Font Capture" on page 351.

Anamorphic Font Character Scaling

Anamorphic font character scaling lets you stretch or compress a character, disregarding the font's original proportions. By default, OS/400 scales fonts proportionally - scaling the width and height the same amount. You can scale outline fonts anamorphically with one of these tools:

DDS Keywords

You can use these DDS keywords to scale fonts:

- **CDEFNT, FNTCHRSET, and FONT** allow you to scale resident and downloaded outline fonts. Use one of these keywords if you have an AFCCU printer.
- **CHRSIZ** lets you scale GDDM graphic characters, printer-resident outline fonts, and downloaded outline fonts.

PPFA You can use a page definition created with PPFA that specifies RATIO to stretch or compress downloaded outline fonts.

Infoprint Designer

You can use the Width field in the font dialogs (such as the Font1 and Font2 pages on the "Field properties" dialog) to control outline character scaling.

Steps to Restore Font Libraries From the Font Collection CD to the iSeries

Follow these steps to restore CPL300 from the CD to library QFNT300CPL on the OS/400. A free download, AFP Font Collection Tools, also has commands for installing Font Collection fonts on your iSeries. Refer to this Web page <http://www.printers.ibm.com/r5psc.nsf/web/rdfont01> for more information.

1. To see the libraries that are on the Font Collection CD:
 - a. Place the CD in the iSeries CD drive.
 - b. On a command line, enter WRKOPTDIR. The Work with Optical Directories screen opens.

- c. In the **Opt** column for the Font Collection CD, specify option 8 (work with files). The Work with Optical Files screen opens and displays a list of all files on the CD. The commonly installed Font Collection files are listed in Table 6 on page 49.
2. To restore that file to an OS/400 library, enter this command:

```
RSTLIB SAVLIB(Font_Collection_file_name) DEV(your_iSeries_CD_device_name)  
RSTLIB(OS/400_library)
```

For example, to restore the Font Collection file CPL300 to the library QFNT300CPL on an iSeries where OPT01 is the CD device, use this command:

```
RSTLIB SAVLIB(CPL300) DEV(OPT01) RSTLIB(QFNT300CPL)
```

3. Check to see if the restore was successful.
 - a. Enter WRKLIB QFNT300CPL
 - b. Specify option 12 (Work with objects) to see the objects in the library.
4. Repeat steps 2 and 3 for all desired font libraries.

Chapter 8. Working with Images

Images are key components in creating advanced electronic print and presentation applications. They can be used to improve the look and effectiveness of output, can be an essential part of the document, or can actually be part of the application. Images are raster patterns of pels that make up a picture. “Images and Graphics” on page 61 introduces images.

This chapter describes how to create images and how to get images into a format you can use in your documents. It also introduces some different options for including images in your iSeries output.

Building Images

Creating an image for OS/400 printing applications involves several steps, from scanning the source document to creating the page segment object. The steps are:

1. “Scanning the Source Image”
2. “Touching up the Image”
3. “Converting Images to IOCA” on page 90
4. “Uploading and Compiling the Image” on page 92

Note: There are also services available from IBM to build image objects for your OS/400 applications.

Scanning the Source Image

The first step in creating an image for use with OS/400 documents is to get an electronic version of the image. Scanning an image converts the source document into digital format. Some scanning software writes the digitized image to a file, in an image format such as TIFF. Other software can pass the digitized image directly to an editing program.

The quality of the source document is essential in getting a good scanned image. The artwork to be scanned should be high-contrast, black and white reproductions. Signatures with medium black felt tip pens on white paper should scan well.

Note: Scanning should be done in the same resolution that the image will be printed in.

Touching up the Image

After scanning, images normally need some touch-up to become clear and sharp. Touching up removes the imperfections in the scanning process.

There are many software choices for image editing. Many Windows programs process image file formats and support image editing. A number of programs, such as Infoprint Designer for iSeries, are part of dedicated systems used to build AFP image and forms. Whatever the tool, the touch up process consists of turning on or off black and white pels to create a sharp, clear image. Below, we show how to use Infoprint Designer to touch up an image.

Figure 37 on page 88 shows the window that opens upon starting the Image Editor component of Infoprint Designer.

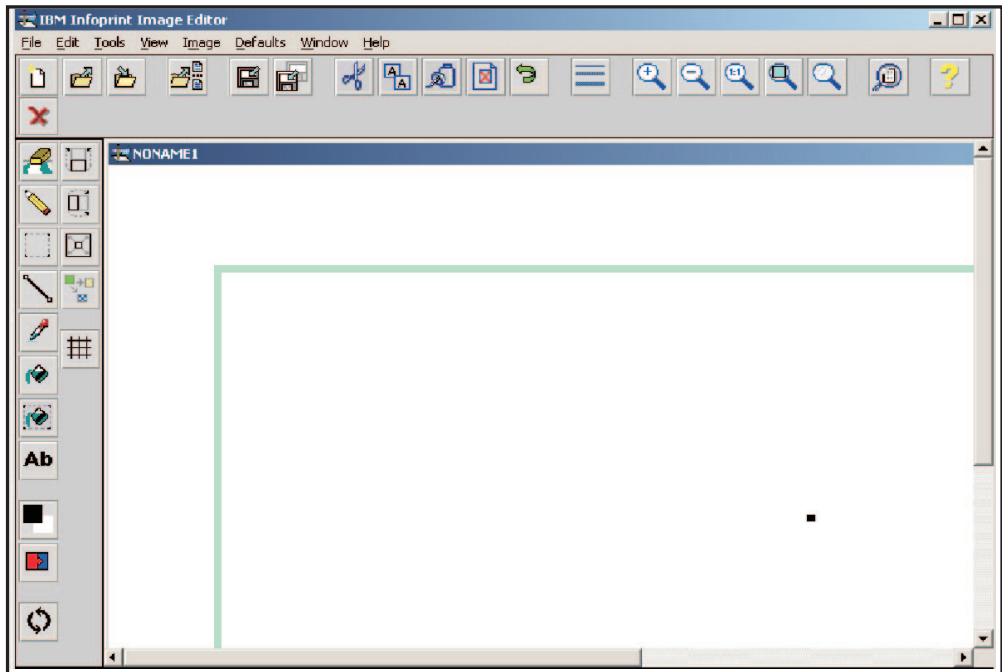


Figure 37. Image Editor

You can scan an image into the editor or open an existing AFP, TIFF, BMP, JPEG, GIF, ICO, or PTR image. Figure 38 shows the Super Sun Seeds logo image as scanned into the Image Editor component of Infoprint Designer. Most image systems like the Image Editor have a series of image touch-up tools with varying levels of granularity that let you work down to the individual pel level.

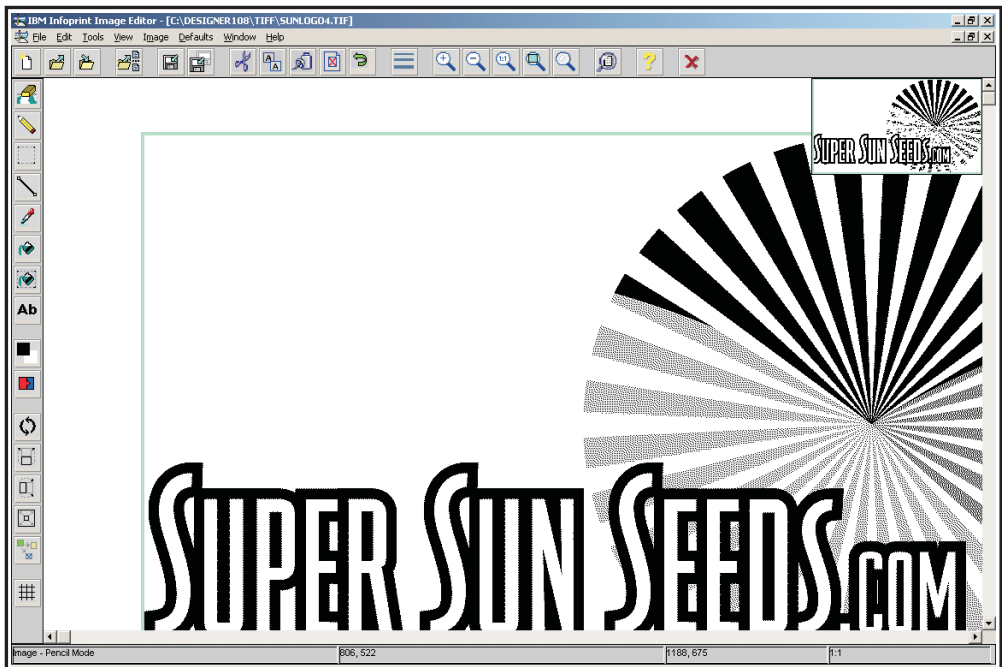


Figure 38. Super Sun Seeds Logo Scanned into the Infoprint Designer Image Editor

Because the Super Sun Seeds logo will be used in 300-pel resolution, each square inch will have 90,000 individual dots or pels. Therefore, it is necessary to zoom in on this image to work with it effectively.

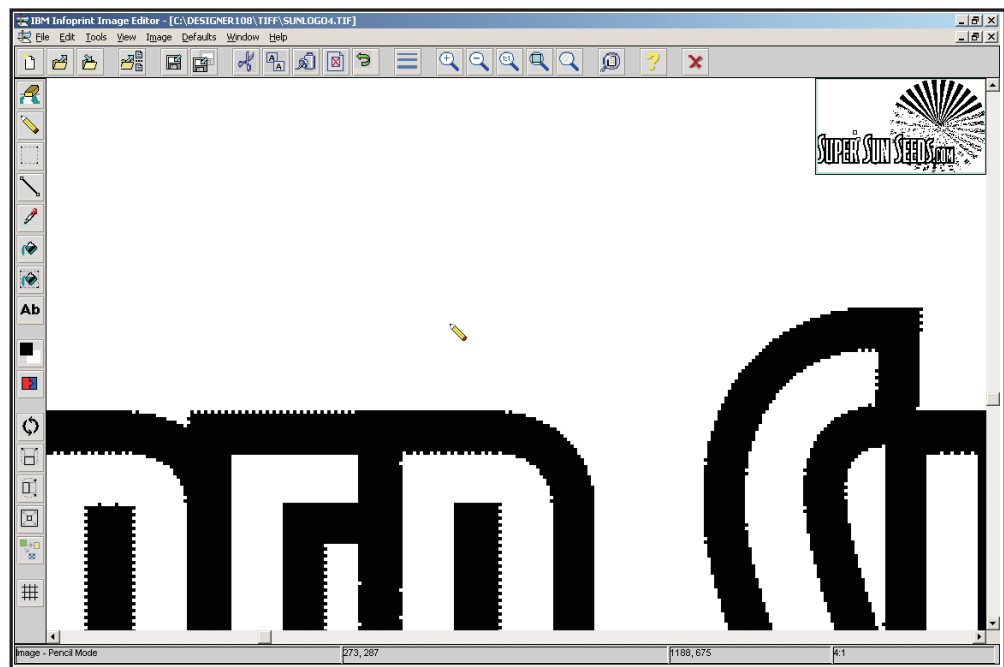


Figure 39. Zooming in on the Super Sun Seeds Logo Image

Figure 39 shows how the tops of the letters "PER" look when they are magnified. Notice that at this level the rounded letters look far less rounded. Any patterns that are not exactly horizontal or vertical will exhibit a "stairstepping" effect at this level of detail. The goal is to smooth out these patterns as much as possible and to remove extraneous pels.

There are several exceptions to the above touch-up process:

- Signatures, as legal images, should only have extraneous black and white pels cleaned up.
- Documents such as check images are taken as is.
- Images that have not been scanned, such as images pulled from clip art or images transformed to AFP with the Infoprint Server image transforms, do not require touch-up.
- Watermarks; grayscale images meant to blend into the background of an electronic document, do not require touch-up.

Image Size:

In some cases, you need the image in several different sizes. For example, a company logo might be used in different sizes in several different documents. For these situations, you might need a page segment in each size that you need. With DDS, you can re-size page segments, although the page segment is sent to the printer each time it is used in a new size. You have these options for creating the page segments in various sizes:

- Work with camera-ready artwork in each size and go through the imaging process (scan, touch-up, upload, object creation) for each size.

- Scan and touch up one size, then use the imaging software to create additional sizes. Because most images are degraded as they are enlarged, you should touch up the largest size you will need. Infoprint Designer resizes image objects on iSeries. Most PC imaging products also provide for resizing image objects.

Image Orientation:

Another image consideration is orientation, or the direction of the image relative to the general direction of the page. The four orientations are 0, 90, 180, and 270 degrees. These orientations are also referred to as *across*, *down*, *back*, and *up*, named by the way text in the image appears with respect to a portrait page in each rotation. For most printers, a page in landscape format is oriented 90 degrees. In this case, an image would also have to be oriented 90 degrees in order to appear correctly.

You might need a page segment in each orientation you need. With DDS, you can rotate page segment, although the page segment is sent to the printer each time it is used in a new orientation. Resource Management Utility creates page segments in any or all of the four orientations. Most PC imaging products also have this capability.

Converting Images to IOCA

The next step in creating an image is to convert it to a usable format. You usually need an IOCA version of the image to use it in OS/400 documents.

At this point, the image is either sitting in the image editor or is in image file format. You can use several different products to get your image in IOCA format:

- Infoprint Designer
- The Infoprint Server image (GIF, TIFF, and JPEG) to AFP transforms, see 91.
- The IBM AFP Printer Driver for Window, see 91.

Creating a Page Segment with Infoprint Designer

You can use Image Editor and Infoprint Designer together to convert a scanned image or a TIFF, BMP, JPEG, GIF, ICO, or PTR image to an AFP IOCA page segment. Follow these steps:

1. In the Image Editor, from the **File** menu select **Open**. Specify the image you want to convert.
2. Make any changes you want.
To use it with Infoprint Designer, it must be black and white. To convert a color image to black and white, from the **Image** menu select **Dither**. In the Dithering parameters window, select **Color B/W** and click **OK**.
3. Save the image in AFP format. From the **File** menu, select **Save as**. In the window that opens, specify the location and file name. For use with Infoprint Designer, it is best to specify *name.300* for the file name. Ensure that **AFP image** is specified for **Save as type**.
4. In Infoprint Designer, from the **File** menu, select **Get image**. Specify your page segment in the window that opens.
5. When you are done working with the project, from the **File** menu select **Upload to AS/400**. On the AS/400 page, ensure that in the Send area, **Page Segments** and **Replace** are specified.

Creating an Image Resource with the Infoprint Server Image Transforms

With Infoprint Server for iSeries, you can transform a GIF, JPEG, or TIFF image file on your workstation to an AFP page segment, overlay, or document.

Before you begin:

You must install the image transforms and complete any setup steps that are appropriate to your installation. For instructions, refer to *Infoprint Server for iSeries: User's Guide*.

To create an AFP page segment from an image file, enter this command from the directory in which the input file resides:

```
xxxx2afp filename -pagetype pseg -o output_filename
```

Where:

xxxx is gif, jpeg, or tiff, as appropriate.

filename is the name of the input file.

output_filename is the name of the output file to create. You can also specify the path to use.

Example:

This example transforms a GIF file called sunlogo.gif into an AFP page segment called sunlogo.300. There is a network drive mapped to \\MyIseries\QDLS. After creation, the page segment is stored in the specified directory in the integrated file system. Enter this command at a DOS prompt:

```
gif2afp sunlogo.gif -pagetype pseg -o \\MyIseries\QDLS\psegs\sunlogo.300
```

For more information about Infoprint Server for iSeries and its image transforms, refer to *Infoprint Server for iSeries: User's Guide* or this Web page:

<http://www.printers.ibm.com/R5PSC.NSF/Web/ipserv400Home>.

Creating an Image Resource with the IBM AFP Printer Driver for Windows

The IBM AFP Printer Driver for Windows creates AFP documents, page segments, and overlays out of any Windows output. This driver is part of iSeries Access and is also available free on an "as is" basis from IBM.

Like any Windows driver, the IBM AFP Printer Driver for Windows uses the Windows GDI (Graphical Data Interface) and maps Windows document elements to AFP. As such, it is an "intelligent" driver. It can map fonts to fonts and lines and boxes to lines and boxes. Where image output is required, the driver uses compressed image. This means that more efficient overlays are created for use on your OS/400.

Before you begin:

If you do not have iSeries Access, you must download and install the IBM AFP Printer Driver for Windows. To download the driver, refer to this Web page: http://www.printers.ibm.com/R5PSC.NSF/Web/afp_printer_driver. Instructions for installing the driver are also available on that Web page.

Steps to create a page segment or overlay:

Note: These are steps for using the Windows Paint program on Windows NT. The process is similar for any Windows image application.

1. From the menu bar, select **File** then **Print....**
2. By **Name**, select **IBM AFP 300** or the AFP device installed.
3. Select **Print to file**.
4. Click **Properties**.
5. Under document options, select **Output Type** then select **Page segment** or **Overlay**, as appropriate.
6. Under document options, select **Clip Limits** then click **Clip Limits....**
7. On the Clip Limits dialog, select **Offset plus size** and click **OK**.
8. Select any other options you want. For example, to select color options, click **Image Options....**
9. For help at any time press **F1**.
10. Click **OK** on the Properties window.
11. Click **OK** on the Print window.
12. When prompted for the file name, specify the name of a specific shared folder if you want to use CL commands to transfer the file to your OS/400. For example, `i:\qd1s\resource\sunlogo`.
13. Click **OK**.

Uploading and Compiling the Image

Once the image file is created on the workstation and resides in an OS/400 folder, it must be transferred to a temporary physical file and then created as an OS/400 page segment or overlay object. You can use one of these methods:

- “Using Infoprint Designer to Upload and Compile an AFP Image”
- “Using AFP Manager to Upload and Compile an AFP Image”
- “Using CL Commands to Upload and Compile an AFP Image” on page 93

Note: When you upload an object to the OS/400, it replaces any file with the same name in the destination library.

Using Infoprint Designer to Upload and Compile an AFP Image

To upload and compile an AFP page segment or overlay with Infoprint Designer, follow these steps:

1. From the **File** menu, select **Upload to AS/400....**
2. The **Upload** dialog opens. The default destination library is specified on the **Preferences** dialog, but you can specify a different library in the **Upload** dialog.

Using AFP Manager to Upload and Compile an AFP Image

Follow these steps to transfer the AFP page segment or overlay to the OS/400 and create it as an OS/400 object.

1. Double-click the iSeries Access icon.
2. Double-click the iSeries Navigator icon.
3. Open a connection to an OS/400.
4. Expand **AFP Manager** then **Resources**.
5. Right-click the image resource type (**Page Segments** or **Overlays**) and select **Import**.
6. Fill in the fields as appropriate. **Source file** is the name and location of the resource on the workstation. Use **Resource name** and **Library** to specify what to name the image resource on the OS/400 and where to store it.
7. For help press F1 or click **Help**.
8. Click **OK**.

Using CL Commands to Upload and Compile an AFP Image

Follow these steps to transfer an AFP resource to your OS/400.

1. Make sure that the PC AFP resource is in `\\system\QDLS\`, where *system* is the name of your OS/400. If the AFP resource is not in that directory, map a network drive to your OS/400 system then use Windows Explorer to move the file.
2. Create a physical file to receive the PC resource. The suggested record length (RCDLEN) is 8201 and level check (LVLCHK) must be *NO (this is the default value). On the command line, specify this:

```
CRTPF FILE(lib/file) RCDLEN(8201)
```

Example:

In this example, the physical file name is PAGESEG and the library is MYLIB:

```
CRTPF FILE(MYLIB/PAGESEG) RCDLEN(8201)
```

3. Copy the PC resource into the physical file on the OS/400 by specifying:

```
CPYFRMPCD FROMFLR(folder) TOFILE(library/file_name) FROMDOC(resource_name) TRNTBL(*NONE)
```

Where:

folder specifies the name of a folder or folder path containing the PC resource. *file_name* and *library* are the same values you specified in the previous step. *resource_name* is the name of the PC resource.

Example:

In this example, the Windows folder RESOURCE contains the resource SUNLOGO.IMG. The resource will be copied into the OS/400 physical file PAGESEG in MYLIB, which was created in step 2.

```
CPYFRMPCD FROMFLR(RESOURCE) TOFILE(MYLIB/PAGESEG) FROMDOC(SUNLOGO.IMG) TRNTBL(*NONE)
```

4. Create the resource.

- To create a page segment, specify:

```
CRTPAGESEG PAGESEG(pagseg_library/pagseg_name) FILE(library/file_name) MBR(file_name)
```

- To create an overlay, specify:

```
CRTOVL OVL(ovl_lib/ovl_name) FILE(library/file_name) MBR(file_name)
```

pagseg_library/pagseg_name and *ovl_lib/ovl_name* specify the name and library of the resource that is created.

library/file_name are the same values you specified in step 2.

Example:

In this example, the page segment SUNLOGO is created in MYLIB from the file PAGESEG, which is also in MYLIB.

```
CRTPAGESEG PAGESEG(MYLIB/SUNLOGO) FILE(MYLIB/PAGESEG) MBR(*PAGESEG)
```

Including Images in your OS/400 Output

The image in OS/400 page segment objects can be built into your application output in a number of ways, including:

- Invoking it in the printer file DDS
- Placing it in an AFP Utilities overlay
- Referencing it in any OS/400 overlay
- Placing it in a print application with the AFP Utilities Print Format Utility
- Specifying it using APU

- Placing it in an Infoprint Designer overlay
- Specifying it within a page definition or form definition
- Using it directly in AFP data streams, such as those created by the AFP Toolbox
- Embedding it in an OfficeVision/400 document. (OfficeVision/400™ is no longer supported. For migration options, see Appendix E, “Migrating from OfficeVision/400,” on page 347.)

Chapter 9. Working with Overlays

An overlay, or electronic form, is a collection of constant data stored as an AFP resource and primarily used in place of preprinted forms. Using overlays instead of pre-preprinted forms gives you greater flexibility, lets you change forms quickly and without waste, eliminates storage space needed for pre-printed forms, and decreases operator intervention needed to print forms. There are several different tools you can use to design overlays. These tools, along with considerations for designing overlays, are discussed in this chapter.

Overlay Design Considerations

With most electronic forms, the design process includes some amount of planning, measuring, and positioning. Typically, you need to:

- Determine the positioning of key elements (you can use a positioning grid overlay to aid in this process).
- Determine the fonts to be used.
- Design and create any images to be placed within the overlay.
- Determine the design sequence, particularly if the overlay has many contiguous lines and boxes.
- Ensure that overlay elements map to the application data.
- If you have a series of forms that are similar, consider breaking out common sections into a partial overlay, or sub-form. This approach will also reduce the maintenance of these forms.

In the Super Sun Seeds example, the application data is produced in traditional row and column format, with standard vertical spacing of 6 lines per inch and 10 characters per inch horizontally. If the overlay is designed with the same measurement grid, then it should match with the application data and be far easier to produce.

Performance Considerations

Printing a full-page image document at high resolutions requires significantly more processing on the part of the OS/400 and the printer than printing one line at low resolution. Performance is not usually a concern in the line-mode printing process, but it is certainly a consideration when you are building and printing electronic documents.

AFP, with its structured data stream, object-based architecture, high-function print managers, and integrated printers, provides a major performance advantage. The overall process of printing is divided into its component parts. For example, IPDS printers offload from the OS/400 the work of translating font, image, and bar code objects into fully resolved (that is, complete bit patterns ready to print) document pages. Additionally, PSF keeps track of what font, image, and overlay resources are already stored in the printer.

Still, it is important to design applications with performance in mind. Because overlays include most of the elements of an electronic output document, performance factors should be considered, but be balanced against the business or design requirements of the application.

The performance notes below are just recommendations, and you might not want to follow all of them. Implementing all of these performance recommendations might create a high-performance electronic overlay, but an ineffective document. In addition, your operational environment should also be considered. If you are printing 1000 Super Sun Seeds invoices, then a few extra seconds at the start of the print job will not matter. However, if your application is printing only one invoice at a time, then every second at job initialization is another second of waiting for that one invoice to print.

These can all affect performance:

Fonts

When OS/400 host-resident fonts are downloaded to the printer, time and printer memory usage are a factor. Large raster fonts and large numbers of fonts can slow performance. Consider doing the following:

- Minimize the number of fonts used. Document effectiveness normally suffers when more than 6-8 fonts are used on a page.
- Standardize fonts used across applications. Setting some organizational standards with respect to fonts will improve both performance and document effectiveness.
- Use outline fonts. Outline fonts have one scalable object for all point sizes. Raster fonts have a separate object for each point size. As more outline fonts become available on OS/400 (both host and printer-resident), consider using them instead of their raster equivalents.
- Use printer-resident fonts. Printer-resident fonts eliminate the download time that host-resident fonts require. Refer to *Printer Device Programming* for more information about printer-resident fonts.
- Use font capturing. Font capturing enables the printer to dynamically store a host-downloaded font. The captured font then acts like a printer-resident font. Printing performance is improved by eliminating subsequent font downloads to the printer. The performance improvement will be especially noticeable if you are using double-byte fonts. "Capturing Fonts" on page 83 describes how to enable this function.

Images

Because of sheer size, images can affect performance in transmission, printing, and printer memory requirements.

Consider doing the following:

- Minimize the use of images. Do not use larger size page segments than the document requires.
- Use compressed images when possible. Be sure that you have created compressed images when creating IOCA page segments. Note that some older printers do not support compressed images.
- Use the same image across applications. If possible, standardize your logo to one size so that it may be used with little performance cost across applications.

Object Size

Certain design or object creation processes may generate larger than expected overlay or image objects. For example, a large page segment might indicate that it was created without image compression, or some overlay characteristics such as shading may affect object size or object performance.

Designing Overlays Using Infoprint Designer

Infoprint Designer's overlay capabilities are all you need to create electronic forms. You can either draw the forms with the Overlay Editor, trace over an existing scanned (TIFF) image of your pre-printed form, or import an AFP overlay object from your iSeries. Then, use Infoprint Designer to create overlay projects containing one or more electronic forms created with the Overlay Editor. For more details, including an example, see Chapter 17, "Using Infoprint Designer," on page 197.

Designing Overlays Using DDS

DDS provides most of the elements used in designing electronic forms, such as fonts, lines, boxes, images, and bar codes. For some applications, it may make sense just to use DDS to build all or part of the overlay. For example, simple forms, such as letterheads, are easily created with DDS. A more common application would involve electronic documents in which portions of the document are handled by an overlay and portions are handled by DDS. You also can use DDS to construct "floating" overlays. A floating overlay is one in which part of the form is placed at different locations on the output page. See "Floating Document Elements" on page 135 for an example of a floating overlay.

Creating Overlays from Existing Images

You can create overlays with a Windows application then transfer them to the OS/400 for use with iSeries applications. Figure 40 on page 98 shows INVBAC, the terms and conditions overlay for the Super Sun Seeds invoicing application.

Terms and Conditions

The sale of the products described herein shall be governed by the terms and conditions contained in any written contract currently in effect between Buyer and Seller covering such sale. If there is no such contract, then Seller hereby offers to sell such products to Buyer only upon the terms set forth herein.

1. Seller's standard prices in effect at the time of shipment will govern the sale of the products described herein. Terms of payment on any approved order are net thirty (30) days from the date of invoice unless otherwise specifically stated. All shipments, unless specifically provided, shall be f.o.b. place of manufacture or warehouse location indicated herein. The price includes cost of packaging for domestic shipment, unless otherwise stated. An additional charge will be made for special domestic or export packing if this involves greater expense. Shipments will be insured at the expense of the Buyer unless Buyer specifically requests that shipments not be insured.
2. Seller reserves the right, among other remedies atherto, to terminate this contract or suspend further deliveries under it in the event Buyer fails to pay for any one shipment when same becomes due. Should Buyer's financial responsibility become unsatisfactory to Seller, cash payments or satisfactory security may be required by Seller for future deliveries and for goods theretofore delivered.
3. In addition to the purchase price, Buyer shall pay to Seller the amount of all taxes, excises or other charges (except taxes on or measured by net income) that Seller may be required to pay to any Government (national, state or local) with respect to the production, sale or transportation of any product delivered hereunder, except where the law otherwise provides. Sixty (60) days written notification must be given to make any changes to delivery schedule.
4. Seller warrants that products delivered hereunder will conform to the description on the face of this document and meet any specifications set forth or incorporated by reference herein and will be adequately contained, packaged and labeled and conform to any promises and affirmations of fact made on the container and label. Seller further warrants any such product against defects in workmanship or materials which develop or become evident within 12 (9 months OEM) after shipment of the product by Seller provided Seller's responsibility under this warranty shall be limited to the repair or if Seller so elects the replacement of the defective product or component thereof and that said warranty is subject to the following exceptions and conditions:
 - (a) All items claimed to be defective must be returned to Seller, transportation charges prepaid and will be returned to Buyer transportation charges collect unless found to be defective in which case Seller will pay all transportation charges.
 - (b) Seller's warranty will not apply to items which have been modified or repaired without Seller's written consent by persons other than Seller's authorized service personnel.
5. It should be noted that Seller has used the powerful tools of AS400 Advanced Function Printing to create Seller's invoice, and despite the arcane terms and conditions which are presently being read, the invoice is a very effective business document. Any attempt to mar the appearance of this invoice should be discouraged.
6. The validity, interpretation and performance of the terms hereof with respect to any product delivered (or to be delivered) hereunder shall be governed by the law of the State of Colorado. . .
7. No modification or waiver of the terms here of shall be binding upon Seller unless approved in writing by one of Seller's Officers or Marketing Managers or shall be affected by acknowledgment or acceptance of purchase order forms containing other or different terms whether or not signed by authorized representative of Seller. (e) Seller does not warrant items which have been damaged due to negligence or misuse.
8. THERE ARE NO EXPRESS WARRANTIES BY SELLER OTHER THAN THOSE SPECIFIED IN THIS PARAGRAPH. NO WARRANTIES BY SELLER (OTHER THAN WARRANTY OF TITLE AS PROVIDED IN THE UNIFORM COMMERCIAL CODE) SHALL BE IMPLIED OR OTHERWISE CREATED UNDER THE UNIFORM COMMERCIAL CODE, INCLUDING BUT NOT LIMITED TO WARRANTY OF MERCHANTABILITY IN OTHER RESPECTS THAN SPECIFIED IN THIS PARAGRAPH AND WARRANTY OF FITNESS FOR A PARTICULAR PURPOSE. Without limiting the generality of the foregoing, Buyer assumes all risk and liability for the results obtained by the use of any products delivered hereunder in combination with other articles or material in the practice of any process.
9. NO CLAIM of any kind with respect to any product specified herein, whether as to product delivered or for nondelivery of product and whether or not based on negligence, SHALL BE GREATER IN AMOUNT than the purchase price of the product in respect of which such claim is made. In no event shall either party be liable for special indirect or consequential damages whether or not caused by or resulting from the negligence of such party.
10. It is expressly understood that any technical advice furnished by Seller with reference to the use of its products is given gratis and Seller assumes no obligation of liability for the advice given or results obtained, all such advice being given and accepted at Buyer's risk.
11. In the event Buyer fails to fulfill Seller's terms of payment for the products specified herein, or in case Seller shall have any doubt at any time as to Buyer's financial responsibility, Seller may decline to make further deliveries except upon receipt of cash or satisfactory security.

Figure 40. Terms and Conditions Overlay Created in a Windows Application

This overlay was created using a Windows application. To maximize the performance and efficiency of the overlay, it was built using Expanded Core fonts (TrueType or Adobe Type Manager - ATM) that map directly into AFP fonts. Specifically, the text is in Helvetica Italic Medium and Helvetica Italic Bold typefaces in whole number point sizes.

If you created the overlay on your workstation as a GIF, JPEG, or TIFF file, you can use the Infoprint Server image transforms to transform the image file to an AFP overlay. However, this creates an all-image overlay, so this method is best for an overlay that is not all text. For information about using the Infoprint Server image transforms, see "Creating an Image Resource with the Infoprint Server Image Transforms" on page 91 or refer to *Infoprint Server for iSeries: User's Guide*.

Alternatively, you can use the AFP print driver to create the AFP overlay. See "Creating an Image Resource with the IBM AFP Printer Driver for Windows" on page 91 for instructions.

You can then transfer the overlay to your OS/400 by using AFP Manager or OS/400 commands. "Uploading and Compiling the Image" on page 92 contains these instructions.

Using Overlays with OS/400 Output

Overlays can be included in OS/400 print applications using many different techniques, including:

- Specifying them as front or back overlays in the printer file
- Specifying them directly with the DDS OVERLAY keyword
- Specifying them with APU
- Specifying them with AFP Toolbox
- Placing them in a print application with the Print Format Utility (a module of AFP Utilities)
- Specifying them within a page definition and form definition object
- Using them with an Infoprint Designer layout
- Using them as cover sheets in Facsimile Support for iSeries (*AFPDS spooled files only)
- Embedding it in an OfficeVision/400 document. OfficeVision/400 is no longer supported. For migration options, see Appendix E, "Migrating from OfficeVision/400," on page 347.

Chapter 10. Working with Bar Codes

Bar codes represent characters with sets of parallel bars or in a two-dimensional matrix. They can be an essential part of your output. “Bar Codes” on page 67 introduces bar codes.


All of the major bar code symbologies are supported on OS/400 directly, using BCOCA. Most of the AFP application enabling tools, such as DDS, provide an interface for specifying bar codes. The figures in this chapter are examples of bar code printing using the DDS BARCODE keyword.

This chapter describes how to best implement bar codes in your installation. It lists considerations for deciding what kind of bar code to use and describes several different ways to specify bar codes in your documents.

IPDS provides an interface to its printers that enables the mixture of different data types, including text, image, graphics, and bar codes. The data carried by each of these types is defined by a data object. The bar code object contains BCOCA data controls that describe a specific bar code, supply the data to be coded, and enable the IPDS printer to decide what is needed to construct and print the bar code.

Because the printer constructs the bar code, the same IPDS commands sent to IPDS printers from different technologies produce similar output. This lets you write the bar code application with little concern about what IPDS printer will be used for output.


Figure 41 on page 102 shows Postnet and Interleaved 2 of 5 bar codes on the Super Sun Seeds sample invoice.



400 CPU Parkway
Vegetation, NJ 55090

Office: 555-499-2367
Fax: 555-415-9794

Bar Code




LOS ARBOLES DEL MUNDO
32483 ARBOL LANE
MESA VERDE
IL 65478-9390

SAME

Customer Number: 141
Invoice Number: 31341
Invoice Date: 7/28/02
Payment Date: 8/28/02

Ship Via:
Ship Date: 7/28/02
Terms: N10
Rep: MICHELE GOODACRE

UOM	Item #	Item Description	Price	Extension
1000 BX	56413213	POT POT	7.65	7,650.00
45 BZ	11005015	CHANTENAY SEEDS	2.19	98.55
900 EA	00001200	ARBOLES DEL SUR	45.00	40,500.00
98 PK	84512023	OREGON SPRING TOMATO SEED	.97	95.06
4 BX	11057893	AFRICAN DAISY, SEEDS	2.35	9.40
951 CT	11005011	LASSO RED SEEDS	892.23	48,510.73
46 DZ	11005014	SCARLET NANTES SEEDS	5.90	271.40
100 EA	31321655	SEMILLAS DEL SUS SOMBEROS	24.95	2,495.00



*Thank You
Because almost half of
your order was Lasso Red
Seeds, you will receive
a 10% discount on your
next order.*

Total Due

\$99,630.14


Return this tear-off strip with your payment.

Payment is due by: 8/28/02

LOS ARBOLES DEL MUNDO
32483 ARBOL LANE
MESA VERDE
IL 65478-9390

Make Checks Payable to: **Super Sun Seeds**

Amount Due is: \$99,630.14



0001410099630146

Bar Code

Figure 41. Bar Codes Printed on an IPDS Printer

Making a Bar Code Decision

Before deciding which bar code you want to use, there are several things you should ask yourself. The information to help you decide the answers are presented in the following sections.

Which Symbology?

This decision is sometimes the easiest to make because the industry or application may dictate which symbology must be used. The retail industry, for example, uses UPC or Code 128.

If you can make your own symbology decision, factors such as character set (do you have numeric only or alphanumeric data to encode), density, and space available on the printed document must be considered.

What Density?

The bar code density is determined by the width of the bars and spaces. The narrower the bars and spaces, the higher the density. You must consider these factors before deciding which density to use: How much space is available for the bar code? Will the bar code be read by a hand-held scanner used close to the bar code or will it be read at a distance? Bar codes read from a distance require a lower density bar code with wider bars.

What density is the printer capable of printing? Impact printers tend to print lower density bar codes than laser printers. Do you have enough space on the document for the density of the bar code the printer is capable of printing? Is there a sufficient quiet zone?

Will your scanner be able to reliably read the bar code density? Most scanners scan with either a small, medium or large scanning dot radius. If your scanner has a large scanning dot and the bar code has a high density or vice-versa, the bar code might not scan reliably.

What Height?

Will the bar code be read by a hand-held wand? If it will, the aspect ratio might need to be changed to produce a bar code with taller bars.

Using picket fence orientation as an example, it is more difficult to scan a long, short bar code with a wand than with a laser gun. How much space is available? Do bearer bars have to be included in the height?

What Orientation?

What space on the document is available? Does available space dictate bar code orientation? If the bar code is being mechanically scanned, is there a requirement for a specific orientation? Is there more than one bar code on the label or document and should each bar code be in a different orientation to simplify scanning?

Laser printers sometimes experience a quality reduction called *keystoning*. Keystoning is a condition where the leading or trailing edge of a character (relative to the movement of the paper through the printer) tends to fade away and does not have a sharp boundary. This is usually noticeable on the bottom edge of a line of text or a bar.

If the bar code is printed with the picket fence orientation, keystoning makes little difference because the loss of quality is at the top or bottom of the bars and it will have minimal affect on scanning the bar code.

If the bar code is printed with the ladder orientation, then the keystoning can affect the leading or trailing edge of each bar and cause first read rate problems.

What Print Volume?

How many bar codes must be printed? Do you need to print them multiple-up? How does this affect printer performance?

What Print Window?

How much time is available for bar code printing? Can the bar codes be printed batch or must they be printed on demand? Are multiple printers necessary for small print windows or for backup requirements?

Specifying Bar Codes with PPFA

To use PPFA to specify 2D bar codes, you must install PTF SF66716. The below source creates a page definition that will generate PDF 417, MaxiCode, and DataMatrix 2D bar codes.

```
5722WDS V5R2M0 020719          SEU SOURCE LISTING      13:04:12          PAGE    1
SOURCE FILE . . . . . FVT/QPPFASRC
MEMBER . . . . . PDTWOD
```

```

SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ..
100 /* This PAGEDEF utilizes PPFA record formatting to print */
200 /* PDF 417, MaxiCode and DataMatrix bar codes. */
300
400 PAGEDEF PDTWOD
500 WIDTH 8.50 IN HEIGHT 11.00 IN DIRECTION ACROSS
600 REPLACE YES PELSPERINCH 300 PAGECOUNT CONTINUE 1 ;
700 FONT BOLD CB10; /* Roman Bold 10 char/inch */
800
900 PAGEFORMAT TWOD;
1000
1100 LAYOUT C'PDF1' /* PRINT PDF417 BAR CODE */
1200 PAGEHEADER NEWPAGE DIRECTION ACROSS;
1300
1400 FIELD
1500 START 1 LENGTH 10 ALIGN LEFT
1600 POSITION 1.00 IN 7 IN DIRECTION ACROSS
1700 BARCODE 2Dpdf1 TYPE 2dpdf417
1800 MOD 0 HRI ON HEIGHT .12 IN
1900 BCXPARMS SIZE 20 10 SECLEV 1 E2A;
2000
2100 LAYOUT C'MXC1' /* PRINT MAXICODE BAR CODE */
2200 PAGEHEADER NEWPAGE DIRECTION ACROSS;
2300
2400 FIELD
2500 START 1 LENGTH 10 ALIGN LEFT
2600 POSITION 1.00 IN 7 IN DIRECTION ACROSS
2700 BARCODE 2DMXC1 TYPE 2DMAXI
2800 BCXPARMS MODE 4 ZIPPER NOESC E2A;
2900
3000 LAYOUT C'DMX1' /* PRINT DATAMATRIX BAR CODE */
3100 PAGEHEADER NEWPAGE DIRECTION ACROSS;
3200
3300 FIELD
3400 START 1 LENGTH 10 ALIGN LEFT
3500 POSITION 1.00 IN 7 IN DIRECTION ACROSS
3600 BARCODE 2DDMX1 TYPE 2DMATRIX
3700 BCXPARMS RDRPROG NOESC E2A;
3800
3900 FORMDEF FDTWOD
4000 REPLACE YES PRESENT PORTRAIT DIRECTION ACROSS
          * * * * E N D O F S O U R C E * * * *

```

Specifying Bar Codes with DDS

This example shows the printer file DDS (BARSMP) source that produced the samples in this chapter. It also generates bar codes not pictured here. Creating bar codes is relatively easy, you only need to specify the application field, the BARCODE keyword, and the type of bar code desired.

```

5722WDS V5R2M0 020719 SEU SOURCE LISTING 07/17/02 22:46:07 1
SOURCE FILE . . . . . FVT/QDSSRC
MEMBER . . . . . BARFVT
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
100 A*BAR CODES USING DDS ON THE AS/400
200 A R TITLE LPI(8)
300 A 11 TLE1 50 18SKIPB(4)
400 A SPACEA(4)
500 *
600 A R BARCDE LPI(8)
700 A N24 HEAD2 30 48
800 A HEAD1 30 9
900 A SPACEA(1)
1000 A 12 CD12@1 11S 11BARCODE(UPCA 4)
1100 A 12 CD12@2 12S 50BARCODE(EAN13 4)
1200 A 12 SPACEA(9)

```

```

1300      *
1400     A 13      CD13@1      10S      11BARCODE (INTERL20F5 4)
1500     A 13      CD13@2      10S      50BARCODE (MATRIX20F5 4)
1600     A 13
1700      *
1800     A 14N24   CD14@2      10A      50BARCODE (CODEABAR 4)
1900     A 14      CD14@1      10S      11BARCODE (INDUST20F5 4)
2000     A 14
2100      *
2200     A 15      CD15@1      11A      11BARCODE (CODE30F9 4)
2300     A 15      CD15@2      10S      50BARCODE (MSI 4)
2400     A 15
2500      *
2600     A 16      CD16@1      11A      11BARCODE (CODE128 4)
2700     A 16      CD16@2      8A       66BARCODE (CODE128 9 *VRT +
2800     A
2900     A 16
3000      *
3100     A 17      CD17@1      5S       11BARCODE (POSTNET)
3200     A 17
3300     A 18      CD18@1      9S       11BARCODE (POSTNET)
3400     A 18
3500     A 19      CD19@1      11S      11BARCODE (POSTNET)
3600      *
3700     A 20      PDF417      50A      11BARCODE (PDF417 (.12 *UOM) +
3800     A
3900     A 20
4000      *
4100     A 22      MAXI       50A      50BARCODE (MAXICODE 1 +
4200     A
4300     A* 22
4400      *
4500     A 22      MATRIX     50A      11BARCODE (DATAMATRIX 1 +
4600     A
4700     A

```

* * * * E N D O F S O U R C E * * * *

Specifying Bar Codes in Infoprint Designer

You can specify bar codes with Infoprint Designer while you design your layout. Follow these steps to specify a bar code:

1. From the project window, click the bar code tool on the toolbar. The bar code tool is circled in Figure 42.

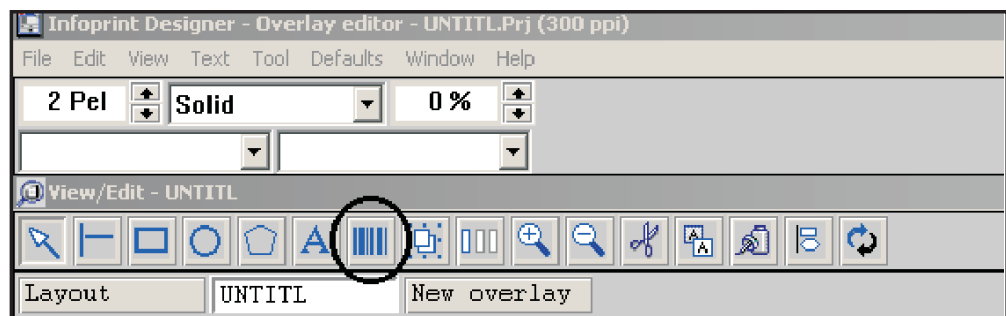


Figure 42. The Bar Code Tool Button

2. The **Bar code type definition** window opens, shown in Figure 43 on page 106. Use the pages on this window to specify the bar code's name, type, modifier,

HRI font (if you want HRI), color, and size. Click **OK**. This defines a new bar code for the layout.

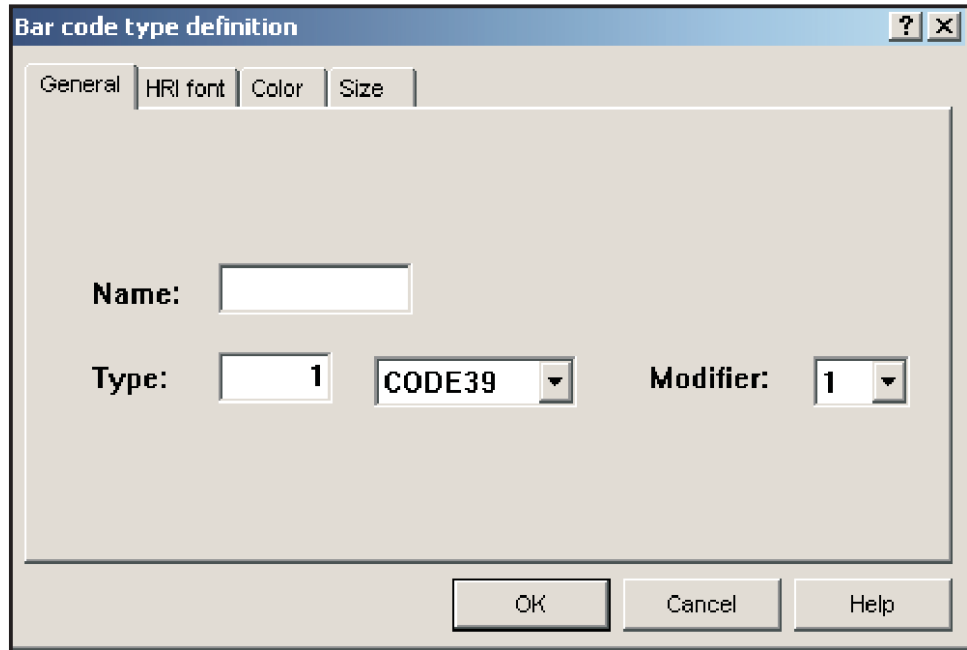


Figure 43. Defining the Bar Code

3. The **Bar code field properties** window opens, shown in Figure 44. Use the pages on this window to specify the properties for this field. You can specify the string to encode (1234567 is specified in Figure 44), whether you want HRI and where you want it placed, and the bar code's orientation. Click **OK**.

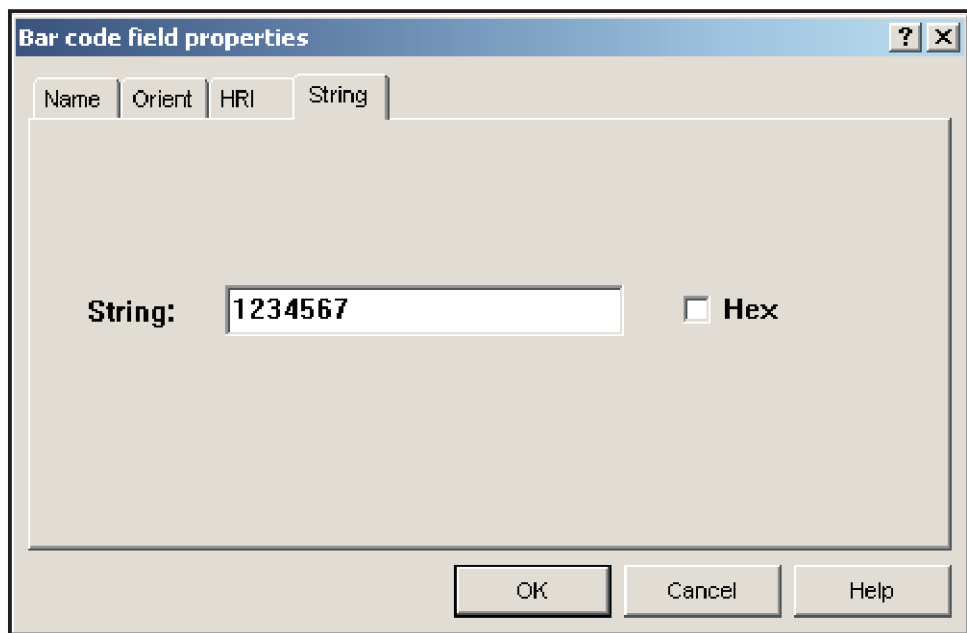


Figure 44. Specifying the Bar Code Properties

4. The bar code is placed on the layout.

Specifying Bar Codes in Print Format Utility

The Print Format Utility of the IBM AFP Utilities can be used to print bar codes. You can specify bar codes when you format data with the Print Format Utility. Bar codes supported by AFP Utilities are the same as those supported by DDS.

Use the Change Bar Code Detail panels to specify bar codes Print Format Utility.

```

Change Bar Code Detail
Mark: *C003           Measurement method: Inch
Type choices, press Enter.
Position:
  Across . . . . . 1.40           0.00-22.75
  Down . . . . . .66           0.00-22.75
  Bar code type . . . . . 9           1=CODE30F9, 2=MSI, 3=UPC-A
                                           4=UPC-E, 5=EAN-8, 6=EAN-13
                                           7=INDUST25, 8=MATRIX25
                                           9=INTERL25, 10=CODABAR
                                           11=CODE128, 12=POSTNET
                                           13=ROYAL MAIL, 14=JPBC
                                           15=AUSTRALIA POST

  Bar code data . . . . . &ITM1.

  Element . . . . .

F3=Exit  F4=Prompt  F5=Refresh  F6=Change measurement method  F12=Cancel
F4 for list
Name
More...

```

Figure 45. Change Bar Code Detail Display

```

Change Bar Code Detail
Mark: *C003           Measurement method: Inch
Type choices, press Enter.
Degree of rotation . . . . . 0           0, 90, 180, 270
Color . . . . . *DEFAULT           *DEFAULT, 1=Blue, 2=Red
                                           3=Magenta, 4=Green, 5=Cyan
                                           6=Yellow, 7=Brown, 8=Black
                                           Y=Yes, N=No

  Check digit . . . . . N           1=Below, 2=Above, 3=No
  HRI . . . . . 3
  HRI font . . . . . *DEFAULT           *DEFAULT, 1-8, F4 for list
  Module width . . . . . *DEFAULT           *DEFAULT, 0.001-0.254
  Element height . . . . . *DEFAULT           *DEFAULT, 0.01-22.75
  Wide to narrow ratio . . . . . *DEFAULT           *DEFAULT, 2.00-3.00

F3=Exit  F4=Prompt  F5=Refresh  F6=Change measurement method  F12=Cancel
Bottom

```

Figure 46. Change Bar Code Detail Display

Usually, you specify the positioning. Optionally, you can specify changes to the bar code characteristics, such as height, width factors, and whether human-readable information (HRI) is printed above or below the bar code.

If bar codes are required in an AFP Utilities application, but the target printer does not support the IPDS bar code object, AFP Utilities generates draw rule bar codes that do not require BCOCA support. AFP Utilities makes the draw rule decision based on the printer type selected in the form specification. Bar codes can also be printed as fonts with bar code fields in form layouts as font changes rather than bar code elements.

Print Format Utility bar codes can be either fixed or variable (dynamic). Variable bar codes can change with each record, producing a unique bar code for each form.

Specifying Bar Codes in APU

APU has an interface to bar codes very similar to Print Format Utility's. These examples show the displays used to define bar code data. In this display, we mapped a field in the SCS output file (in this case, the zip code) to bar code.

```

Define Field Mapping
.....
Edit Bar Code Mapping
:
: Type choices, press Enter.
:
: From Row / Column : 15 / 16
: Bar code data . . : 45789-2637
:
: Position across . . 1.25 *INCH Value
: Position down . . . 1.9 *INCH Value
:
: Rotation . . . . . *DEFAULT *DEFAULT, 0, 90, 180, 270
: Color . . . . . *PRTDEF *PRTDEF, Value F4 for list
:
: Bar code type . . . 12 Value F4 for list
:
:
:
:
: F4=Prompt F12=Cancel F15=Repeat More...
: F16=Delete F22=Set Units
:
:
:
.....

```

Figure 47. Defining a Bar Code in APU

You can specify type, positioning, size, and other factors to control how the bar code is presented.

```

Define Field Mapping
.....
Edit Bar Code Mapping
:
: Type choices, press Enter.
:
: Bar code type . . : 12 - POSTNET
: Bar code data . . : 45789-2637
:
: Length . . . . . 10 1-50
:
: POSTNET Type . . . 4 1=ZIP Code,
: 2=ZIP+4 Code,
: 3=Advanced Bar Code,
: 4=Variable length data
:
:
:
:
: Bottom
: F12=Cancel
:
:
:
.....

```

Figure 48. Defining a Bar Code in APU

Chapter 11. Working with Object Containers

Object containers are used to carry non-OCA objects, such as EPS and PDF, in an AFP data stream. There are different ways to include them in your data, depending on your installation and how you want to use them.

The DDS keyword AFPRES lets you reference a print resource stored in the integrated file system, including an object container.

Object containers can be mapped using the MO:DCA Map Data Resource (MDR) structured field. Mapping causes PSF for iSeries to send the object container to the printer to be used multiple times, just like it does fonts, overlays and page segments. If you want them downloaded once to the printer and used repeatedly on subsequent pages or overlays, they should be included as *data object resources*. Data object resources are those object containers (or IOCA objects) that are sent to the printer in the IPDS home state. Object containers sent to the printer in this state are saved as data object resources to be used later while processing a page or overlay.

You can specify at most 10 object containers per page of data. When using object containers, you must ensure that the device to which you send the data supports the type of object included. Also, the OS/400 searches for object containers differently than other objects. The search method is described in the documentation of the DDS AFPRSC keyword on the iSeries Information Center.

Refer to *Printer Device Programming* for more information about object containers.

Part 2. Super Sun Seeds Case Study

In this part, various tools that can be used to produce the Super Sun Seeds invoice as an electronic output application are introduced. Each chapter describes how to use a specific tool, then shows how to use the tool to work with the Super Sun Seeds case study. The part contains these chapters:

- Chapter 12, “Introduction to the Super Sun Seeds Case Study,” on page 113
- Chapter 13, “Using Printer Files,” on page 121
- Chapter 14, “Using Data Description Specifications,” on page 125
- Chapter 15, “Using Advanced Print Utility,” on page 153
- Chapter 16, “Using Page Printer Formatting Aid,” on page 183
- Chapter 17, “Using Infoprint Designer,” on page 197
- Chapter 18, “Using IBM AFP Toolbox,” on page 217
- Chapter 19, “Using the Print Format Utility,” on page 223

See Chapter 25, “Deciding on an Output Strategy and Set of Tools,” on page 287 for additional guidance about which document tool would be best for your environment or application.

INVPRE Application Program Example

The application program producing the Super Sun Seeds invoice data is INVPRE, shown in this example.

```

5722WDS V5R2M0                      SEU SOURCE LISTING    10/10/02 17:47:39    1
SOURCE FILE . . . . . SAMPLER/QRPGSRC
MEMBER . . . . . INVPRE
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
100 * INVPRE - Super Sun Seeds - Sample of a Preprinted Invoice
200 *
300 * This program reads a transaction file (SEEDDETL), retrieves info
400 * from item master (SEEDITEM) and customer master (SEEDCUST), then
500 * prints invoices.
600 *
700 * Invoice detail lines are processed until either end of customer order
800 * or lines=25, at which time the invoice page is printed. At end of
900 * order, a check is made to see if there are more than 18 detail lines
1000 * to print. If there are, that would not leave room in the invoice body
1100 * to print the customer offer. In this case, the page is printed, and a
1200 * new final page is built with the offer and the payment coupon.
1300 *
1400 * The offer is a message based on item sales in the customer master.
1500 *
1600 *
1700 FSEEDDETLIP E          K          DISK
1800 FSEEDITEMIF E         K          DISK
1900 FSEEDCUSTIF E         K          DISK
2000 FINVPRE  0  E                PRINTER
2100 *
2200 E                      WDS      1  40 24                Text for Offer
2300 *
2400 IINVCDETL  01
2500 I                      STNAME          32
2600 I                      CUST# L1
2700 ICUSTDATA  02
2800 IITEMDATA  03
2900 *
3000 IBARTOT     DS
3100 I                      1  60CUST#
3200 I                      7  152TOTDUE
3300 *
3400 IPAYDAA     DS
3500 I                      1  20PAYMO
3600 I                      3  40PAYDA
3700 I                      5  60PAYYR
3800 *
3900 C          *IN90      IFEQ '0'                First Pass Only
4000 C                      SETON                90
4100 C                      EXSR DATESR
4200 C                      ENDIF
4300 *
4400 * Start of customer order
4500 *
4600 C          *INL1      IFEQ '1'                Start Customer
4700 C          CUST#      ADD 31200      INVC#  60      Invent Invoice#
4800 C                      Z-ADD0      TOTDUE 92      Reset Totals/Ctrs
4900 C                      Z-ADD0      ITMCNT 30
5000 C                      Z-ADD0      PAGCNT 30
5100 C          CUST#      CHAINSEEDCUST          20
5200 C  32                MOVEL'  SAME'STNAME
5300 C                      Z-ADDZIP      ZIPPN  90
5400 C*
5500 C                      WRITEINVTOP
5600 C*
5700 C                      ENDIF
5800 C*

```


5900	C*	Detail Processing			
6000	C*				
6100	C	*IN01	CABNE'1'	ENDDT	Item Processing
6200	C		ADD 1	ITMCNT	
6300	C	ITM1	CHAINSEEDITEM		21
6400	C*				
6500	C	QTY1	MULT SELPRC	EXTPRC 72	
6600	C		Z-ADDQTY1	QTY 40	
6700	C	EXTPRC	ADD TOTDUE	TOTDUE	
6800	C*				
6900	C		WRITEDETLIN		Write Detail Line
7000	C*				
7100	*	Check for end of space on page			
7200	*				
7300	C	ITMCNT	IFEQ 25		Full 1st Page
7400	C		ADD 1	PAGCNT	
7500	C		SETON		31
7600	C		WRITEPAGEOF		Pg 1 of 2 Msg
7700	C		WRITEINVTOP		Invoice Heading
7800	C		SETOF		31
7900	C		Z-ADD0	ITMCNT	
8000	C		ENDIF		
8100	C*				
8200	C	ENDDT	TAG		
8300	C*				
8400	C*	End of customer order			
8500	C*				
8600	CL1	ITMCNT	IFGT 18		No Room for Offer
8700	CL1		ADD 1	PAGCNT	
8800	CL1		SETON		31
8900	CL1		WRITEPAGEOF		Pg 1 of n
9000	CL1		WRITEINVTOP		Invoice Heading
9100	CL1		SETOF		31
9200	CL1		ENDIF		
9300	C*				
9400	C*	Payment coupon fields			
9500	C*				
9600	CL1		MOVE BARTOT	BARPRC 150	Load Totals
9700	CL1		Z-ADDTOTDUE	TOTD@2 92	
9800	CL1		MOVE NAME	NAME@2 25	
9900	CL1		MOVE STREET	STRE@2 25	
10000	CL1		MOVE CITY	CITY@2 25	
10100	CL1		MOVE STATE	STAT@2 2	
10200	CL1		MOVE ZIP	ZIP@2 9	
10300	CL1		ADD 1	PAGCNT	
10400	C*				
10500	C*	Write Offer			
10600	C*				
10700	CL1		EXSR OFFSR		
10800	CL1 30		WRITEOFFER		Write Offer and
10900	CL1		WRITEINVBOT		Invoice Totals
11000	CL1		SETOF		30
11100	C*				
11200	C*	Set up Date			
11300	C*				
11400	CSR	DATESR	BEGSR		
11500	CSR		Z-ADDUDAY	PAYDA	
11600	CSR	UMONTH	ADD 1	PAYMO	
11700	CSR	PAYMO	IFGT 12		
11800	CSR		Z-ADD1	PAYMO	
11900	CSR	UYEAR	ADD 1	PAYYR	
12000	CSR		ELSE		
12100	CSR		Z-ADDUYEAR	PAYYR	
12200	CSR		ENDIF		
12300	CSR		MOVE PAYDAA	PAYDAT 60	
12400	CSR		Z-ADDPAYDAT	PAYDA@ 60	
12500	CSR		ENDSR		

```

12600      *
12700      * Set up Special Offer
12800      *
12900      CSR      OFFSR      BEGSR
13000      CSR      SLSSSED    IFGE 500      Seeds
13100      CSR      Z-ADD1      IX      20
13200      CSR      SETON      30
13300      CSR      GOTO WRTOFR
13400      CSR      ENDIF
13500      CSR      SLSFRT      IFGE 500
13600      CSR      Z-ADD7      IX      30      Fruit
13700      CSR      SETON
13800      CSR      GOTO WRTOFR
13900      CSR      ENDIF
14000      CSR      SLSSUP      IFGE 500      Trees
14100      CSR      Z-ADD13     IX
14200      CSR      SETON      30
14300      CSR      GOTO WRTOFR
14400      CSR      ENDIF
14500      CSR      SLSCHM      IFGE 500      Stuff
14600      CSR      Z-ADD19     IX
14700      CSR      SETON      30
14800      CSR      GOTO WRTOFR
14900      CSR      ENDIF
15000      C*
15100      CSR      GOTO ENDOFR
15200      C*
15300      CSR      WRTOFR      TAG
15400      CSR      MOVE WDS,IX  OFFR@1 24      Build Offer Text
15500      CSR      ADD 1      IX
15600      CSR      MOVE WDS,IX  OFFR@2 24
15700      CSR      ADD 1      IX
15800      CSR      MOVE WDS,IX  OFFR@3 24
15900      CSR      ADD 1      IX
16000      CSR      MOVE WDS,IX  OFFR@4 24
16100      CSR      ADD 1      IX
16200      CSR      MOVE WDS,IX  OFFR@5 24
16300      CSR      ADD 1      IX
16400      CSR      MOVE WDS,IX  OFFR@6 24
16500      CSR      ENDOFR      ENDSR
16600      C*
16700      *
16800      ** WDS      WORDS FOR OFFER
16900      Thank You .....
17000      Because you have ordered
17100      over $500 of seeds this
17200      year, on your next seed
17300      order you will receive
17400      a 10% discount.
17500      Thank You .....
17600      Because you have ordered
17700      over $500 of fruit this
17800      year, on your next fruit
17900      order you will receive
18000      a 10% discount.
18100      Thank You .....
18200      Because you have ordered
18300      over $500 of trees this
18400      year, on your next tree
18500      order you will receive
18600      a 10% discount.
18700      Thank You .....
18800      Because you have ordered
18900      over $500 of stuff this
19000      year, on your next stuff
19100      order you will receive

```

19200 a 10% discount.
19300

* * * * * E N D O F S O U R C E * * * * *

INVPRE Application Program General Processing Flow

The general processing flow for the source program above (INVPRE) is as follows:

1. Customer order (file SEEDDETL) is processed.
2. At the beginning of each customer order, customer information is retrieved from the customer master (file SEEDCUST). This provides the bill to and ship to information for the top half of the invoice page.
3. For each line item, item master information is retrieved from the item master (file SEEDITEM).
4. Line extensions are calculated, and order totals are accumulated.
5. An invoice detail line is printed. The preprinted invoice accommodates up to 25 line items per page.
6. If a customer order contains more than 25 line items, the message *Continued* is printed in the total box, and the invoice is continued on the next form.
7. At the end of a customer invoice, an optional sales offer might be printed, based on that customer's sales history.
8. The invoice total is printed at the bottom of the page in the form of a payment coupon.

DDS referenced in the INVPRE printer file formats and positions the output external to the application program, using DDS in the.

DDS Source for INVPRE Example

This is the DDS printer file source for the INVPRE DDS application:

```
5722WDS V5R2M0                      SEU SOURCE LISTING    10/10/02 17:47:39    1
SOURCE FILE . . . . . SAMPLER/QDDSSRC
MEMBER . . . . . INVPRE
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
100    A*  INVPRE: Sample of standard invoicing output used with
200    A*           preprinted forms.
300    A*
400    A*  Printer File Create or Change Parameters:
500    A*  - Printer Device Type (DEVTYPE)      *SCS
600    A*  - Overflow Line (OVRFLW)            64
700    A*  - Front Overlay (FRONTOVL)         INVAL
800    A*
900    A*  Page 1 Header
1000   A*- includes Postnet Zip+4
1100   A*
1200   A          R  INVTOP                      SKIPB(12)
1300   A          NAME                25A        12
1400   A          STNAME               25A        48
1500   A                                     SPACEA(1)
1600   A          STREET                25A        12
1700   A          STSTRT               25A        48
1800   A                                     SPACEA(1)
1900   A          CITY                  25A        12
2000   A          STCITY                25A        48
2100   A                                     SPACEA(1)
2200   A          STATE                 2A         12
2300   A          ZIP                   9S         16 EDTWRD(' - ')
2400   A          STSTE                 2A         48
2500   A          STZIP                 9S         52 EDTWRD(' - ')
2600   A                                     SPACEA(3)
2700   A          CUST#                  6S 0       14 EDTCDE(Z)
```

2800	A	INVC#	6S 0	32	EDTCDE(Z)		
2900	A			49	DATE	EDTCDE(Y)	
3000	A	PAYDAT	6S 0	66	EDTCDE(Y)		
3100	A				SPACEA(2)		
3200	A	SHPVIA	10A	14			
3300	A			34	DATE	EDTCDE(Y)	
3400	A	TERMS	10A	47			
3500	A	SLSMAN	16A	64			
3600	A				SPACEA(4)		
3700	A*						
3800	A*	Detail Lines					
3900	A*						
4000	A	R DETLIN					
4100	A	QTY	4S 0	8	EDTCDE(Z)		
4200	A	UOM	2A	13			
4300	A	ITEM#	8S 0	18			
4400	A	ITMDES	25A	28			
4500	A	SELPRC	6S 2	58	EDTCDE(J)		
4600	A	EXTPRC	7S 2	70	EDTCDE(J)		
4700	A				SPACEA(1)		
4800	A*						
4900	A*	Multipage Message					
5000	A*						
5100	A	R PAGEOF			SKIPB(51)		
5200	A				70	'Continued'	
5300	A	PAGCNT	3S 0	73	EDTCDE(Z)		
5400	A				SKIPB(63)		
5500	A				65	'Page'	
5600	A*						
5700	A*	Invoice Totals					
5800	A*						
5900	A	R INVBOT			SKIPB(51)		
6000	A	TOTDUE	9S 2	67	EDTWRD(' , , \$0. -')		
6100	A				SPACEA(4)		
6200	A	PAYDA@	6S 0	25	EDTCDE(Y)		
6300	A	TOTD@2	9S 2	67	EDTWRD(' , , \$0. -')		
6400	A				SPACEA(2)		
6500	A	NAME@2	25A	12			
6600	A				SPACEA(1)		
6700	A	STRE@2	25A	12			
6800	A				SPACEA(1)		
6900	A	CITY@2	25A	12			
7000	A				SPACEA(1)		
7100	A	STAT@2	2A	12			
7200	A	ZIP@2	9A	16			
7300	A	PAGCNT	3S 0	73	EDTCDE(Z)		
7400	A				SKIPB(63)		
7500	A				65	'Page'	
7600	A*						
7700	A*	Offer Print					
7800	A*	- Font 92 is Courier Italic 12-pitch					
7900	A*						
8000	A	R OFFER			SKIPB(43)		
8100	A*				FONT(92)		
8200	A	OFFR@1	24A	33			
8300	A				SPACEA(1)		
8400	A	OFFR@2	24A	33			
8500	A				SPACEA(1)		
8600	A	OFFR@3	24A	33			
8700	A				SPACEA(1)		
8800	A	OFFR@4	24A	33			
8900	A				SPACEA(1)		
9000	A	OFFR@5	24A	33			
9100	A				SPACEA(1)		
9200	A	OFFR@6	24A	33			
9300	A				SPACEA(1)		

* * * * * E N D O F S O U R C E * * * * *

Choosing the Proper Tool

Moving the Super Sun Seeds invoice to an electronic invoice involves choosing from a number of application enabling tools. That choice is influenced by at least three factors: (1) the application requirements of the new invoice, (2) the characteristics of the current application, and (3) the skills available or required. See Chapter 25, “Deciding on an Output Strategy and Set of Tools,” on page 287 for more help deciding which product is best for you.

First, the new invoicing application might require:

- Changes in overall page layout.
- Static or dynamic layout.
- Single or multiformat pages.
- Bar codes, images, and other document elements.
- Specification of how copies are handled.
- Certain processing or performance characteristics.

Second, certain characteristics of the existing invoicing application, primarily access to source code, is a factor.

Finally, skill requirements or preferences are a factor:

- Availability of programming skills.
- Scope of electronic printing applications, for example, the number of applications and electronic forms.

All of these approaches create an electronic Super Sun Seeds document (Print Format Utility is better suited to a related application, such as creating shipping labels):

Printer file parameters provide you with an easy, automatic way to change to your output. For information about using printer files, see Chapter 13, “Using Printer Files,” on page 121.

DDS (Data Description Specifications) is the standard interface to the output page. DDS keywords provide complete and dynamic control of all page elements. DDS is integrated with the application program, enabling page content to be customized by application data and program logic. For information about using DDS, see Chapter 14, “Using Data Description Specifications,” on page 125.

Infoprint Designer lets you seamlessly download spooled files that you have retained from an application run and display them in the Layout Editor for drag-and-drop placement on your overlay. Infoprint Designer’s Layout Editor also allows variable page-to-page formatting, such as ‘spot carbon’ and ‘multiple-up’ applications. For more advanced output, you still use the Layout Editor, but you approach the output with a logical sequence of what goes where and when. For information about using Infoprint Designer, see Chapter 17, “Using Infoprint Designer,” on page 197.

Advanced Print Utility (APU) is an interactive tool for transforming existing line-oriented application output to electronic pages, independent of the application program. A new document design is created interactively and that design is applied to the output file created by the application. For more information about APU, see Chapter 15, “Using Advanced Print Utility,” on page 153.

Page definitions and form definitions provide application-independent document formatting. These definitions, created with **PPFA**, **Infoprint Designer**, or other document applications, are referenced by the printer file. This dynamically transforms existing line-mode output to complex electronic documents. See Chapter 16, "Using Page Printer Formatting Aid," on page 183 for information about using PPFA to create page definitions and form definitions.

AFP Toolbox provides complete control over the AFP print data stream, using high-function objects and APIs. AFP Toolbox provides the programming interface to create critical, complex output - output with dynamic elements, image integration, indexing, or output data stream transformation. For information about using AFP Toolbox, see Chapter 18, "Using IBM AFP Toolbox," on page 217.

Print Format Utility (PFU) creates electronic output directly from iSeries database files, using an interactive user interface. PFU is well suited to quick turnaround applications and multi-up documents (such as bar coded labels). For information about using PFU, see Chapter 19, "Using the Print Format Utility," on page 223.

Chapter 13. Using Printer Files

As described in Chapter 4, "Introduction to Printer Files," on page 25, every OS/400 job has a printer file that provides page defaults such as margins, line spacing, and overflow. The printer file also provides printer and output queue options such as printer device, file type, and whether the spooled file is saved. It lets you specify page composition objects, such as front and back overlays and page definitions and form definitions.

One of the simplest ways to create fully graphical documents is to add an overlay to an application with a printer file. This technique instantly transforms a preprinted form application into a completely electronic application. Using printer file overlays does not provide you with as many functions as those done with DDS or with page definitions and form definitions, but for many output applications, this method is adequate.

Figure 3 on page 13 shows the Super Sun Seeds sample case study invoice application. The existing application program, whether data fields are program-described or externally-described, only places lines of output in the spooled file. The document is completed by printing those lines on a fixed, preprinted form.

Specifying an overlay in the printer file eliminates the need for pre-printed forms. To add an overlay, modify the printer file overlay and positioning parameters with the Create Printer File (CRTPRTF), Change Printer File (CHGPRTF), or Override Printer File (OVRPRTF) commands. These are the key parameters to use when adding an overlay, shown in bold on the Create Printer File screens below:

```

Create Printer File (CRTPRTF)

Type choices, press Enter.

AFP Characters . . . . . *NONE          Character value, *NONE
      + for more values
Degree of page rotation . . . . *AUTO          *AUTO, *DEVD, *COR, 0, 90...
Pages per side . . . . . 1             1-4
Reduce output . . . . . *TEXT          *TEXT, *NONE
Print text . . . . . *JOB
Hardware justification . . . . 0           0, 50, 100
Print on both sides . . . . . *NO        *NO, *YES, *TUMBLE, *FORMDF
Unit of measure . . . . . *INCH        *INCH, *CM
Front side overlay:
  Overlay . . . . . *NONE              Name, *NONE
  Library . . . . .                  Name, *LIBL, *CURLIB
  Offset down . . . . .                0-57.790
  Offset across . . . . .              0-57.790

More...
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

```

Create Printer File (CRTPRTF)

Type choices, press Enter.

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

More...
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

Figure 50. Create Printer File - Adding an Overlay

- Printer device type**
Since overlays are AFPDS objects, Printer device type must specify *AFPDS.
- Unit of Measure**
Specify whether measurements are given in inches or centimeters.
- Front side overlay**
Specify an overlay to put on the front side of every page. You can specify its offset from the page origin.
- Back side overlay**
Specify an overlay to put on the back side of every page. You can specify its offset from the page origin. You can also specify whether the Constant back option is used. For a duplex job, this lets you print an overlay on the back of each page in the spooled file without manually adding blank pages for the back overlay to be printed on.

Front side margin

Specify the margins to use for the front of every page. You can specify the down and across values or use the values in the device description.

Back side margin

Specify the margins to use for the back of every page. You can specify the down and across values or use the values in the device description.

Usually, you specify the overlay object name for the front overlay parameter and use zero for the offsets. If the new invoice overlay and the invoice data do not line up, you can use the overlay offsets and margins to fine-tune the positioning. Note that in this case, the margin parameters move all page elements – the application data and the overlay. Therefore, it is usually best to use the overlay offset to correctly position the electronic form with respect to the data. Any offsets you use, overlay or margin, work in the units (inches or centimeters) defined by the Unit of measure parameter.

Additional Considerations

Using printer file overlay support to transform the Super Sun Seeds sample application is easy and quick, and it requires no change to the application program. In fact, you can make this change after the application has already run and created the spooled file. You use the Change Spooled File Attributes (CHGSPLFA) command and the parameters described above. The limitation of this approach is that you have cannot address more complex documents or reformat the application data.

Duplexing and multiple copies are often built into preprinted forms. The continuous form has multiple copies and might have some fixed text on the back of one or all of the copies. With printer file overlay support, you can create multiple copies by printing the spooled file multiple times and overriding it each time with a different overlay name; that is, by specifying invoice overlays that differ only in that they say “Customer Copy”, “Packing List”, and so forth at the bottom of the form.

This process can be automated with a CL program. Using this technique, you would produce uncollated copies of the invoices, which means that all the customer copies are in one stack, all the packing lists are in another, and so on. If one or more of the copies needs to be grouped together for your invoicing function; that is, collated, then one of the other application enabling techniques covered in the following chapters is more appropriate.

Placing constant text on the back side of the invoice can be done with back overlay support, but requires an application change. Constant text usually consists of terms and conditions information. If you have specified DUPLEX(*YES) and you specified both front and back overlays, then the invoice data will print on both sides. This means that the back side prints with the right overlay and with unwanted application data. The cause of the problem is that there is no page eject to move that page of invoice data to the front of the next page. A simple change to the application program can add a page eject after each page of invoice data is printed.

There are some limitations to this approach, specifically as they relate to a preprinted invoicing application. You cannot specify these with a printer file:

- Multiple overlays, either per page or over groups of pages
- Images, graphics, or bar codes
- Integrated application logic

The above functions can only be specified with the more advanced printing techniques covered in subsequent chapters. These elements all require you to take some action based on the content of the program data. For example, you might want to use a different overlay based on the company number in the invoice data, print an output field in bar code, or print an image based on customer information. These functions are beyond the scope of the overlay support in the printer file.

Chapter 14. Using Data Description Specifications

Data Description Specifications (DDS) is included in OS/400. Using it for application output gives you access to the advanced printing capabilities of OS/400 in much the same way as using external database files gives access to OS/400's advanced database capabilities. DDS printer file support provides full control of each page and all the elements that come together on a page, such as overlays, images, bar codes, lines, and boxes. For example, with DDS you can control what overlays or images go on what pages. DDS also supports the advanced printing capabilities of today's printers, such as finishing.

Nearly all DDS support is dynamic; that is, the application controls both the element and its positioning at run time.

As described in Chapter 4, "Introduction to Printer Files," on page 25, every OS/400 job has a printer file that provides page defaults such as margins, line spacing, and overflow. The printer file also provides printer and output queue options such as printer device, file type, and whether the spooled file is saved. It lets you specify page composition objects, such as front and back overlays and page definitions and form definitions. Generally, these parameters apply to the entire job or spooled file. DDS works with the printer file to format and define application output.

Printer File DDS

DDS printer file support externalizes application output formatting and extends it to full-page applications. Output that is defined in the program is called program-described output. Output that is defined with DDS is called externally-described output.

Figure 51 on page 126 shows a simple example of how DDS is used in the printer file to define application output. The application program prepares the variable data. It also establishes the logic of printing, or in this case, the "writes" to DDS records. DDS groups one or more individual fields together to create a record. A DDS print record is the collection of fields and print keywords that are to be run when the application program issues the write command. The application program controls output by specifying when to write which records.

The data fields are referenced in the record format. The printer file specifies the DDS to use on the Source file (SRCFILE) parameter. Through DDS keywords, the printer file controls the position, orientation, font, and other characteristics for those fields. In addition, DDS provides access to all the elements—text, overlays, images, graphics, bar codes, lines, and boxes—that comprise AFP documents.

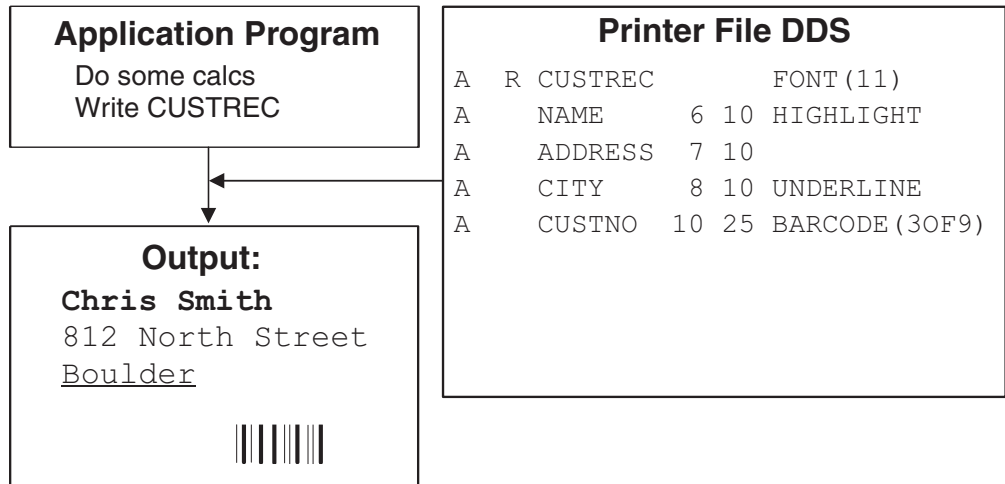


Figure 51. Externally Described Printer File with DDS

Keywords for AFP Applications

The keywords described in this section are used with applications that use full page mode. Keywords for page layout, page composition, and finishing are listed. For a description of all DDS keywords used in printer files, refer to *DDS Reference: Printer Files* in the iSeries Information Center.

Table 14. DDS Keywords for AFP Applications

Page layout keywords: record-level keywords unless otherwise specified	
Keyword	Description
DOCIDXTAG (Document Index Tag)	Creates indexing tags in the document for use by presentation systems such as the AFP Viewer and OnDemand. You specify Attribute-name (such as 'Policy Number'), Attribute -value (such as '43127'), and Tag Level (Group or Page).
DRAWER (Drawer)	Specifies the drawer from which noncontinuous forms are selected.
DTASTMCMD (Data Stream Command)	Use this record- or field-level keyword to store a data stream command or some other piece of information in a spooled file. You can use this command to determine how to process a record or field on a particular page of the spooled file.
DUPLEX (Duplex)	Specifies whether output is printed on one or on two sides of the paper.
ENDPAGE (End Page)	Ejects the current page after the current output record is printed.
ENDPAGGRP (End Page Group)	Ends a logical group of pages. The Infoprint Server PDF subsystem can break up a file between STRPAGGRP and ENDPAGGRP keywords and create a PDF file from each group.
FORCE (Force)	Forces a new sheet of paper for duplex printing. If currently printing on the front side of the sheet, the current sheet is ejected, and a new sheet is fed in.
INVMMAP (Invoke Medium Map)	Specifies the name of a medium map in a form definition. A medium map lets you select and change print parameters such as drawer, page rotation, overlays to use, finishing, and offset.

Table 14. DDS Keywords for AFP Applications (continued)

LPI (Lines Per Inch)	Changes lines per inch. When the LPI keyword is specified, it overrides the LPI parameter of the printer file.
OUTBIN (Output Bin)	Specifies the destination bin of the output on printers that support multiple output bins.
PAGRTT (Page Rotation)	Specifies the degree of rotation of the constants and text that is printed. It overrides the PAGRTT parameter value in the printer file.
STRPAGGRP (Start Page Group)	Begins a logical group of pages. The Infoprint Server PDF subsystem can break up a file between STRPAGGRP and ENDPAGGRP keywords and create a PDF file from each group. You can also include an e-mail address, mail tag, or index tag with this parameter for use with the PDF subsystem.
Page composition keywords: Most of these keywords enable program-to-system fields that are used to dynamically change the values of the keyword parameters while the application program is processing. These are record-level keywords unless otherwise specified.	
Keyword	Description
BARCODE (Bar Code)	A field-level keyword that prints a field in a user-specified bar code in the specified height. DDS supports both linear and 2D bar codes.
BOX (Box)	Prints a box with the specified shading and color. You specify the box by specifying the down and across positions of two opposite corners, and then giving the line width. The corner co-ordinates can be specified dynamically.
CDEFNT (Coded Font Name)	A record- or field-level keyword that specifies the coded font for printing constants or fields. A coded font specifies an OS/400-resident font by pointing to a specific font character set and code page.
CHRSIZ (Character Size)	Changes the width and height of a constant or field.
COLOR (Color)	Specifies the color for a field. Color is used by printers that support color, such as the 4224 printer, and by the AFP Viewer.
FNTCHRSET (Font Character Set)	A record- or field-level keyword that specifies the font character set used for printing constants or fields. You can specify point size for outline fonts.
FONT (Font)	A record- or field-level keyword that specifies the font ID used for printing constants or fields. You can specify the height and width for printer-resident outline fonts. This scaling can be uniform (by specifying only the height) but does not have to be.
GDF (Graphic Data File)	Prints a graphic data file created by a program such as BGU or GDDM.
HIGHLIGHT (Highlight)	Prints a field in bold characters. The active font for the field must be a numeric font (FONT keyword) that supports bold printing.
LINE (Line)	Prints a horizontal or vertical line. Lines are drawn by specifying the down and across starting positions, the horizontal and vertical direction, and the line width and length. Line coordinates can be specified dynamically.
OVERLAY (Overlay)	Prints an overlay at the specified location, with the specified rotation.
PAGSEG (Page Segment)	Prints a page segment at the specified location on the page, with the specified size and rotation.

Table 14. DDS Keywords for AFP Applications (continued)

POSITION (Position)	A field-level keyword used to specify the exact position of a field.
TXTRTT (Text Rotation)	A field-level keyword used to rotate fields.
UNDERLINE (Underline)	Underlines a field.
Finishing operation keywords:	
Keyword	Description
ZFOLD (Z-fold)	A record-level keyword that causes the current sheet to be first folded in half inwards (so the front side of the sheet is now inside the fold) along a line parallel to the reference edge. The half of the sheet furthest from the reference edge is again folded in half outwards along a line parallel to the reference edge. For example, when applied to an 11 by 17 inch sheet with the reference edge along a short side, the result is an 8.5 by 11inch fold-out.
STAPLE (Staple)	Turns on and off the stapling that is specified in the printer file. Use this keyword to staple groups of pages. The staple commands must be issued on a page boundary. There are three ways to use the STAPLE keyword. To staple consecutive groups of pages, such as pages 1-4 and 5-9, use STAPLE with no parameters on pages 5 and 10. To staple non-consecutive groups of pages, such as pages 1-4 and 7-10, use STAPLE(*OFF) on page 5 and STAPLE(*ON) on page 7 then STAPLE on page 11.

DDS Functions

Now that the DDS keywords have been introduced, we will show how these keywords are used in a sample case. Figure 52 on page 129 shows how DDS can be used to produce complex output.

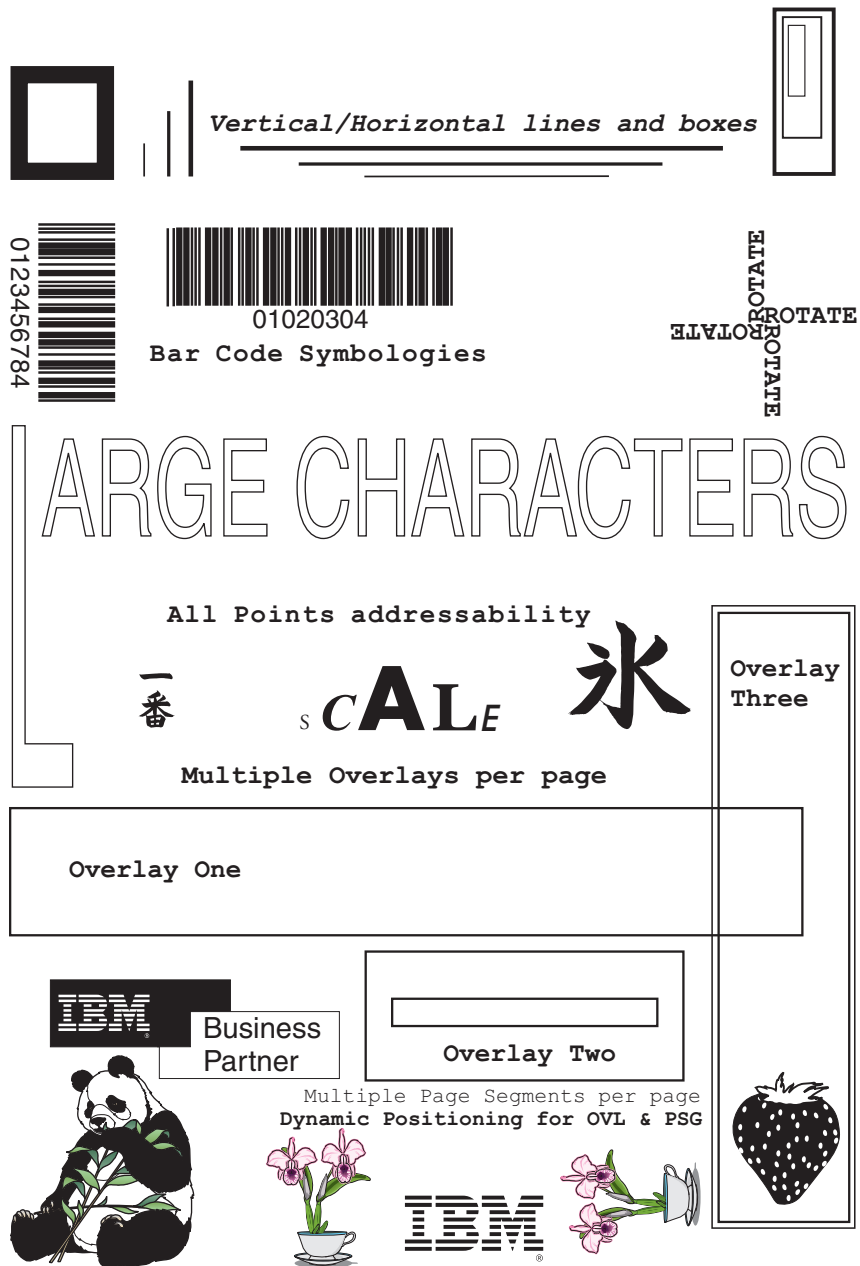


Figure 52. DDS Functionality

DDS Example

These are the DDS printer specifications used to produce the multiple function output example shown in Figure 52:

```

5722WDS V5R2M0                               SEU SOURCE LISTING
SOURCE FILE . . . . . SAMPLER/QDDSSRC
MEMBER . . . . . DDSFUN
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ..
100      A* DDSFUN: DDS for Functional DDS Demo
200      A*
300      A*      Printer File Parameters when compiling:
400      A*
500      A*      CHGPRTF FILE(DDSFUN) DEVICE(XXXXXXXX) TYPE(*AFPDS)

```

```

600      A*          OVRFLW(64)
700      A*
800      A          R HEADR1
900      A          PAGRTT(0)
1000     A*
1100     A* 'DDS Functionality'
1200     A* - Helvetica Bold 20-pt using outline font
1300     A* - Down 0.7 Across 3.0
1400     A*
1500     A          LIN01          35A
1600     A          FNTCHRSET(CZH400, POINTSIZE 20)
1700     A          POSITION(0.7 3.0)
1800     A*
1900     A* 'OS/400 V3R1 and later releases'
2000     A* - Helvetica Reg 12-pt using outline font
2100     A* - Var Positioning
2200     A*
2300     A          LIN02          35A
2400     A          FNTCHRSET(T1V10037, POINTSIZE 12)
2500     A          POSITION(&LINDWN &LINACR)
2600     A          LINDWN          5S 3P
2700     A          LINACR          5S 3P
2800     A*
2900     A* LINE Parameters:
3000     A* - Start Position Down
3100     A* - Start Position Across
3200     A* - Length (use UOM in printer file)
3300     A* - *HRZ or *VRT
3400     A* - Line Width
3500     A*
3600     A          R LINE1
3700     A          LINE(1.3 2.6 0.2 *VRT *NARROW
3800     A          LINE(1.1 2.8 0.4 *VRT *MEDIUM
3900     A          LINE(0.9 3.0 0.6 *VRT *WIDE)
4000     A          LINE(1.4 3.3 3.2 *HRZ *WIDE)
4100     A          LINE(1.5 3.7 2.4 *HRZ *MEDIUM
4200     A          LINE(1.6 4.1 1.6 *HRZ *NARROW
4300     A*
4400     A* BOX Parameters
4500     A* - Start Position Down
4600     A* - Start Position Across
4700     A* - Diag Position Down
4800     A* - Diag Position Across
4900     A* - Line width
5000     A*
5100     A          R BOX1
5200     A          BOX(0.8 1.0 1.5 2.0 .1)
5300     A          BOX(0.4 7.0 2.1 7.2 .02)
5400     A          R TXT0
5500     A          LIN08          35A 36 27
5600     A          LIN09          35A 50 31
5700     A          51 33 'Dynamic Positioning for OVL
5800     A          FONT(85)
5900     A* LIN03 'Vertical/Horizontal lines and boxes'
6000     A* - Font 18 = Courier Italic
6100     A* LIN04 'Bar Code Symbologies'
6200     A* LIN05 'L'
6300     A* - Font ADMWMOB = Open Block (GDDM)
6400     A* LIN06 'Large Characters'
6500     A* - Font ADMWMOB = Open Block (GDDM)
6600     A* LIN07 'All Points Addressability'
6700     A* - Font 46 = Courier Bold
6800     A*
6900     A          R TXT1
7000     A          LIN03          35A
7100     A          POSITION(1.3 3.3)
7200     A          FONT(18)

```



```

7300      A          LIN04          35A
7400      A
7500      A          LIN05          1A
7600      A
7700      A
7800      A
7900      A          LIN06          15A
8000      A
8100      A
8200      A
8300      A          LIN07          35A
8400      A
8500      A
8600      A*  Print 'ROTATE' in 4 orientations
8700      A*
8800      A          R TXT3
8900      A          TXT1@1          6
9000      A
9100      A          TXT1@2          6
9200      A
9300      A          TXT1@3          6
9400      A
9500      A          TXT1@4          6
9600      A
9700      A*
9800      A*  Bar Code Parameters:
9900      A*  BARCODE (Type, Height, + 6 parameters:
10000     A* - Horizontal/Vertical: *HRZ,*VRT
10100     A* - Human Readable: *HRI,*HRITOP,*NOHRI
10200     A* - Asterisk *AST, *NOAST
10300     A* - Check Digit: type
10400     A* - Unit Width: narrow bar/space
10500     A* - Wide/Narrow Ratio
10600     A*
10700     A          R BAR1
10800     A          BAR1@1          8S
10900     A
11000     A          BAR2@1          8
11100     A
11200     A*
11300     A*  Print 'POSITION' with different offsets
11400     A*
11500     A          R TXT2
11600     A          TXT2@1          8
11700     A          TXT2@2          8
11800     A          TXT2@3          8
11900     A          TXT2@4          8
12000     A          TXT2@5          8
12100     A*
12200     A*  Page Segment IBMLOGOT
12300     A* - dynamic name
12400     A* - dynamic position
12500     A*
12600     A          R PSG1
12700     A
12800     A          PSGNAM          8A P
12900     A          PSGDWN          5S 3P
13000     A          PSGACR          5S 3P
13100     A*  Overlays
13200     A* - dynamic name
13300     A* - dynamic position
13400     A*
13500     A          R OVL1
13600     A
13700     A
13800     A
13900     A

```

```

14000      A                                OVERLAY(DDSOVL3 4.5 6.5)
14100      A          OVLNM2              8A P
14200      A          OV3DWN              5S 3P
14300      A          OV3ACR              5S 3P
          * * * * E N D   O F   S O U R C E * * * *

```

The most significant print elements in the DDSFUN printer DDS specifications are:

Text Text, whether constant or field data, is printed in a variety of ways in DDSFUN.

“DDS Functionality”(LIN01)

Printed in a 20-point Helvetica Roman-Bold font, 0.7 inches down and 3.0 inches across the page. The FRONTMGN parameter of the printer file is set to 0, so the down and across positions are measured from the top left edge of the page. Note that the POSITION keyword specifies the baseline or bottom left point of the first character to print.

The font is specified using FNTCHRSET, which defines the character set and code page to use. In the outline font CZH400, “CZ” is the prefix for outline fonts, “H” is for Helvetica, and “400” defines Roman Bold. The point size is specified separately with the POINTSIZE parameter.

This is a typographic font, part of the AFP Font Collection. For 300-pel printers, C0H400J0 is found in library QFNT300LA1. Code page T1V10037 is the USA and Canada code page, and it is normally located in library QFNTCPL.

“OS/400 V3R1 and Later releases”(LIN02)

Prints in Helvetica Roman-Medium 12-point, 0.9 inches down and 3.3 inches across the page. The FNTCHRSET value is CZH200. “CZ” is the outline font prefix, “H” is for Helvetica, and “200” is for Roman Medium. Dynamic positioning is used, with program variables LINDWN and LINACR loaded with the down and across values and referenced in the DDS as program-to-system fields.

“Vertical/Horizontal lines and boxes”(LIN03)

Prints in Courier Italic printer-resident font, starting 1.3 inches down and 3.3 inches across the page. The keyword FONT(18) specifies Courier Italic.

“Bar Code Symbolologies”(LIN04)

Prints in Courier 10, which is the default font. In this case it is specified as font identifier 011 in the printer device description.

“L”(LIN05)

L is printed in the Open Block font scaled by the CHRSIZ keyword to 9.0 width and 20.0 height. ADMWMOB is the Open Block font and it is a GDDM scalable font located in the QGDDM library.

“Large Characters”(LIN06)

Also prints in Open Block, but is scaled to 6.0 wide and 6.0 high.

“All points addressability”(LIN07)

Prints in Courier Bold, which is FONT(46).

“Multiple Overlays per page”(LIN08)

Prints in the default font, Courier - FONT(011).

“Multiple Page Segments per page”(LIN09)

Prints in the default font, Courier - FONT(011).

“Dynamic Positioning for OVL and PSG”

Prints in font 85, Prestige Elite.

“Rotate”

Prints the fields TXT1@1 through TXT@4 in the four different rotations: 0, 90, 180, and 270. Each field from TXT@1 through TXT@4 contains the character “ROTATE”. TXTRTT is valid only for fields and not for constants. Note how the POSITION (2.7 inches down and 6.4 inches across the page) defines a baseline starting point for each rotation.

Lines Three vertical and three horizontal lines are printed. The first vertical line begins at a point 1.3 inches down and 2.6 inches across the page, and has a length of 0.2 inches. The line width is *NARROW, 0.008 inches.

All five parameters of the LINE keyword can be program-to-system variables, enabling the application to dynamically draw lines.

Boxes Two boxes are drawn in the DDSFUN example. The first, thicker box is defined by top left (0.8 inch down, 1.0 inch across the page) and bottom right (1.5 inches down, 2.0 inches across the page) positions. The box width is 0.1 inch. Box width also can be specified by the *NARROW, *MEDIUM, and *WIDE special values.

All five parameters of the BOX keyword can be program-to-system variables, enabling the application to dynamically draw boxes.

Page Segments

The page segments explicitly placed on the DDSFUN page are IBM logos. The largest IBM logo is a page segment object (*PAGSEG) called IBMLOGOT, and is positioned 8.7 inches down and 3.6 inches across the page. Unlike text, this position marks the top left point of the page segment image (top left when printed in standard, or 0 rotation).

Note that the strawberry image, a page segment called “STRWNB”, is not explicitly placed by DDS. It is included in Overlay Three.

The three IBM logos are placed using program-to-system variables, enabling the application to dynamically position images.

Overlays

Three very simple overlays are depicted in the DDSFUN example. While the overlays used with the Super Sun Seeds case study are more familiar and realistic, these overlays demonstrate how overlays are used by DDS.

Overlay One is an OS/400 overlay object (*OVL) called DDSOVL1. It is placed 6.0 inches down and 1.3 inches across the page. Again, this is relative to the page margins and marks the top left point of the overlay.

Overlay Two is dynamically referenced from the program by the variable OVLNM2.

Overlay Three is dynamically positioned from the program by the variables OV3DWN and OV3ACR for down and across, respectively.

Bar Codes

Two examples of bar codes are specified.

The field BAR1@1 is printed vertically in the Interleaved 2 of 5 bar code symbology, starting at 2.0 inches down and 1.8 inches across the page. The bar code is printed with a height of 3, which at 6 LPI prints a 1/2-inch

high bar code. Interleaved 2 of 5 is a numeric-only bar code. The human readable field value (012345678) is printed below the bar code, along with the check digit (4).

The field BAR2@1 is printed horizontally in the Code 3 of 9 bar code symbology, starting at 2.0 inches down and 2.5 inches across the page. It prints horizontally because *HRZ is the default. The human readable (01020304) field value is also the default. Note that Code 3 of 9 is an alphanumeric bar code (up to 50 characters), and does not include a check digit.

RPG for Example

This is the RPG programming specifications used to produce the multiple function output example shown in Figure 52 on page 129:

```

5722WDS V5R2M0                      SEU SOURCE LISTING
SOURCE FILE . . . . . SAMPLER/QRPGSRC
MEMBER . . . . . DDSFUN
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ..
 100      H
 200      H* DDSFUN: DDS Functional Demo
 300      H*
 400      H* Uses the printer file DDSFUN.
 500      H*
 600      FDDSFUN 0 E                      PRINTER
 700      E                                DTA      1 10 8          Constant Data
 800      E                                LIN       1  9 35          Title Lines
 900      *
1000      IFIELDS      DS
1100      I                                1 315 LIN
1200      I                                1  35 LIN01
1300      I                                36  70 LIN02
1400      I                                71 105 LIN03
1500      I                                106 140 LIN04
1600      I                                141 141 LIN05
1700      I                                176 190 LIN06
1800      I                                211 245 LIN07
1900      I                                246 280 LIN08
2000      I                                281 315 LIN09
2100      *
2200      C                                MOVE DTA,1    BAR1@1  80      12345678
2300      C                                MOVE DTA,2    BAR2@1   8      01020304
2400      C*
2500      C                                MOVE 'ROTATE' TXT1@1  6      ROTATE ex.
2600      C                                MOVE TXT1@1  TXT1@2  6
2700      C                                MOVE TXT1@1  TXT1@3  6
2800      C                                MOVE TXT1@1  TXT1@4  6
2900      C*
3000      C                                MOVE DTA,3    TXT2@1  8      POSITION ex.
3100      C                                MOVE DTA,3    TXT2@2  8
3200      C                                MOVE DTA,3    TXT2@3  8
3300      C                                MOVE DTA,3    TXT2@4  8
3400      *
3500      C                                Z-ADD0.9    LINDWN  53      Var Positionin
3600      C                                Z-ADD3.3    LINACR  53
3700      C                                WRITEHEADR1
3800      C                                WRITELINE1
3900      C                                WRITEBOX1
4000      *
4100      C                                WRITETXT0
4200      C                                WRITETXT1
4300      C                                WRITEBAR1
4400      C                                WRITETXT2
4500      C                                WRITETXT3
4600      *

```

```

4700      C                MOVE 'IBMLOGOT'PSGNAM 8          Var Image (Page
4800      C                Z-ADD8.7      PSGDWN 53          Var Positioning
4900      C                Z-ADD3.6      PSGACR 53
5000      C                WRITEPSG1
5100      *
5200      C                MOVE 'IBMLOGO 'PSGNAM
5300      C                Z-ADD8.8      PSGDWN 53
5400      C                Z-ADD5.3      PSGACR 53
5500      C                WRITEPSG1
5600      *
5700      C                MOVE 'IBSMALL 'PSGNAM
5800      C                Z-ADD9.35     PSGDWN 53
5900      C                Z-ADD6.8      PSGACR 53
6000      C                WRITEPSG1
6100      *
6200      C                MOVE 'DDSOVL2 'OVLNM2 8          Var Overlay
6300      C                Z-ADD4.4      OV3DWN 53          Var Positioning
6400      C                Z-ADD6.3      OV3ACR 53
6500      C                WRITEOVL1
6600      *
6700      C                SETON                      LR
6800 **   BAR   Bar code data
6900 12345678
7000 01020304
7100 POSITION
7200 **   LIN   LINES DATA
7300 DDS Functionality
7400 OS/400 V3R1 and later releases
7500 Vertical/Horizontal lines and boxes
7600 Bar Code Symbolologies
7700 L
7800 ARGE CHARACTERS
7900 All Points addressability
8000 Multiple Overlays per page
8100 Multiple Page Segments per page
          * * * * E N D O F S O U R C E * * * *

```

Floating Document Elements

Because DDS can variably place document elements such as lines, boxes, page segments, and overlays, you can precisely tailor the output to the application data. You can float these elements on the page; that is, actually change the document structure as the data changes.

For example, consider a brokerage statement. It has sections for portfolio transactions, checking transactions, portfolio balances, and other categories. The transactions are all placed in separate boxes or frames on the statement. Those frames end when the customer transactions end. The result is a document tailored for that customer.

As shown in “Using DDS for Super Sun Seeds” on page 137, you can use DDS to accomplish the same thing. While the invoice was tailored depending on whether it was a one, two, or more page document, it still has a fixed frame for the detailed line items. A customer invoice with five line items has a frame that is part of the 35 line item overlay. If DDS line keywords are used, the frame can be built dynamically; that is, as line items are processed. If five line items are all the customer needs, that is all that is drawn.

Program INVNEW3 implements the Super Sun Seeds invoice as a floating document. A sample of the output is shown below.




400 CPU Parkway Vegetation, NJ 55090				Office: 555-499-2367 Fax: 555-415-9794	
IMPROVED PRINTING CORP PERFORMANCE BOULEVARD PRINTERSVILLE CO 45789-2637		SAME			
-- Sold To --				-- Ship To --	
Customer Number:	100	Invoice Number:	31300	Invoice Date:	3/12/02
				Payment Date:	4/12/02
Ship Via: BEST WAY Ship Date: 3/12/02 Terms: NET 30 Rep: YOUR PRINTER REP					
Qty	UOM	Item #	Item Description	Price	Extension
1	CT	01100517	HIGH ALTITUDE WATERMELON	1.01	1.01
1	PK	04569870	SPARTAN SEEDS	2.39	2.39
9	PK	11005004	NORTHERN LITE BLUE SPRUCE	858.32	7,724.88
12	BX	11005011	BUSH GREEN SEEDS	2.50	30.00
12	CT	11005018	LASSO RED SEEDS	892.23	10,706.76
26	PK	11057893	EARLY BANTAM SEEDS	.38	9.88
5	BX	15975365	AFRICAN DAISY, SEEDS	2.35	11.75
1	PK	32746510	HEAVY OAK	129.09	129.09
33	BX	46578913	HOPS BREWING LIGHT	1.20	39.60
6	EA	56412113	SEED SURVEYING SITE	50.00	300.00
2	BX	65412384	POT POT	7.65	15.30
80	PK	84512023	SEED SCRUBBER	888.79	71,103.20
1	PK	96325874	OREGON SPRING TOMATO SEED	.97	.97
2	DZ	98412006	PINEAPPLE-ORANGE SEED	1.29	2.58
11	BX	98546320	BLACK BEAUTY ZUCCHINI	2.30	25.30
5	EA	00000300	FROZEN JUICE PROCESSOR	109.90	549.50
				Thank You Because you have ordered over \$500 of seeds this year, on your next seed order you will receive a 10% discount.	
This invoice overlay designed using IBM AFP Utilities/400				Total Due	\$90,652.21
Return this tear-off strip with your payment.		Make Checks Payable to:		Super Sun Seeds	
Payment is due by: 4/12/02		Amount Due is:		\$90,652.21	
IMPROVED PRINTING CORP PERFORMANCE BOULEVARD PRINTERSVILLE CO 457892637					

Figure 55. New One Page DDS Version of Super Sun Seeds Invoice

The fictional customer, Organic Garden Supplies, on the other hand, has ordered 46 items so their invoice is two pages long. The first page of their invoice is shown in Figure 56 on page 139.

400 CPU Parkway Vegetation, NJ 55090				Office: 555-499-2367 Fax: 555-415-9794	
ORGANIC GARDEN SUPPLIES 546 PRODUCE WAY GOLDENOATS CO 94523-4852 -- Sold To --			ORGANICS-ON-THE-MOVE 3872 NATURE'S WAY NOCHEMS AK 49972-5341 -- Ship To --		
Customer Number:	136	Invoice Number:	31336	Invoice Date:	3/13/02
		Payment Date:		4/13/02	
Ship Via:CLEAN TRK Shipped Date: 3/13/02 Terms:NET 30 Rep: CHRIS SEEDER					
Qty	UOM	Item #	Item Description	Price	Extension
90	CT	00000300	HIGH ALTITUDE WATERMELON	1.01	90.99
550	CT	00000300	HIGH ALTITUDE WATERMELON	1.01	555.50
100	EA	00001200	ARBOLES DEL SUR	45.00	4,500.00
25	EA	00231300	SEED ROASTER OVEN SET	199.99	4,999.75
150	PK	04569870	NORTHERN LITE BLUE SPRUCE	858.32	28,748.00
2	BX	11005000	FAVA SEEDS	3.90	7.80
2	BX	11005001	PURPLE TEEPEE SEEDS	4.44	8.88
52	BX	11005002	BUSH WAX SEEDS	2.00	104.00
52	BX	11005003	KINGHORN WAX SEEDS	2.13	110.76
8	BX	11005004	BUSH GREEN SEEDS	2.50	20.00
8	BX	11005005	BLUE LAKE GREEN SEEDS	4.00	32.00
2	BX	11005006	KINGHORN WAX SEEDS	3.00	6.00
2	CT	11005007	VENTURE GREEN SEEDS	1.50	3.00
100	CT	11005008	NORTHEASTERN POLE SEEDS	1.29	129.00
100	CT	11005009	KENTUCKY BLUE SEEDS	2.10	210.00
58	CT	11005010	EARLY DWARF DANISH SEEDS	3.01	174.58
58	CT	11005011	LASSO RED SEEDS	892.23	51,749.34
84	EA	11005012	BLUE MAX SAVOY BEANS	1.23	103.32
84	DZ	11005013	MINCOR NANTES CARROT SEEDS	.87	73.08
10	DZ	11005014	SCARLET NANTES SEEDS	5.90	59.00
5	DZ	11005014	SCARLET NANTES SEEDS	5.90	29.50
10	BZ	11005015	CHANTENAY SEEDS	2.19	21.90
63	BZ	11005016	TOUCHON SEEDS	2.83	178.29
65	BZ	11005016	TOUCHON SEEDS	2.83	183.95
2	PK	11005018	EARLY BANTAM SEEDS	.38	.76
2	PK	11005019	NORTHERN PICKLING SEEDS	.39	.78
90	PK	11005020	FRENCH PICKLING SEEDS	2.39	215.10
100	BX	11057893	AFRICAN DAISY, SEEDS	2.35	235.00
25	CT	12382910	SUCCOTASH SEEDS	.38	9.50
45	CT	13145340	SOUR GRAPE SEEDS	.15	6.75
10	PT	15789342	BLUE BELLS, BRIGHT BLUE	18.57	185.70
50	PK	15975365	HEAVY OAK	129.09	6,454.50
25	EA	31321654	BELLSTAR SEEDS	7.88	197.00
2	EA	31321654	BELLSTAR SEEDS	7.88	15.76
25	DZ	32154657	PETERSBURG PALM TREE	34.90	872.50
6	BZ	32165478	BLACK EYED BANANA	3.01	18.06
45	BX	32746510	HOPS BREWING LIGHT	1.20	54.00
10	CT	35456031	SUNNY SUNFLOWER SEEDS	1.23	12.30

This invoice overlay designed using IBM AFP Utilities/400

Page 1

Figure 56. Page One of Two-Page DDS Version of Super Sun Seeds Invoice

The second page of the Organic Garden Supplies invoice consists of a shortened heading, sales offer (with strawberry image), and the payment coupon as shown in Figure 57 on page 140.



400 CPU Parkway Vegetation, NJ 55090				Office: 555-499-2367 Fax: 555-415-9794	
ORGANIC GARDEN SUPPLIES					
Customer Number:	136	Invoice Number:	31336	Invoice Date:	3/13/02
				Payment Date:	4/13/02
Qty	UOM	Item #	Item Description	Price	Extension
50	EA	35715924	SEED SIFTER SET	2,900.00	45,000.00
18	EA	40113254	FRESH FRUIT CANNED CANNER	22.97	413.46
6	BX	5613213	POT POT	7.65	45.90
1000	PK	64132029	PITLESS PEACH SEEDS	.97	970.00
500	EA	90978412	TREE TRIMMER TUBING	.20	100.00
6	CT	94875081	EARLIROUGE TOMATO SEEDS	.49	2.94
45	BX	98412006	BLACK BEAUTY ZUCCHINI	2.30	103.50
5	EA	98546320	FROZEN JUICE PROCESSOR	109.90	549.50
					
			<i>Thank You Because you have ordered over \$500 of fruit this year, on your next fruit order you will receive a 10% discount.</i>		
This invoice overlay designed using IBM AFP Utilities/400				Total Due	\$147,561.56
Return this tear-off strip with your payment.		Make Checks Payable to:		Super Sun Seeds	
Payment is due by: 4/13/02		Amount Due is:		\$147,561.56	
ORGANIC GARDEN SUPPLIES 546 PRODUCE WAY GOLDENOATS CO 94523-4852					

Figure 57. Page Two of Two-Page DDS Version of Super Sun Seeds Invoice

Multiple overlays are used to customize the format for one, two, and more page invoices, as shown in Figure 58 on page 141. This approach can be more customized by using a floating overlay (see “Floating Document Elements” on page 135) so that the electronic form exactly matches the individual customer transactions. Additionally, a postal bar code can be added to the address, as well as a payment coupon with the customer number and invoice amount encoded in bar code, and a custom marketing offer with tailored images and discounts.

We need to construct a different overlay for all of the possible scenarios. For example, a one page invoice can fit the full ship to and bill to heading, the item detail lines, and payment coupon all on one page. The invoice only uses the INVALL overlay. A multiple page invoice would have the full heading with item details on the first page (INVFST), the continuation heading and item details on the middle pages (INVMID), and the continuation heading, item details and payment coupon on the last page (INVLST). The overlays we constructed are shown in Figure 58 on page 141.

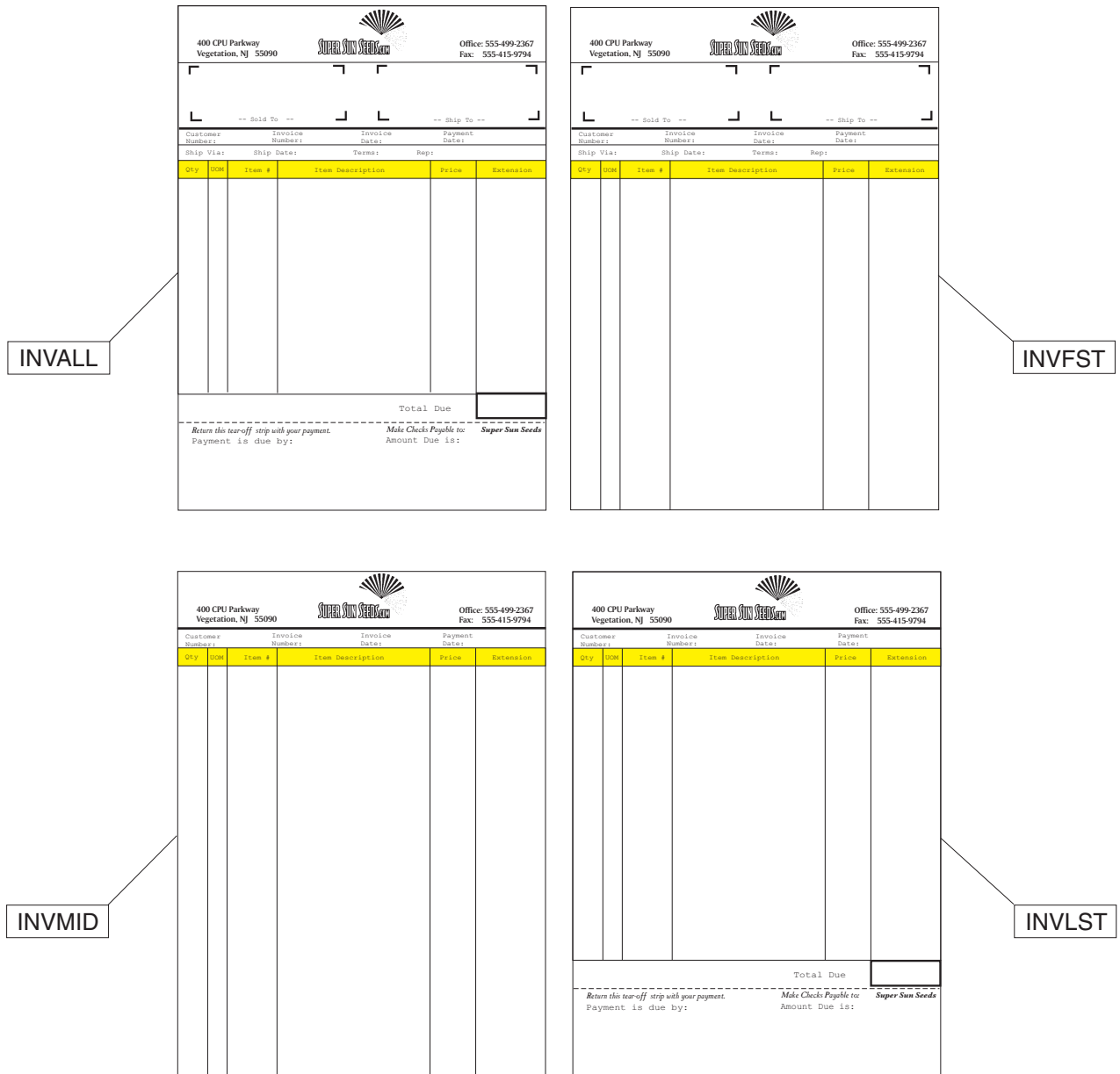


Figure 58. Electronic Overlays on New DDS Super Sun Seeds Invoice

DDS for New Invoice Output

The DDS printer specifications used to produce the overlays shown in Figure 58 are given:

```

5722WDS V5R2M0                               SEU SOURCE LISTING    10/10/02 17:47:39    1
SOURCE FILE . . . . . SAMPLER/QDDSSRC
MEMBER . . . . . INVNEW2
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
100    A*  INVNEW2 - Printer File DDS for Super Sun Seeds Invoice
200    A*          Example 2 (copies and duplex function added)
300    A*
400    A*  Printer File Create or Change Parameters:
500    A*  - Printer Device Type (DEVTYPE)      *AFPDS

```

600	A*	- Overflow Line (OVRFLW)		64
700	A*	- Print on Both Sides (DUPLEX)		*YES
800	A*	if running on duplex printer		
900	A*			
1000	A*	Page 1 Header		
1100	A*-	includes Postnet Zip+4		
1200	A*			
1300	A	R INVTOP		SKIPB(10)
1400	A	ZIPPN	9S	12 BARCODE(POSTNET)
1500	A			SPACEA(2)
1600	A	NAME	25A	12
1700	A	STNAME	25A	48
1800	A			SPACEA(1)
1900	A	STREET	25A	12
2000	A	STSTRT	25A	48
2100	A			SPACEA(1)
2200	A	CITY	25A	12
2300	A	STCITY	25A	48
2400	A			SPACEA(1)
2500	A	STATE	2A	12
2600	A	ZIP	9S	16 EDTWRD(' - ')
2700	A	STSTE	2A	48
2800	A	STZIP	9S	52 EDTWRD(' - ')
2900	A			SPACEA(3)
3000	A	CUST#	6S 0	14 EDTCDE(Z)
3100	A	INVC#	6S 0	32 EDTCDE(Z)
3200	A			49DATE EDTCDE(Y)
3300	A	PAYDAT	6S 0	66EDTCDE(Y)
3400	A			SPACEA(2)
3500	A	SHPVIA	10A	14
3600	A			34DATE EDTCDE(Y)
3700	A	TERMS	10A	47
3800	A	SLSMAN	16A	64
3900	A			SPACEA(4)
4000	A*			
4100	A*	Page 2 Header		
4200	A*			
4300	A	R INVTP2		SKIPB(10)
4400	A	NAME	25A	12
4500	A			SPACEA(2)
4600	A	CUST#	6S 0	14 EDTCDE(Z)
4700	A	INVC#	6S 0	32 EDTCDE(Z)
4800	A			49DATE EDTCDE(Y)
4900	A	PAYDAT	6S 0	66EDTCDE(Y)
5000	A			SPACEA(4)
5100	A*			
5200	A*	Detail Lines		
5300	A*			
5400	A	R DETLIN		SPACEA(1)
5500	A	QTY	4S 0	8 EDTCDE(Z)
5600	A	UOM	2A	13
5700	A	ITEM#	8S 0	18
5800	A	ITMDES	25A	28
5900	A N51	SELPRC	6S 2	58 EDTCDE(J)
6000	A N51	EXTPRC	7S 2	70 EDTCDE(J)
6100	A*			
6200	A*	Multipage Message		
6300	A*			
6400	A	R PAGEOF		
6500	A	PAGCON	4A	POSITION(10.7 7.3)
6600	A			FNTCHRSET(C0H200A0 T1V10037)
6700	A	PAGCNT	2S 0	POSITION(10.7 7.8)
6800	A			FNTCHRSET(C0H200A0 T1V10037)
6900	A			EDTCDE(Z)
7000	A	PAGNAM	25A	POSITION(10.7 3.8)
7100	A			FNTCHRSET(C0H400B0 T1V10037)
7200	A*			

```

7300      A* Invoice Totals
7400      A* - includes Interleaf 2 of 5 bar code
7500      A*
7600      A          R INVBOT          SKIPB(51)
7700      A N51      TOTDUE          9S 2      67 EDTWRD(' , , $0. -')
7800      A
7900      A 51
8000      A          SPACEA(4)
8100      A          PAYDA@          6S 0      25 EDTCDE(Y)
8200      A N51      TOTD@2          9S 2      67 EDTWRD(' , , $0. -')
8300      A
8400      A 51
8500      A          SPACEA(2)
8600      A          NAME@2          25A      12
8700      A          SPACEA(1)
8800      A          STRE@2          25A      12
8900      A          BARPRC          15S 0     52BARCODE(INTERL20F5 3)
9000      A          SPACEA(1)
9100      A          CITY@2          25A      12
9200      A          SPACEA(1)
9300      A          STAT@2          2A      12
9400      A          ZIP@2           9A      16
9500      A*
9600      A* Offer Print
9700      A* - Font 92 is Courier Italic 12-pitch
9800      A*
9900      A          R OFFER          SKIPB(43)
10000     A          FONT(92)
10100     A          OFFR@1          24A      36
10200     A          SPACEA(1)
10300     A          OFFR@2          24A      36
10400     A          SPACEA(1)
10500     A          OFFR@3          24A      36
10600     A          SPACEA(1)
10700     A          OFFR@4          24A      36
10800     A          SPACEA(1)
10900     A          OFFR@5          24A      36
11000     A          SPACEA(1)
11100     A          OFFR@6          24A      36
11200     A          SPACEA(1)
11300     A*
11400     A* Images/Page Segments
11500     A* - variable page segment name from program
11600     A*
11700     A          R PAGSEG          PAGSEG(&PSEG 7.0 2.6)
11800     A          PSEG            8A P
11900     A*
12000     A*
12100     A* Images/Page Segments
12200     A* - variable overlay name from program
12300     A*
12400     A          R PRTOVL          OVERLAY(&OVLAY 0 0)
12500     A          OVLAY            8A P
12600     A*
12700     A* Forces page advance
12800     A*
12900     A          R ENDPAG          ENDPAGE

```

* * * * E N D O F S O U R C E * * * *

The printer file DDS used to create the invoice output shown in Figure 58 on page 141 is divided into seven records as follows:

INVTOP	Full invoice heading information
INVT P2	Continuation heading, on multipage invoice
DETLIN	Detail lines format
INVBOT	Invoice bottom, or payment coupon
OFFER	Discount offer

PAGSEG Print variable page segment, using:
 FLWRNB Seed Image
 STRWNB Strawberry Image
 TREENB Tree Image
 BETLNB Beetle Image

PRTOVL Print variable overlay, as follows:
 INVALL Whole or 1-page invoice overlay
 INVST First page of multipage invoice overlay
 INVMID Middle page of multipage invoice overlay
 INVLST Last page of multipage invoice overlay

RPG Source for New Invoice Output (INVNEW1)

This is the RPG source used to produce the new Super Sun Seeds invoice output example shown in Figure 58 on page 141.

```

5722WDS V5R2M0                      SEU SOURCE LISTING      10/10/02 17:47:39    1
SOURCE FILE . . . . . SAMPLER/QRPGSRC
MEMBER . . . . . INVNEW1
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
 100 * INVNEW1 - Super Sun Seeds Electronic Invoice
 200 *           Example 1
 300 *
 400 * This program reads a transaction file (SEEDDETL), retrieves info
 500 * from item master (SEEDITEM) and customer master (SEEDCUST), then
 600 * prints invoices.
 700 *
 800 * Different electronic overlays are used depending on whether this
 900 * is a one page or multipage invoice, as well as whether there is
1000 * room remaining on the invoice to print a customer offer. The following
1100 * overlays are used:
1200 *
1300 *   INVALL   1-page invoice (full address at top, payment at bottom)
1400 *   INVST   Page 1 of multipage invoice (no payment at bottom)
1500 *   INVMID  Continuation page (No address, payment at bottom)
1600 *   INVLST  Last page of multipage invoice
1700 *
1800 * Invoice detail lines are processed until either end of customer order
1900 * or lines=35, at which time the invoice page is printed. At end of
2000 * order, a check is made to see if there are more than 18 detail lines
2100 * to print. If there are, that would not leave room in the invoice body
2200 * to print the customer offer. In this case, the page is printed, and a
2300 * new final page is built with the offer and the payment coupon.
2400 *
2500 * The offer is a message (and corresponding image, or page segment) based
2600 * on item sales in the customer master. For example, a flower image
2700 * would print if seed sales exceeded $500.
2800 *
2900 * ENHANCEMENTS:
3000 * See program INVNEW2 for the following functional enhancements:
3100 *
3200 *   1. Multiple copies of invoice with routing designation, such as
3300 *      "Customer Copy", "File Copy", etc.
3400 *   2. Overlay specifying terms and conditions to print on back of
3500 *      customer copy.
3600 *   3. Packing List copy that suppresses prices
3700 *
3800 FSEEDDETLIP E          K          DISK
3900 FSEEDITEMIF E          K          DISK
4000 FSEEDCUSTIF E          K          DISK
4100 FINVNEW1 O  E          PRINTER
4200 *
4300 E                      WDS      1  40 24          Text for Offer
4400 *
4500 IINVCDETL  01
4600 I                      STNAME      32

```


11400	CL1		MOVE BARTOT	BARPRC	150		Load Totals
11500	CL1		Z-ADDTOTDUE	TOTD@2	92		
11600	CL1		MOVE NAME	NAME@2	25		
11700	CL1		MOVE STREET	STRE@2	25		
11800	CL1		MOVE CITY	CITY@2	25		
11900	CL1		MOVE STATE	STAT@2	2		
12000	CL1		Z-ADDZIP	ZIP@2	90		
12100	C*						
12200	CL1		EXSR OFFSR				
12300	CL1	PAGCNT	IFGT 0				
12400	CL1		MOVE 'INVLST	'OVLAY	8		Cont. Overlay
12500	CL1		ELSE				
12600	CL1		MOVE 'INBALL	'OVLAY			1 Page Invoice
12700	CL1		ENDIF				
12800	CL1	30	WRITEOFFER				Write Offer and
12900	CL1	30	WRITEPAGESEG				Page Segment
13000	CL1		WRITEINVBOT				Invoice Totals
13100	CL1		WRITEPRTOVL				Print Overlay
13200	CL1		WRITEENDPAGE				End Page
13300	CL1		SETOF			30	
13400	C*	Set up Date					
13500	CSR	DATESR	BEGSR				
13600	CSR		Z-ADDUDAY	PAYDA			
13700	CSR	UMONTH	ADD 1	PAYMO			
13800	CSR	PAYMO	IFGT 12				
13900	CSR		Z-ADD1	PAYMO			
14000	CSR	UYEAR	ADD 1	PAYYR			
14100	CSR		ELSE				
14200	CSR		Z-ADDUYEAR	PAYYR			
14300	CSR		ENDIF				
14400	CSR		MOVE PAYDAA	PAYDAT	60		
14500	CSR		Z-ADDPAYDAT	PAYDA@	60		
14600	CSR		ENDSR				
14700		* Set up Offer at end of invoice					
14800	CSR	OFFSR	BEGSR				
14900	CSR	SLSSED	IFGE 500				
15000	CSR		Z-ADD1	IX	20		
15100	CSR		MOVE 'FLWRNB	'PSEG	8		Seed
15200	CSR		SETON			30	
15300	CSR		GOTO WRTOFR				
15400	CSR		ENDIF				
15500	CSR	SLSFRT	IFGE 500				
15600	CSR		Z-ADD7	IX			Strawberry
15700	CSR		MOVE 'STRWNB	'PSEG			
15800	CSR		SETON			30	
15900	CSR		GOTO WRTOFR				
16000	CSR		ENDIF				
16100	CSR	SLSSUP	IFGE 500				
16200	CSR		Z-ADD13	IX			
16300	CSR		MOVE 'TREENB	'PSEG			Tree
16400	CSR		SETON			30	
16500	CSR		GOTO WRTOFR				
16600	CSR		ENDIF				
16700	CSR	SLSCHM	IFGE 500				
16800	CSR		Z-ADD19	IX			
16900	CSR		MOVE 'BETLNB	'PSEG			Beetle
17000	CSR		SETON			30	
17100	CSR		GOTO WRTOFR				
17200	CSR		ENDIF				
17300	C*						
17400	CSR		GOTO ENDOFR				
17500	C*						
17600	CSR	WRTOFR	TAG				
17700	CSR		MOVE WDS,IX	OFFR@1	24		Build Offer Text
17800	CSR		ADD 1	IX			
17900	CSR		MOVE WDS,IX	OFFR@2	24		
18000	CSR		ADD 1	IX			


```

18100      CSR                MOVE WDS,IX    OFFR@3 24
18200      CSR                ADD 1             IX
18300      CSR                MOVE WDS,IX    OFFR@4 24
18400      CSR                ADD 1             IX
18500      CSR                MOVE WDS,IX    OFFR@5 24
18600      CSR                ADD 1             IX
18700      CSR                MOVE WDS,IX    OFFR@6 24
18800      CSR                ENDOFR          ENDSR
18900      C*
19000      *
19100 ** WDS      WORDS FOR OFFER
19200 Thank You .....
19300 Because you have ordered
19400 over $500 of seeds this
19500 year, on your next seed
19600 order you will receive
19700 a 10% discount.
19800 Thank You .....
19900 Because you have ordered
20000 over $500 of fruit this
20100 year, on your next fruit
20200 order you will receive
20300 a 10% discount.
20400 Thank You .....
20500 Because you have ordered
20600 over $500 of trees this
20700 year, on your next tree
20800 order you will receive
20900 a 10% discount.
21000 Thank You .....
21100 Because you have ordered
21200 over $500 of stuff this
21300 year, on your next stuff
21400 order you will receive
21500 a 10% discount.
21600

```

* * * * E N D O F S O U R C E * * * *

The RPG program shown above uses the logic described in the following sections to build the Super Sun Seeds invoices.

Beginning the Invoice

To begin the invoice, do this:

1. Read Customer Master (SEEDCUST)
2. Reset invoicing totals

Processing Items

To process the items on the invoice, do this:

1. Read records from transaction file (SEEDDETL)
2. Do item calculations
3. Write detail line using DETLIN format

Overflow Items (ITMCNT = 35)

To process overflow items, do this:

1. If page 1, use overlay INVST
2. If page n, use overlay INVMID
3. Write PRTOVL to print overlay
4. Write INVTP2 to eject page and print continuation header

Ending the Invoice (No Room for Offer)

If there is no more room on the invoice, end it by doing this:

1. If page 1, use overlay INVST
2. If page n, use overlay INVMID
3. Write PRTOVL to print overlay
4. Write INVTP2 to eject page and print continuation header

Ending the Invoice

If the invoice still has room left on it but it should be ended anyway, do this:

1. If page 1, use overlay INVST or INVALL
2. If page n, use overlay INVLST
3. Select page segment image and discount offer
4. Write OFFER to print offer
5. Write PAGSEG to print image
6. Write INVBOT to print invoice total/payment coupon
7. Write PRTOVL to print overlay

Referring again to the DDS printer specifications for the new Super Sun Seeds application, the special functions can be summarized in each print record as follows:

INVTOP	All program fields are printed in the default font (Courier). The 9-digit zip code is printed in postal bar code. The ship to and bill to lines use standard (based on 6 LPI) spacing and skipping. No exact positioning (down and across) is done.
INVTP2	Continuation header is printed, again with standard skipping and spacing.
DETLIN	Detail lines are printed in Courier at 10 lines per inch.
INVBOT	Payment coupon is printed. A field containing the customer number and total amount due is printed in Interleaved 2 of 5 bar code.
OFFER	The special discount offer is printed in FONT(92), or Courier Italic 12-pitch.
PAGSEG	The program passes the page segment name (based on customer sales data), and the page segment is printed 7.0 inches down and 2.6 inches across the page.
PRTOVL	The program passes the appropriate overlay in the OVLAY field, and it is printed at 0 inches down and 0 inches across the page.

Enhancing the Super Sun Seeds Invoice

The Super Sun Seeds invoice is an effective business document, making use of electronic forms, bar codes, custom images and marketing messages. Because it is an electronic document, it easily can be updated or changed.

A number of enhancements to the Super Sun Seeds invoicing application can significantly add to its value as a document:

- A standard set of terms and conditions can be printed on the back side of selected pages. This is called a constant back form or overlay.
- A set of collated copies of the invoice can be automatically produced. A packing list and a file copy, for example.
- The information about the copies can be tailored. For example, pricing information about the packing list can be suppressed.
- The variable positioning capabilities of DDS can be used to create a dynamic or "floating" invoice, one in which the form itself is tailored exactly to customer data.

Two additional Super Sun Seeds invoice examples, INVNEW2 and INVNEW3, demonstrate how to add the functions described above. INVNEW2 implements the copies, price suppression, and constant back overlay. INVNEW3 adds the dynamic, or floating, form. The RPG and DDS source for INVNEW2 and INVNEW3, as well as the output for both examples can be found in Appendix B, “Additional DDS Examples,” on page 297.

The following sections show how the enhancements described above were made.

Specifying Constant Back Overlays

An invoice or similar document might need a set of instructions or terms and conditions printed on the back side. This cannot be done with the BACKOVL parameter of the printer file because it cannot force the required page eject. For the Super Sun Seeds application, simply use the PRTOVL record format to print the back side overlay and add the ENDPAGE keyword to force the page eject. A sample of the terms and conditions form (INVBAC) is shown in Figure 40 on page 98.

Specifying Copies

A typical invoicing application requires more than just the customer copy of the bill. Copies of the invoice, such as a packing list, a file copy, and an accounting copy are common. In addition, the paper workflow might require that the copies are either collated or uncollated.

For uncollated copies, the easiest technique is to run the application multiple times, creating a separate spooled file for each copy. The application runs could be tailored to use different overlays for the different copies, or to simply print the copy designation (for example, **Packing List**) in text at the bottom of each copy.

Collated copies normally require saving some of the invoicing information in order to print the multiple copies at either a page or invoice break. Because electronic printing is page-mode printing, you need to save the page contents to reprint the page as a copy.

Figure 59 on page 150 shows a sample of the “Customer Copy”, one of three copies created in sequence by INVNEW2.

Chapter 15. Using Advanced Print Utility

Advanced Print Utility (APU), a part of AFP PrintSuite, is an end-user utility for transforming existing application output to advanced electronic documents. APU enables you to transform new and existing applications with line-mode output and preprinted forms to full-function electronic documents without changes to the “line of business” application program. Generally, Infoprint Designer provides most of the design functions of APU, and has a fully graphical design interface.

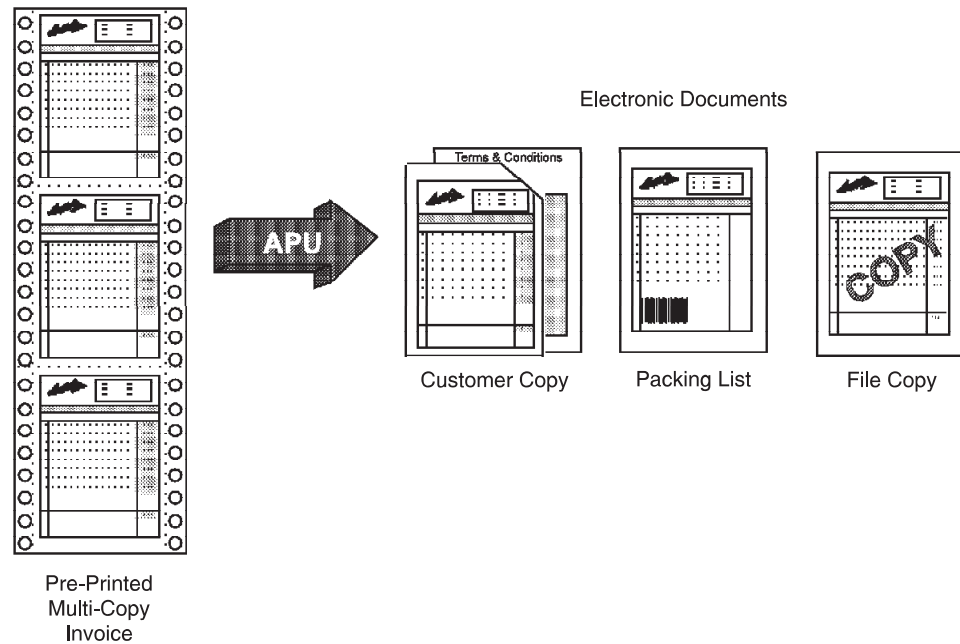


Figure 61. Typical APU Application

Figure 61 shows a typical example with invoices. With APU, you can use data that produces multi-copy preprinted invoices as input to create electronic pages with customized copies.

APU, designed to build complex documents, has a simple end-user interface. Documents can include such elements as multiple page formats, multiple copies, and standard back overlays. APU provides application data remapping, which lets you completely change the way the application data is printed. APU also provides output conditional processing, where data in the existing spooled output is used to determine document layout and flow.

Figure 62 on page 154 illustrates the APU print flow.

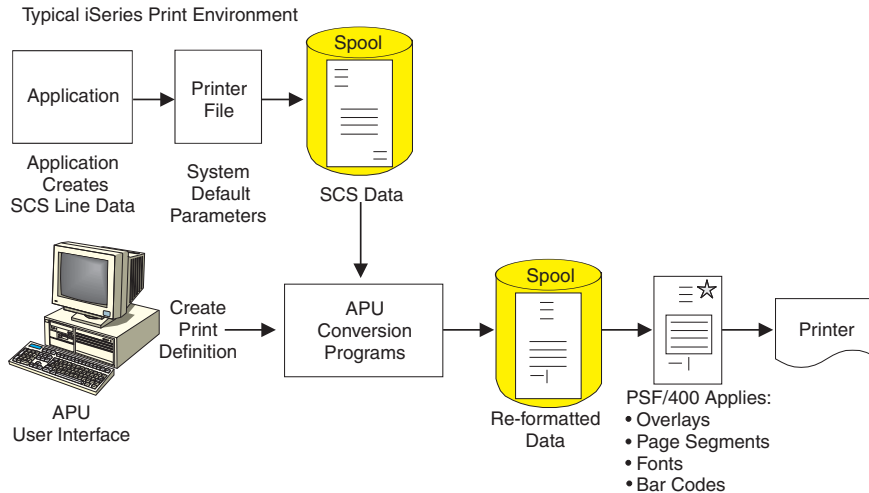


Figure 62. APU Print Flow

APU has a design phase and a production phase. This design process (which is done one time) includes retrieving a sample spooled SCS file and displaying it for use. The user can then create a print definition, which can define conditional logic, redefine data on the page, change fonts, and specify overlays, page segments, and bar codes.

After the new design is ready, you can use the APU Print Monitor or the Apply Print Definition command (APYPRTDEF) to apply the Print Definition to the target SCS spooled file to create the new output. The new output is re-spooled. At this point, PSF takes over, retrieves required external print resources, and manages the printing of the job.

What You Can Do with APU

Output specifications for iSeries application programs generate either SNA Character Stream (SCS) or AFP spooled files. APU works on SCS spooled files. SCS is a line-oriented datastream that for the most part uses preprinted forms to create the final document. With APU, you can eliminate the need for preprinted forms. Instead, you create a completely electronic document. With APU, you can:

- Create multi-copy documents, with each page customized.
- Use data that is contained within a page to determine which of multiple output formats to use.
- Remap any field that the input SCS pages contain (change position, font, orientation, color, and so on).
- Print application data in any of the standard bar code symbologies.
- Add document elements such as electronic forms (overlays), images, lines, boxes, and constant text.
- Place a new application into production for automatic processing.
- Manage the production of input and output files, including the routing of different output files to different queues, printers, and output bins.
- Implement user-defined programs that can address unique document or distribution requirements.

APU provides an interactive interface for defining new output applications. For simpler applications, APU provides a fast path. You use the current spooled file (SCS) interactively to redefine the formatting of application data.

Why Use APU?

APU helps you build AFP-compatible electronic output. Effective electronic output can provide significant benefits to an organization, particularly in the areas of information systems costs, process reengineering, and better communications, for example:

- Replacing preprinted multipart forms with electronic forms supplies significant cost savings.
 - You can print a variety of different forms one after the other without switching forms at the printer.
 - You can eliminate carbon forms by printing multiple copies of the same page, that includes the capability of varying the output on each copy.
 - You can eliminate storage space for preprinted forms because the forms are stored electronically rather than physically.
 - You can change the form outside of the application program, and you do not need to scrap or reorder preprinted forms.
- Documents, whether printed, stored, viewed, or e-mailed, are critical to the workflow in any organization. The capabilities of electronic documents provide a wealth of opportunities to reengineer organizational processes. In fact, in many industries, document reengineering such as bar coding is a “must.” Coding an electronic document with bar code, optical character recognition (OCR), magnetic ink character recognition (MICR), and images enables you to easily integrate the document into the workflow.
- Electronic documents are more effective. Document elements, such as images, text, and overlays, allow you to compose a document that does a better job of communicating or marketing. Electronic flexibility, the ability to change a document dynamically, down to the individual transaction level, provides a wide variety of application possibilities. Electronic documents project the image of a strong, professional organization.

Data Mapping and Field Selection

APU enables you to select a target spooled file and bring that file into the user definition flow for spooled data remapping and field selection. This sample spooled file is brought up in Non-Programmable Terminal (NPT) mode and enables you to:

- Select and define fields for remapping, including:
 - Changing position, font, color, or rotation of field
 - Printing the field separately as bar code
 - Suppressing the printing of a field
- Select and name fields to be used with conditional logic to select and control the page layout

APU Data Structure

APU redefines the output structure by creating a print definition that contains page formats, as well as copy and page layout information. Figure 63 on page 156 shows this structure.

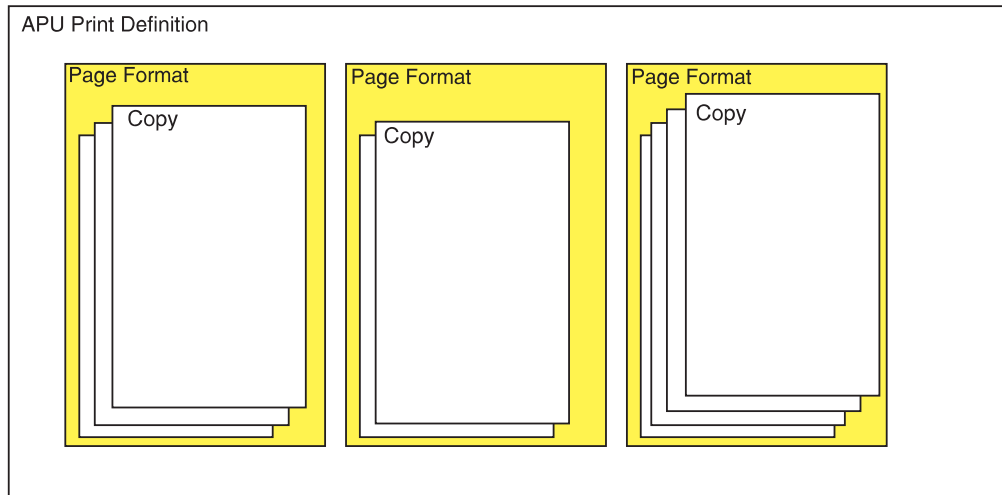


Figure 63. APU Data Structure

The structure is as follows:

Print Definition

The object that contains all of the instructions for formatting a print job. A print definition can contain one or more page formats. A print definition with one page format is considered a “fast-path” definition.

If multiple page formats are specified, the print definition also specifies the conditions under which each page format is selected. The conditions are defined in the form of tests on fields of data in the existing output that you can specify. For example, you can specify that columns 1-5 in row 1 of the existing output data contain a “trigger” field to be analyzed. If that field is *JOB 1*, then the page format INVOICE is to be used. If that field is *JOB 2* then the page format SHIPPING is to be used.

Page Format

The level within a print definition that contains all of the instructions for processing each input page that meets a certain set of criteria. A page format can contain one or more copies.

Copy

The level within a page format that contains all of the instructions for formatting a single output page for a single input data page. Using multiple copies, APU can produce multiple pages from a single input page, with each output page having different characteristics (customer copy, file copy, and so on).

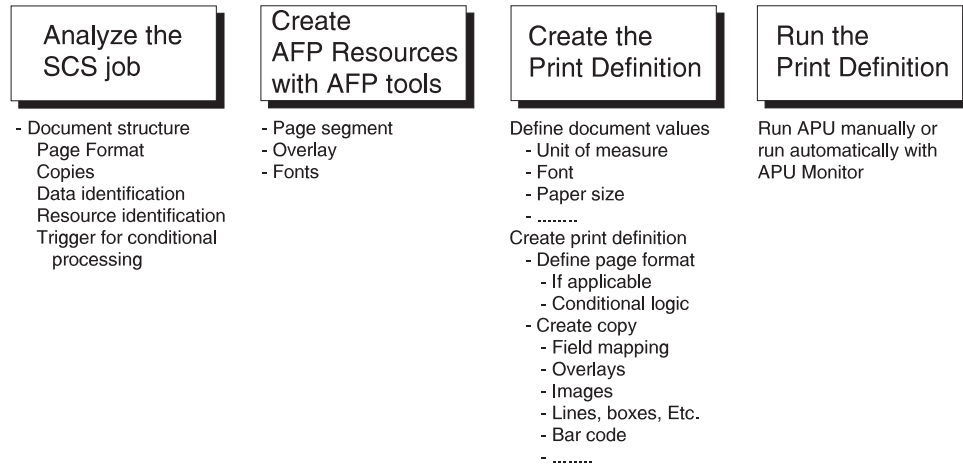


Figure 64. APU Step by Step

Figure 64 shows a step-by-step view of APU.

1. Analyze the current SCS job. Determine what the new document structure and layout will be, what triggers exist for conditional processing, what resources are required.
2. Create the required resources - overlays and page segments (images).

Note: You can use Infoprint Designer to create these resources. However, if you're using Infoprint Designer for resources, you should be using it for document composition (layout) as well.

3. Build the APU print definition. This design process flows with the new document structure. The general steps include:
 - Set overall document values
 - Define each page format, and within that:
 - Define each copy
 - Go to the next page format (if more than one)
4. Run the Print Definition. After you have completed the design, you can run APU either manually (to test, for example) or start the APU Print Monitor for automatic operation.

Example of a Single-Page Format Document

Following are two illustrations:

- "Example of the SCS File to be Formatted" shows the SCS file to be transformed by APU
- "Example of the Formatted File" on page 158 shows the same file after it has been transformed into an AFP file by APU

Example of the SCS File to be Formatted

IMPROVED PRINTING CORP		SAME	
PERFORMANCE BOULEVARD			
PRINTERSVILLE			
CO 45789-2637			
100	31300	1/26/02	2/26/02
BEST WAY	1/26/02	NET 30	YOUR PRINTER REP
1 CT 00000300	HIGH ALTITUDE	WATERMELON	1.01 1.01
1 PK 01100517	SPARTAN SEEDS		2.39 2.39
9 PK 04569870	NORTHERN LITE	BLUE SPRUCE	858.32 7,724.88

12 BX	11005004	BUSH GREEN SEEDS	2.50	30.00
12 CT	11005011	LASSO RED SEEDS	892.23	10,706.76
26 PK	11005018	EARLY BANTAM SEEDS	.38	9.88
5 BX	11057893	AFRICAN DAISY, SEEDS	2.35	11.75
1 PK	15975365	HEAVY OAK	129.09	129.09
33 BX	32746510	HOPS BREWING LIGHT	1.20	39.60
6 EA	46578913	SEED SURVEYING SITE	50.00	300.00
2 BX	56413213	POT POT	7.65	15.30
80 PK	65412384	SEED SCRUBBER	888.79	71,103.20
1 PK	84512023	OREGON SPRING TOMATO SEED	.97	.97
2 DZ	96325874	PINEAPPLE-ORANGE SEEDS	1.29	2.58
11 BX	98412006	BLACK BEAUTY ZUCCHINI	2.30	25.30
5 EA	98546320	FROZEN JUICE PROCESSOR	109.90	549.50

Thank You
 Because you have ordered
 over \$500 of seeds this
 year, on your next seed
 order you will receive
 a 10% discount.

\$90,652.21

2/26/02

\$90,652.21

IMPROVED PRINTING CORP
 PERFORMANCE BOULEVARD
 PRINTERSVILLE
 CO 457892637

Example of the Formatted File

The formatted output that you want to produce is shown in Figure 65.



400 CPU Parkway Vegetation, NJ 55090		 Super Sun Seeds A Growth Company		Office: 555-499-2367 Fax: 555-415-9794	
IMPROVED PRINTING CORP PERFORMANCE BOULEVARD PRINTERSVILLE CO 45789-2637			SAME		
-- Sold To --		-- Ship To --			
Customer Number: 100	Invoice Number: 31300	Invoice Date: 1/22/98	Payment Date: 2/22/98		
Ship Via: BEST WAY		Shipped Date: 1/22/98	Terms: NET 30	Salesman: YOUR PRINTER REP	
Qty	UOM	Item	Item Description	Price	Extension
1	CT	00000300	HIGH ALTITUDE WATERMELON	1.01	1.01
1	PK	01100517	SPARTAN SEEDS	2.39	2.39
9	PK	04569870	NORTHERN LITE BLUE SPRUCE	858.32	7,724.88
12	BX	11005004	BUSH GREEN SEEDS	2.50	30.00
12	CT	11005011	LASSO RED SEEDS	892.23	10,706.76
26	PK	11005018	EARLY BANTAM SEEDS	.38	9.88
5	BX	11057893	AFRICAN DAISY, SEEDS	2.35	11.75
1	PK	15975365	HEAVY OAK	129.09	129.09
33	BX	32746510	HOPS BREWING LIGHT	1.20	39.60
6	EA	46578913	SEED SURVEYING SITE	50.00	300.00
2	BX	56413213	POT POT	7.65	15.30
80	PK	65412384	SEED SCRUBBER	888.79	71,103.20
1	PK	84512023	OREGON SPRING TOMATO SEED	.97	.97
2	DZ	96325874	PINEAPPLE-ORANGE SEEDS	1.29	2.58
11	BX	98412006	BLACK BEAUTY ZUCCHINI	2.30	25.30
5	EA	98546320	FROZEN JUICE PROCESSOR	109.90	549.50
Total Due					\$90,652.21
Return this tear-off strip with your payment. Payment is due by: 2/22/98			Make Checks Payable to: Super Sun Seeds Amount Due is: \$90,652.21		
IMPROVED PRINTING CORP PERFORMANCE BOULEVARD PRINTERSVILLE CO 457892637					
Page 1					

Figure 65. Super Sun Seeds Invoice

Refer to *Advanced Print Utility for iSeries: User's Guide* for more information about APU.

Super Sun Seeds Case Study

The APU example of the Super Sun Seeds case study is a simple, straightforward application. As shown in Chapter 12, "Introduction to the Super Sun Seeds Case Study," on page 113, the existing Super Sun Seeds invoicing application is a preprinted invoice produced by program INVPRE. APU is used to transform the output of INVPRE to a multi-copy electronic invoice.

To do this, an APU print definition (SUPER) is built that will:

- Specify only one page format, the basic invoice page
- Select a sample spooled file (from INVPRE)
- Automatically create the original copy (*ORIGINAL)
- Define the composition of the original or customer copy, as follows:
 - Specify general page layout options, including the constant terms and conditions back overlay
 - Map data in the spooled file, such as the zip code to POSTNET bar code
 - Write "Customer Copy" at bottom on copy
 - Place INVALL overlay on front
- Define two additional copies, the packing list and file copies, by first copying the definition of the *ORIGINAL
- Modify the packing list copy, suppressing pricing information and printing "Packing List" as constant text
- Modify the file copy, printing "File Copy" at the bottom

Stepping through the following APU displays can help you to understand how the print definition process works. Starting from the APU main menu, select option 1 to create a print definition.

```
APU                                IBM Advanced Print Utility

Select one of the following:

  Build and Test APU Print Definitions
    1. Work with Print Definitions
    2. Work with Spooled Files

  Run APU in Batch Mode
    3. Work with APU Monitor
    4. Start APU Monitor
    5. End APU Monitor

  Configure APU
    6. Set APU Defaults
    7. Work with Fonts
    8. Configure APU Monitor Action

Selection or command
===> 1
F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F16=System main menu
F23=Set initial menu
```

Figure 66. APU Main Display

Other functions you can select from the main menu include:

- Working with output queues, applying APU print definitions to selected spooled files
- Managing the APU Monitor, which automatically selects target spooled files and transforms them with an APU print definition
- Setting the APU defaults, which includes measurement method, libraries, and code page
- Working with fonts, which maintains the interface to the fonts used on the system

From the “Work with Print Definitions” display, create a print definition called SUPER. Define SUPER as a print definition with a single page format.

```

Work with Print Definitions

Library . . . . . APUDATA          Name, *CURLIB

Type options, press Enter.
 1=Create  2=Change  3=Copy   4=Delete  5=Display contents
 6=Print contents  7=Rename  10=Define  12=Work with

Opt Name      Text
 10 SUPER     Super Sun Seeds Invoice

Bottom

F3=Exit  F5=Refresh  F12=Cancel

```

Figure 67. Work with Print Definitions Display

When you select option 10 (Define), the Define a Print Definition display appears. Select the Set print definition attributes option to set attributes for the SUPER print definition.

```

Set Print Definition Attributes

Print Definition . . . : SUPER
Library . . . . . : APUDATA

Type choices, press Enter.

Unit of Measure . . . . *INCH          *INCH, *CM, *ROWCOL, *UNITS
Default line increment *INPUT *INCH          *INPUT, Value
Default column inc. . . *INPUT *INCH          *INPUT, Value
Page length . . . . . 11 *INCH          *INPUT, Value
Page width . . . . . 8.5 *INCH          *INPUT, Value
Top margin (down) . . . 0 *INCH          0, Value
Left margin (across) . . 0 *INCH          0, Value
Page orientation . . . . 0 *INCH          0, 90, 180, 270
Apply field attributes  1=Yes

More...

F3=Exit  F12=Cancel  F22=Set Units

```

Figure 68. Print Definition Attributes (First Display)

The print definition attributes include page characteristics, the default font family, and resource libraries.

Note: Type **INPUT* if you want to use the value contained in the input spooled file.

```

Set Print Definition Attributes

Print Definition . . . : SUPER
Library . . . . . : APUDATA

Type choices, press Enter.

Default font family . . *APUDFT          *APUDFT, Value   F4 for List
Point size . . . . . *CALC            *CALC, Value
Bold . . . . .          1=Yes
Italic . . . . .        1=Yes
Default Color . . . . . *APUDFT          *APUDFT, Value   F4 for List
Addl. resource libs. . QGPL              Name
                                      Name
                                      Name
                                      Name
Bottom

F3=Exit  F4=Prompt  F12=Cancel

```

Figure 69. Print Definition Attributes (Second Display)

After you have defined these print definition attributes and pressed Enter, the “Work with Print Definitions” display appears again.

Note: If you specify *CALC, the point size is determined by the value specified in the input spooled file.

```

Work with Print Definitions

Library . . . . . APUDATA          Name, *CURLIB

Type options, press Enter.
1=Create  2=Change  3=Copy  4=Delete  5=Display contents
6=Print contents  7=Rename  10=Define  12=Work with

Opt  Name      Text
12  SUPER     Super Sun Seeds Invoice
Bottom

F3=Exit  F5=Refresh  F12=Cancel

```

Figure 70. Work with Print Definitions Display

Select option 12 to display the Work with Copies display. On this display you can define the first copy.

```

Work with Copies

Print Definition . . . : SUPER          Page Format . . . . . : *DEFAULT
Library . . . . . : APUDATA

Type options, press Enter.
1=Create  2=Change  3=Copy  4=Delete  7=Rename
10=Define

Opt  Name      Text
10  *ORIGINAL  Original (first copy)
Bottom

F3=Exit  F5=Refresh  F12=Cancel

```

Figure 71. Work with Copies Display


```

                                Set Page Layout Options
Print Definition . . . SUPER          Page Format . . . . . : *DEFAULT
  Library . . . . . APUDATA        Copy . . . . . : *ORIGINAL

Type choices, press Enter.

Input drawer . . . . . *DEFAULT          *DEFAULT, 1, 2, 3, 4
Default line increment *PRTDEF *INCH      *PRTDEF, *INPUT, Value
Default Column inc. . . *PRTDEF *INCH      *PRTDEF, *INPUT, Value
Page length . . . . . *PRTDEF *INCH      *PRTDEF, *INPUT, Value
Page width . . . . . *PRTDEF *INCH      *PRTDEF, *INPUT, Value
Top margin (down) . . . *PRTDEF *INCH      *PRTDEF, 0, Value
Left margin (across) . . *PRTDEF *INCH      *PRTDEF, 0, Value
Page orientation . . . *PRTDEF          *PRTDEF, 0, 90, 180, 270
Duplex Printing . . . . 1=Yes, 2=Tumble
Back overlay . . . . . INVBAC          *NONE, Name F4 for list
  Position across . . . 0 *INCH          0, Value
  Position down . . . . 0 *INCH          0, Value

F3=Exit F4=Prompt F12=Cancel F22=Set Units

```

Figure 74. Set Page Layout Options Display

On the Set Page Layout Options display, select the general page layout options for the customer copy. You can specify the:

- Input drawer for this page to be drawn from
- Line and column increments for this page, such as 6 lines per inch for line and 10 characters per inch for column
- Page length and width; in this example, 8-1/2 by 11 inches
- Page margins and orientation; in this example, all zero

Many of these values default to *PRTDEF, which points to values set for this print definition. However, these values can also be different, which means that an individual copy can have different orientation, margins, input selections, and other characteristics.

Specify the Terms and Conditions overlay (INVBAC), shown below, as the back overlay.

Terms and Conditions

The sale of the products described herein shall be governed by the terms and conditions contained in any written contract currently in effect between Buyer and Seller covering such sale. If there is no such contract, then Seller hereby offers to sell such products to Buyer only upon the terms set forth herein.

1. Seller's standard prices in effect at the time of shipment will govern the sale of the products described herein. Terms of payment on any approved order are net thirty (30) days from the date of invoice unless otherwise specifically stated. All shipments, unless specifically provided, shall be to the place of manufacture or warehouse location indicated hereon. The price includes cost of packaging for domestic shipment, unless otherwise stated. An additional charge will be made for special domestic or export packing if this involves greater expense. Shipments will be insured at the expense of the Buyer unless Buyer specifically requests that shipments not be insured.
2. Seller reserves the right among other remedies a hereto to terminate this contract or suspend further deliveries under it in the event Buyer fails to pay for any shipment when same becomes due. Should Buyer's financial responsibility become unsatisfactory to Seller, cash payments or satisfactory security may be required by Seller for future deliveries and for goods theretofore delivered.
3. In addition to the purchase price, Buyer shall pay to Seller the amount of all taxes, excises or other charges (except taxes on or measured by net income) that Seller may be required to pay to any Government (national, state or local) with respect to the production, sale or transportation of any product delivered hereunder, except where the law otherwise provides. Sixty (60) days written notification must be given to make any changes to delivery schedule.
4. Seller warrants that products delivered hereunder will conform to the description on the face of this document and meet any specifications set forth or incorporated by reference herein and will be adequately contained, packaged and labeled and conform to any promises and affirmations of fact made on the container and label. Seller further warrants any such product against defects in workmanship or materials which develop or become evident within 12 (9 months OEM) after shipment of the product by Seller provided Seller's responsibility under this warranty shall be limited to the repair or if Seller so elects the replacement of the defective product or component thereof and that said warranty is subject to the following exceptions and conditions:
 - (a) All items claimed to be defective must be returned to Seller, transportation charges prepaid and will be returned to Buyer transportation charges collect unless found to be defective in which case Seller will pay all transportation charges.
 - (b) Seller's warranty will not apply to items which have been modified or repaired without Seller's written consent by persons other than Seller's authorized service personnel.
5. It should be noted that Seller has used the powerful tools of AS400 Advanced Function Printing to create Seller's invoice, and despite the arcane terms and conditions which are presently being read, the invoice is a very effective business document. Any attempt to mar the appearance of this invoice should be discouraged.
6. The validity, interpretation and performance of the terms hereof with respect to any product delivered (or to be delivered) hereunder shall be governed by the law of the State of Colorado.
7. No modification or waiver of the terms here of shall be binding upon Seller unless approved in writing by one of Seller's Officers or Marketing Managers or shall be affected by acknowledgment or acceptance of purchase order forms containing other or different terms whether or not signed by authorized representative of Seller. (e) Seller does not warrant items which have been damaged due to negligence or misuse.
8. THERE ARE NO EXPRESS WARRANTIES BY SELLER OTHER THAN THOSE SPECIFIED IN THIS PARAGRAPH. NO WARRANTIES BY SELLER (OTHER THAN WARRANTY OF TITLE AS PROVIDED IN THE UNIFORM COMMERCIAL CODE) SHALL BE IMPLIED OR OTHERWISE CREATED UNDER THE UNIFORM COMMERCIAL CODE, INCLUDING BUT NOT LIMITED TO WARRANTY OF MERCHANTABILITY IN OTHER RESPECTS THAN SPECIFIED IN THIS PARAGRAPH AND WARRANTY OF FITNESS FOR A PARTICULAR PURPOSE. Without limiting the generality of the foregoing, Buyer assumes all risk and liability for the results obtained by the use of any products delivered hereunder in combination with other articles or material in the practice of any process.
9. NO CLAIM of any kind with respect to any product specified herein, whether as to product delivered or for nondelivery of product and whether or not based on negligence, SHALL BE GREATER IN AMOUNT than the purchase price of the product in respect of which such claim is made. In no event shall either party be liable for special indirect or consequential damages whether or not caused by or resulting from the negligence of such party.
10. It is expressly understood that any technical advice furnished by Seller with reference to the use of its products is given gratis and Seller assumes no obligation of liability for the advice given or results obtained, all such advice being given and accepted at Buyer's risk.
11. In the event Buyer fails to fulfill Seller's terms of payment for the products specified herein, or in case Seller shall have any doubt at any time as to Buyer's financial responsibility, Seller may decline to make further deliveries except upon receipt of cash or satisfactory security.

Figure 75. Back Overlay (Terms and Conditions) - INVBC

After you have specified the Terms and Conditions overlay, you need to define field mapping for your output. Select the Define field mapping option on the Define a Copy display and press Enter. The Define Field Mapping display, shown below, appears.

Define Field Mapping

Spooled file : INVPRE Page/Line : 1/1
 Control Columns : 1 - 78
 *...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...

IMPROVED PRINTING CORP SAME
 PERFORMANCE BOULEVARD
 PRINTERSVILLE
 CO 45789-2637

More...

F3=Exit F11=Hide mapping F12=Cancel F14=Start field
 F15=End field F16=Delete range F20=Right

Figure 76. Define Field Mapping Display

APU displays the Super Sun Seeds invoice spooled file for data mapping. With data mapping, you can redefine how the application data is placed on output, including:

- Moving fields to new locations
- Changing how fields are printed, such as font and color

- Printing fields as bar code
- Suppressing fields from printing

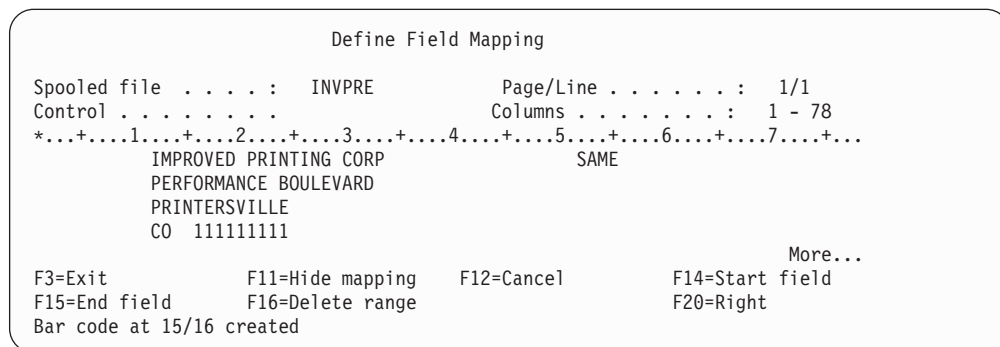


Figure 77. Define Field Mapping Display

To reprint the zip code in the name and address in POSTNET bar code, you first define the zip code field to APU. Use F14 to specify the beginning of the field and F15 to specify the end of the field.

Select what you want to do with the zip code field. In this case, map it as bar code.

Note: To successfully print your POSTNET bar code mapping, you must specify the correct number of digits (5 or 9, for example) for the type of POSTNET bar code you selected. If you do not map enough numeric digits, question marks (?) appear in the Bar code data field to indicate that you need to map the correct number of digits. If you do not map the correct number of digits, your bar code will not print.

Non-numeric characters, such as the dash contained in the ZIP + 4 code, are removed for you.

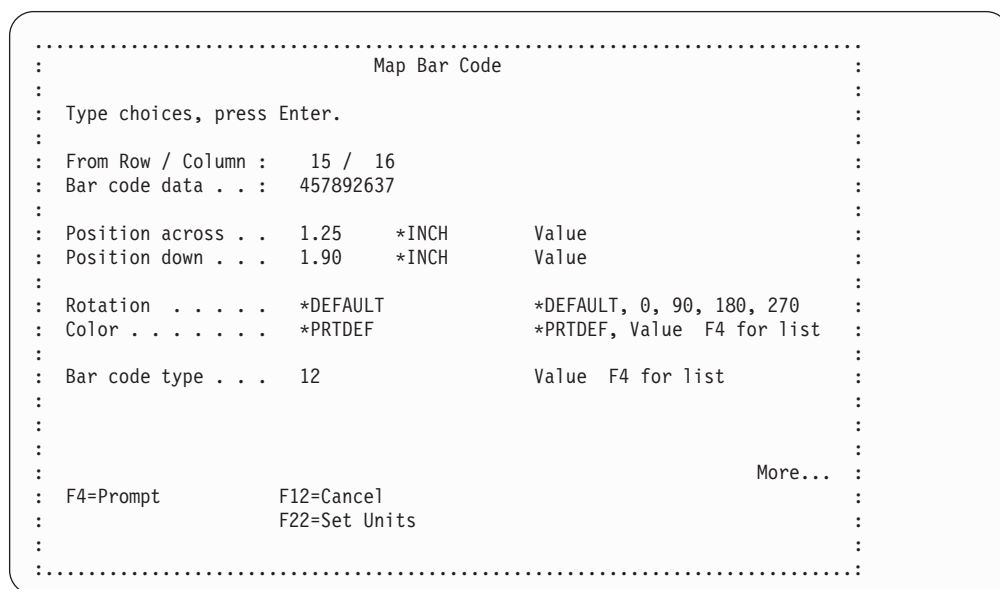


Figure 78. Map Bar Code Display

The Map Bar Code display is a multi-screen pop-up used to select, define, and position the bar code. The zip code is at row 15, column 16, and the value is

shown. Map it (make a copy of the zip code) in bar code to a position starting at 1.25 inches down and 1.90 inches across the paper. It could have also been positioned by row and column. In either case, it will appear just above the name and address. Position the cursor in the Bar code type field and press F4 to display a list of supported bar code types. Select 12, which is POSTNET. Press Enter to return to the Define Field Mapping display. Press F3. The Define Constants display appears.

```

Define Constants

Print Definition . . . : SUPER          Page Format . . . . . : *DEFAULT
Library . . . . . : APUDATA      Copy . . . . . : *ORIGINAL

Type options, press Enter.
1=Create  2=Change  3=Copy  4=Delete
  Constant  Position  Position  Unit of
Opt type   across   down    measure  Constant value
1  *TEXT

F3=Exit  F5=Refresh  F12=Cancel

```

Figure 79. Define Constants Display

Next, define the constant text Customer Copy to appear at the bottom of this copy. Specify a text type for this constant. You could also specify a constant bar code. Select option 1 and press Enter. The Create Constant Text display appears.

```

Define Constants
.....
:                               Create Constant Text                               :
:                                                                           :
: Type choices, press Enter.                                             :
:                                                                           :
: Position across . .  4.0      *INCH      Value                               :
: Position down . . .  10.8     *INCH      Value                               :
: Constant value . .  Customer Copy                                             :
:                                                                           :
:                                                                           :
: Font . . . . . *PRTDEF          *PRTDEF, Value F4 for list           :
: Point size . . . *CALC, Value                                           :
: Bold . . . . . 1=Yes                                                    :
: Italic . . . . . 1=Yes                                                  :
: Rotation . . . . . *DEFAULT      *DEFAULT, 0, 90, 180, 270           :
: Color . . . . . *PRTDEF          *PRTDEF, Value F4 for list           :
:                                                                           :
: F4=Prompt  F12=Cancel  F22=Set Units                                       :
:                                                                           :
:                                                                           :
:.....

```

Figure 80. Create Constant Text Display

Specify the text Customer Copy and position it 10.8 inches down, and 4 inches across the page. Again, you could specify the positioning by row and column.

Position the cursor in the Font field, and press F4. The font database appears. For this example, select a Helvetica, 10-point, bold font as shown in the following display.

```

Define Constants
.....
:                               Select a Font                               :
:                               :                                           :
:                               :                                           :
: Font family . . . . . HELVETICA      Name, Generic*, *ALL      :
: Point size . . . . . *ALL           Value, *ALL           :
: Bold . . . . .                      1=Yes, 0=No           :
: Italic . . . . .                    1=Yes, 0=No           :
:                               :                                           :
: Type Options, press Enter.         :
: 1=Select 5=Details                 :
:                               :                                           :
: Opt  Font family      Size  Style   :
:                               :                                           :
: 1  HELVETICA          10  Bold      :
:   HELVETICA          10  Bold-Italic :
:   HELVETICA          11  Normal     :
:   HELVETICA          11  Italic     :
:   HELVETICA          11  Bold      :
:   HELVETICA          11  Bold-Italic :
:                               :                                           :
:                               More... :
: F5=Refresh  F12=Cancel             :
:                               :
:                               :
:.....

```

Figure 81. Select a Font Display

Select a Font is a pop-up display on which you specify a font family. You can type point size and bold and italic selections. Or you can select the font you want from the list of fonts in the bottom half of the display. In addition, you can use option 5 to display the details of a specific font.

```

Define Constants
.....
:                               Create Constant Text                          :
:                               :                                           :
: Type choices, press Enter.         :
:                               :                                           :
: Position across . . 4.0  *INCH      Value :
: Position down . . . 10.8 *INCH     Value :
: Constant value . . Customer Copy    :
:                               :                                           :
:                               :                                           :
:                               :                                           :
: Font . . . . . HELVETICA      *PRTDEF, Value F4 for list :
: Point size . . . 10          *CALC, Value           :
: Bold . . . . . 1            1=Yes                   :
: Italic . . . . .           1=Yes                   :
: Rotation . . . . . *DEFAULT  *DEFAULT, 0, 90, 180, 270 :
: Color . . . . . *PRTDEF     *PRTDEF, Value F4 for list :
:                               :                                           :
: F4=Prompt  F12=Cancel  F22=Set Units :
:                               :
:                               :
:.....

```

Figure 82. Create Constant Text Display

Press Enter. The Define Constants display reappears.

```

Define Constants
Print Definition . . . : SUPER          Page Format . . . . . : *DEFAULT
Library . . . . . : APUDATA      Copy . . . . . : *ORIGINAL

Type options, press Enter.
1=Create 2=Change 3=Copy 4=Delete
  Constant Position Position Unit of

Opt type      across  down   measure Constant value
1 *TEXT      4.0    10.8  *INCH  Customer Copy

F3=Exit F5=Refresh F12=Cancel

```

Figure 83. Define Constants Display

Press F3. The Define Overlays display appears.

```

Define Overlays
Print Definition . . . : SUPER          Page Format . . . . . : *DEFAULT
Library . . . . . : APUDATA      Copy . . . . . : *ORIGINAL

Type options, press Enter.
1=Create 2=Change 3=Copy 4=Delete

  Position Position Unit of
Opt across  down   measure Overlay
1 0         0     *INCH  INVALL

F3=Exit F5=Refresh F12=Cancel

```

Figure 84. Define Overlays Display

On the Define Overlays display, select the Super Sun Seeds invoice overlay (INVALL) to be printed on this copy. A sample of the INVALL overlay is shown below:


400 CFC Parkway Fayetteville, NJ 08300		 Super Sun Seeds A Growth Company		Option: 333-499-2567 Fax: 333-413-8784	
-- Sold To --				-- Ship To --	
Customer Number:	Invoice Number:	Invoice Date:	Payment Date:		
Ship Via:	Shipped Date:	Terms:	Salesman:		
This invoice overlay designed using IBM AFP Utilities/400				Total Due	
Return this tear-off strip with your payment.			Make Checks Payable to: Super Sun Seeds		
Payment is due by:			Amount Due is:		

Figure 85. Sample INVALL Overlay

At this point, the composition of the customer copy has been defined, and the Work with Copies display appears again.

```

Work with Copies
Print Definition . . . : SUPER          Page Format . . . . . : *DEFAULT
Library . . . . . : APUDATA

Type options, press Enter.
 1=Create  2=Change  3=Copy  4=Delete  7=Rename
10=Define

Opt  Name      Text
 *ORIGINAL  Customer Copy

Bottom

F3=Exit  F5=Refresh  F12=Cancel

```

Figure 86. Work with Copies Display

Select option 3 (copy) to replicate the contents of the customer copy to two additional copies: the packing list and the file copy.

Select option 10 to define these copies, and change the constant text printed at the bottom to Packing List and File Copy.

```

Work with Copies

Print Definition . . . : SUPER          Page Format . . . . . : *DEFAULT
Library . . . . . : APUDATA

Type options, press Enter.
1=Create 2=Change 3=Copy 4=Delete 7=Rename
10=Define

Opt Name      Text
*ORIGINAL    Customer Copy
FILE         File Copy
PACKING      Packing List

Bottom

F3=Exit  F5=Refresh  F12=Cancel

```

Figure 87. Work with Copies Display

In addition, for the packing list, suppress the printing of the prices. This is done on the Define Field Mapping display shown below.

```

Define Field Mapping

Spooled file . . . . . : INVPRE          Page/Line . . . . . : 1/17
Control . . . . . :                      Columns . . . . . : 1 - 78
*.....1.....+.....2.....+.....3.....+.....4.....+.....5.....+.....6.....+.....7.....+...

          100          31300          3/03/96          4/03/96
          BEST WAY          3/03/96  NET 30          YOUR PRINTER RE

          1 CT  00000300  HIGH ALTITUDE WATERMELON          100          100
          1 PK  01100517  SPARTAN SEEDS          100          100
          9 PK  04569870  NORTHERN LITE BLUE SPRUCE          100          900
          12 BX 11005004  BUSH GREEN SEEDS          100          1200
          12 CT 11005011  LASSO RED SEEDS          100          1200
          26 PK 11005018  EARLY BANTAM SEEDS          100          2600
          5 BX  11057893  AFRICAN DAISY, SEEDS          100          500
          1 PK  15975365  HEAVY OAK          100          100
          33 BX 32746510  HOPS BREWING LIGHT          100          3300

          More...
F3=Exit          F11=Hide mapping  F12=Cancel          F14=Start field
F15=End field    F16=Delete range  F20=Right

```

Figure 88. Define Price Suppression for Packing List

To suppress the price information on the packing list copy (PACKING), select option 10 to define the packing list copy and bring up the sample spooled file with the field mapping function. Use F14 and F15 to mark the first field in a column to be suppressed. In this example, the price and extension fields are suppressed from printing.


```

                                Define a Print Definition

Print Definition . . . : SUPER2
Library . . . . . : APUDATA

Type options, press Enter.
1=Select

Opt  Function

        Select a sample spooled file
        Set print definition attributes
1    Define selection fields for page formats
1    Define selection rules for page formats

F3=Exit  F12=Cancel

```

Figure 90. Specifying Field and Rules Selection

Now, having specified on creation that it has multiple page formats, PAGE1 and PAGEN, start the SUPER2 print definition. Because this print definition has multiple page formats, you are prompted to define selection fields and selection rules.

```

                                Define Selection Fields

Spooled file . . . . . : INVPRE          Page/Line . . . . . : 1/49
Control . . . . .      :                  Columns . . . . . : 1 - 78
*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...

                                                $90,691.8

                                4/03/96          $90,691.8

        IMPROVED PRINTING CORP
        PERFORMANCE BOULEVARD
        PRINTERSVILLE
        CO 45789-2637

                                                Page      1
                                                More...

F3=Exit          F11=Hide fields    F12=Cancel
F14=Start field  F15=End field
Mark end of field, press F15 or press F12 to cancel    F20=Right

```

Figure 91. Defining Selection Fields

The sample spooled file, INVPRE, is displayed. Locate the page number field at the bottom on each invoice, and use F14 and F15 to mark it as a field.

```

Define Selection Fields
Spooled file . . . . . : INVPRE          Page/Line . . . . . : 1/49
Control . . . . .      :                Columns . . . . . : 1 - 78
*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...

:                Define Selection Field                :
:                                                        :
: Type choices, press Enter.                            :
:                                                        :
: Starting Row / Column : 63 / 73                    :
IMPROVED PRIN : Length . . . . .      3          Value :
PERFORMANCE B : Name . . . . .      F.063.073      Name :
PRINTERSVILLE :
CO 45789-263 : F12=Cancel
:
:
:.....
Page 1
More...

F3=Exit          F11=Hide fields    F12=Cancel
F14=Start field  F15=End field                F20=Right
Mark end of field, press F15 or press F12 to cancel

```

Figure 92. Selection Field Pop-up

A pop-up display shows the selected field, and gives it a default name (“F.063.073”).

```

Define Selection Fields
Spooled file . . . . . : INVPRE          Page/Line . . . . . : 1/49
Control . . . . .      :                Columns . . . . . : 1 - 78
*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...

:                Edit Selection Field                :
:                                                        :
: Type choices, press Enter.                            :
:                                                        :
: Starting Row / Column : 63 / 73                    :
IMPROVED PRIN : Length . . . . .      3          Value :
PERFORMANCE B : Name . . . . .      PAGENBR       Name :
PRINTERSVILLE :
CO 45789-263 : F12=Cancel  F16=Delete
:
:
:.....
Page 000
More...

F3=Exit          F11=Hide fields    F12=Cancel
F14=Start field  F15=End field                F20=Right

```

Figure 93. Selection Field Naming

Change the default field name to “PAGENBR” in order for this print definition to be more readable. Note that the number field has changed to 000 indicating that it is a mapped field.

```

.....
:                                     Define a Rule                                     :
:                                                                                       :
: Type choices, press Enter.                                                           :
:                                                                                       :
: Rule number . . . . . : 10                                                         :
:                                                                                       :
: Condition 1 . . . . . IF                                                           IF, blank :
: Field . . . . . PAGENBR                                                           Name      F4 for list :
: Test . . . . . *EQ                                                                *EQ, *NE, *GT...    :
: Field or 'Characters' . ' 1'                                                       Value     F4 for list :
: Action . . . . . *USE                                                                *USE, *OMIT, *GOTO  :
: Page format or rule . . PAGE1                                                       Value     F4 for list :
:                                                                                       :
: Condition 2 . . . . . AND, OR                                                       :
: Field . . . . . Name      F4 for list :
: Test . . . . . *EQ, *NE, *GT...    :
: Field or 'Characters' . Value     F4 for list :
: Action . . . . . *USE, *OMIT, *GOTO  :
: Page format or rule . . Value     F4 for list :
:                                                                                       :
:                                                                                       More... :
: F4=Prompt  F12=Cancel                                                             :
:                                                                                       :
:                                                                                       :
.....

```

Figure 94. Define Selection Rules for Page 1

Using the Define a Rule display, enter the logic for identifying Page 1. If the value of PAGENBR is 1, then APU will select page format PAGE1.

```

.....
:                                     Define a Rule                                     :
:                                                                                       :
: Type choices, press Enter.                                                           :
:                                                                                       :
: Rule number . . . . . : 20                                                         :
:                                                                                       :
: Condition 1 . . . . . IF                                                           IF, blank :
: Field . . . . . PAGE#                                                           Name      F4 for list :
: Test . . . . . *NE                                                                *EQ, *NE, *GT...    :
: Field or 'Characters' . ' 1'                                                       Value     F4 for list :
: Action . . . . . *USE                                                                *USE, *OMIT, *GOTO  :
: Page format or rule . . PAGEN                                                       Value     F4 for list :
:                                                                                       :
: Condition 2 . . . . . AND, OR                                                       :
: Field . . . . . Name      F4 for list :
: Test . . . . . *EQ, *NE, *GT...    :
: Field or 'Characters' . Value     F4 for list :
: Action . . . . . *USE, *OMIT, *GOTO  :
: Page format or rule . . Value     F4 for list :
:                                                                                       :
:                                                                                       More... :
: F4=Prompt  F12=Cancel                                                             :
:                                                                                       :
:                                                                                       :
.....

```

Figure 95. Define Selection Rules for All Other Pages

Enter the logic for identifying pages greater than page one. If the value of PAGENBR is not 1, then APU will select page format PAGEN.

```

Define Selection Rules

Print definition . . . : SUPER2
Library . . . . . : APUDATA

Type options, press Enter.
1=Create 2=Modify 4=Delete

Opt Rule Cond Field Test Field or 'Characters' Act. Page format
or Rule

10 IF PAGE# *EQ ' 1' *USE PAGE1
20 IF PAGE# *NE ' 1' *USE PAGEN

F3=Exit F5=Refresh F12=Cancel F22=Renumbr
Rule 20 changed

Bottom

```

Figure 96. Summary of Selection Rules

APU summarizes the current selection rules for the SUPER2 print definition.

After you have identified the page formats, you can perform the same definition process to define copies. The major difference in defining these two page formats is that they have different front overlays.

Printing with APU

Once you have completed your APU application design, you are ready to put it into production. For each APU application, you can define:

- How to select the target spooled files
- Which APU print definition to use
- What additional functions, through user exits, might be required
- What output files to create
- Where to route the output files
- How to handle input and work files during and after processing

With this information, the APU production monitor can select the target spooled file when it reaches the output queue, retrieve the APU print definition, invoke the APU print engine to transform the input spooled records into AFP, and manage the disposition of files.

The APU production monitor can be invoked in three ways:

1. Manually Associating a Print Definition with a Spooled File. For manual printing, you just “apply” your print definition to a specific spooled file. Refer to 175.
2. Using the Apply Print Definition Command, which enables you to imbed an APU printing step within existing application procedures. Refer to 178.
3. Automatic Printing with APU Monitor to automatically identify the sample spooled file and run the conversion programs. This method is described in 178.

Manually Associating a Print Definition with a Spooled File

This process works as follows. Using your print definition, SUNSD1 or INVOICE, select option 2 from the APU Main Menu. The Work with Spooled Files panel appears.

```

                                Select a Sample Spooled File
Output Queue . . . . . QYPUOUTQ      Name, *ALL   F4 for list
Library . . . . . QAPU              Name, *LIBL
User . . . . . *ALL                 Name, *CURRENT, *ALL
Type choices, press Enter.
1=Select  5=Display

Opt  File      Nbr  User      User Data  Queue      Sts  Total
     INVPRE    1   PERELMAN  QYPUOUTQ  RDY       7
     INVSCS    2   PERELMAN  QYPUOUTQ  HLD       6

                                Bottom

F4=Prompt  F5=Refresh  F12=Cancel

```

Figure 97. Select a Spooled File Panel

You can display selected output queues and spooled files with this option, and then apply a print definition to the spooled file for example, INVSCS.

Panel 1: Apply Print Definition

```

                                Apply Print Definition (APYPRDEF)
Type choices, press Enter.
SCS Spooled File . . . . . > INVSCS      Name
Job name . . . . . > QPADEV0016      Name, *
User . . . . . > JOHN                Name
Number . . . . . > 098677            000000-999999
SCS Spooled file number . . . . . > 10  1-9999, *ONLY, *LAST
Print Definition . . . . . > INVOICE   Name, *NONE, *SPOOLFILE
Library name . . . . . > QAPU         Name, *PRTDEFLIB, *LIBL
Run option . . . . . > *NORMAL        *NORMAL, *NOCOPY, *REPRINT
Post processing SUCCESS:
SCS Spooled File . . . . . > *NONE     *HOLD, *NONE, *DELETE, *OUTQ
Output queue . . . . .
Library name . . . . .
Post processing FAILURE:
SCS Spooled File . . . . . > *NONE     *HOLD, *NONE, *DELETE, *OUTQ
Output queue . . . . .
Library name . . . . .
                                More...
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

Figure 98. Apply Print Definition Panel

APU will move the input spooled file to the output queue defined in the Success or Failure fields, depending on the result, and then places the file in one of the four status conditions shown above. *HOLD is the default for both success and failure.

You should specify values for these fields:

SCS Spooled File

Name of the input spooled (SCS) file to which a print definition is to be applied.

Print Definition

Name of the print definition to be applied to the spooled file.

Post processing Success and Failure

Specify what you want to happen to the input spooled file on success or failure.

Press Page Down to access the next panel.

Note: INVSCS can also be used with the SUPER and SUPER2 print definition samples that are in the QAPU library. INVPRE can be used with the AMASTER print definition, that is also in the QAPU library.

Panel 2: Apply Print Definition

```

Apply Print Definition (APYPRTDEF)
Type choices, press Enter.
User exit BEFORE:
  Program . . . . . *NONE      Name, *NONE
  Library Name . . . . .      Name, *LIBL
  User parameter . . . . .
User exit MIDDLE:
  Program . . . . . *NONE      Name, *NONE
  Library Name . . . . .      Name, *LIBL
  User parameter . . . . .
Device Name . . . . . *JOB      Name, *JOB
Output queue . . . . . *SPOOLFILE  Name, *DEV, *SPOOLFILE
  Library Name . . . . .      Name, *LIBL
Spooled file name . . . . . *SPOOLFILE  Name, *SPOOLFILE, *PRTDEF
User data . . . . . *SPOOLFILE  Character value...
Form type . . . . . *SPOOLFILE  Character value...
Hold spooled file . . . . . *NO      *NO, *YES
Save spooled file . . . . . *NO      *NO, *YES, *SPOOLFILE
More...
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

Figure 99. Apply Print Definition Panel

Make the entries you want. Press F1 for descriptions of the fields. Here is a description of some of the key fields:

User exit BEFORE

A program you want run before processing.

User exit MIDDLE

Name of a program you want to be run after the input spooled file has been copied to the input spool database.

Device name

Name of the printer to which the output is sent.

Output Queue

Name of the queue on which to put the output.

Spooled file name

Name of the output spooled file.

User data

A user-defined parameter in the form of a character string to be placed on the output, such as an e-mail address.

Hold or Save

Specify what you want done with the spooled file after it is processed.

Panel 3: Apply Print Definition

```
Apply Print Definition (APYPRTDEF)
Type choices, press Enter.
Output bin . . . . . *SPOOLFILE 1-65536, *SPOOLFILE, *DEV
User exit AFTER:
  Program . . . . . *NONE      Name, *NONE
  Library Name . . . . .      Name, *LIBL
  User parameter . . . . .

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys
```

Figure 100. Apply Print Definition Panel

Fill in the information you want. Press F1 for descriptions of the fields. Key field entries here are:

Output bin

Where the output is to be placed.

User exit AFTER

Name of a program you want to be run after the AFPDS output spooled file has been created.

Using the Apply Print Definition Command

One method of starting APU is to use the Apply Print Definition (APYPRTDEF) command. This is the same command that is called when you manually apply a print definition to a spooled file, working from the Work with Spooled Files option that is described in “Manually Associating a Print Definition with a Spooled File” on page 175.

To use the apply print definition command, place the command within existing application procedures, such as a CL program. An example of the command is shown below:

```
APYPRTDEF FILE(INVSCS) JOB(098677/JOHN/QPADEV0016) SPLNBR(10)+
PRTDEF(QAPU/INVOICE) SUCCESS(*NONE) FAILURE(*NONE) DEV(PRT3130)
```

This command selects the INVSCS spooled file from the currently running job (either batch or interactive) and starts APU printing, using the INVOICE print definition. The new AFP output is sent to the output queue that is associated with the PRT3130 printer device.

Note: The APYPRTDEF command can also be used to run in batch.

Automatic Printing with APU Monitor

This section describes the third method of printing APU print definitions.

To give you an understanding of the operation of the APU Monitor, the following topics are described here:

- “Understanding How the APU Monitor Works” on page 179
- “An Example of APU Monitor Processing” on page 179

Understanding How the APU Monitor Works

In the APU Monitor, you can specify values for the parameters that govern APU Monitor processing. These parameters are grouped together and called an *Action*.

There are three Action groups in the new APU Monitor. The Monitor processes these actions in the following sequence:

- Selection for input spooled file
- Action for input spooled file
- Action for output spooled file

Note: If an action group contains more than one action, the actions are processed in the order in which they are defined.

Figure 101 illustrates the sequence in which the APU Monitor processes the action groups.

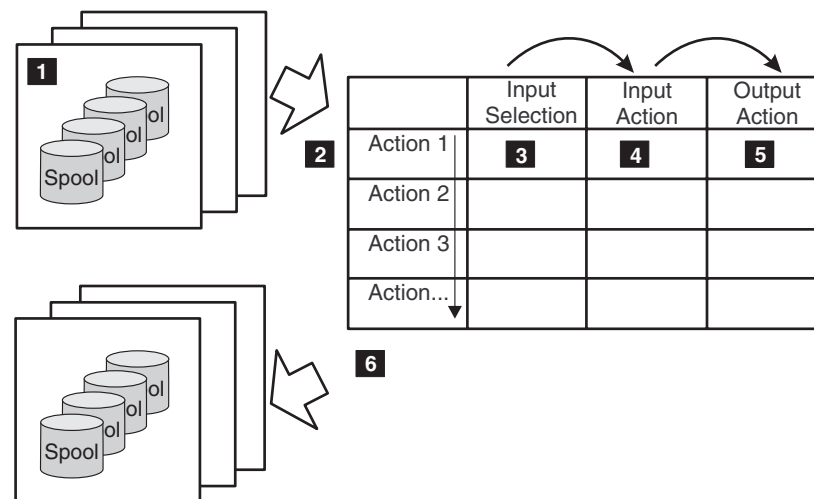


Figure 101. APU Monitor Processing Sequence

The diagram illustrates these steps:

1. The monitor is invoked each time a spooled file arrives in a monitored output queue or if the spooled file status from a spool in a monitored queue changes to *RDY. Spooled files with other status conditions are not processed.
2. The monitor checks the input selection from each action rule sequentially.
3. As soon as a spooled file matches the action input selection, the input and output action are performed. Subsequent actions are ignored.
4. The input action is applied after the selection matches a spooled file. The action performed depends on whether or not APU is able to complete the job successfully.
5. You can define up to 16 output actions. This allows you, for example, to use several different APU print definitions for the same spooled file.
6. One or more spooled files are placed into one or more output queues.

An Example of APU Monitor Processing

This section describes an example of how the concepts of the APU Monitor can be implemented in a customer environment.

A Customer Environment

Assume that a customer wants to set up the following environment:

- Three different output types are needed, with each output type going to a different output queue (OUTQn).
- Two printers are available (PRT0n). The monitor is to be set up with these requirements:
 - System output (QSYSPRT) must not use an APU print definition.
 - All jobs in OUTQ1 must be sent to PRT01
 - All jobs in OUTQ2 and OUTQ3 must be sent to PRT02
 - Application jobs APP01 and APP02 must be sent with a print definition **SAMPLE** applied
 - The application's original spooled files must be placed in the OUTQ called **SAVE**.
 - The original QSYSPRT spooled files must be deleted.

The Figure 102 diagrams these customer requirements. The numbers in the figure are used to identify the sequence of notes provided following this diagram.

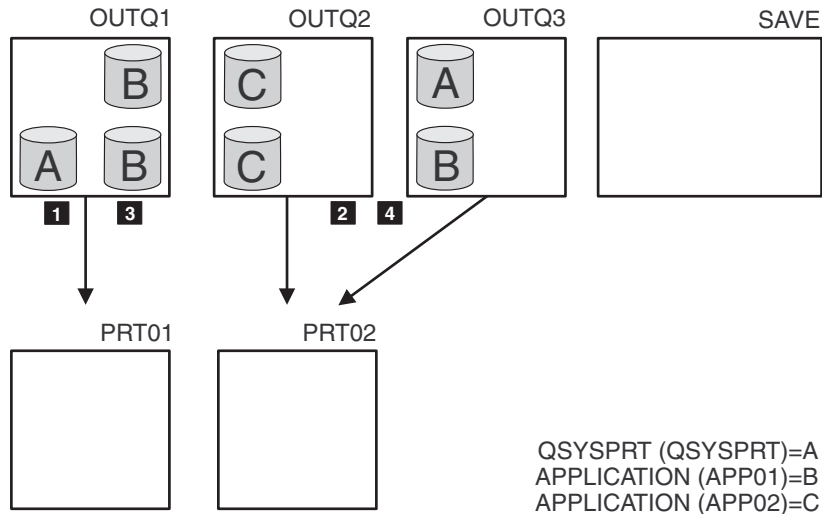


Figure 102. APU Monitor Before Processing

Actions:

1. All QSYSPRT spooled files from the OUTQ1 are moved to OUTQ PRT01.
2. All QSYSPRT spooled files from all other OUTQs are moved to OUTQ PRT02.
3. A print definition is to be applied to all application spooled files coming into OUTQ1. A new APU spooled file (the result of the APU processing) is to be placed in the output queue PRT01. The original SCS spooled file is moved into OUTQ SAVE.
4. A print definition is to be applied to all application spooled files coming into all other OUTQs. A new APU spooled file (the result of the APU processing) is to be placed in the output queue PRT02 for each original spooled file. The original SCS spooled file is to be moved into OUTQ SAVE.

Implementing the Customer Requirements on the APU Monitor

In the example, we can define two groups of spooled files: the application spooled files and the QSYSPRT spooled files. Only the application spooled files need an

APU print definition. In this case, we want to define actions for the application spooled files first and then the action for the QSYSPRT spooled files. So we can say that all spooled files that are not eligible for APU are moved following the QSYSPRT spooled file actions.

Figure 103 shows which parameters must be defined for each action in the order of the action. The monitor takes the **Input selection** parameters of the first action to identify if the spool and selection match. If the input selection parameters do not match the spooled file, the monitor takes the next action. As soon as the input selection parameters match the spooled file, all action sequences such as **Input action** and **Output actions** proceed.

The numbers in the figure indicate the actions that correspond with Figure 103.

Action	Input selection	Input action	Output action
1. Action for spool 3	File = APP* OUTQ = Outq1	5 Success = *outq OUTQ = SAVE Failure = *hold	Prtdef = Sample OUTQ = PRT01
2. Action for spool 4	File = APP* OUTQ = *all	Success = *outq OUTQ = SAVE Failure = *hold	Prtdef = Sample OUTQ = PRT02
3. Action for spool 1	File = *all OUTQ = Outq1	Success = *outq OUTQ = PRT01 Failure = *hold	Prtdef = *none
4. Action for spool 2	File = *all OUTQ = *all	Success = *outq OUTQ = PRT02 Failure = *hold	Prtdef = *none

Figure 103. APU Monitor - Action Example

Actions:

1. Action for the application spooled files in OUTQ1
2. Action for all other application spooled files in all monitored OUTQs
3. Action for all other spooled files in OUTQ1
4. Action for all other spooled files in all other OUTQs
5. Action to take upon success or failure

Many other options are possible for each action. You can decide, for example, to delete the original spooled files after processing or hold the spooled files.

Condition of the Output Queue after Processing

Figure 104 on page 182 shows the two QSYSPRT spooled files (A) were sent to print without being saved, that all the original application spooled files are in output queue **SAVE**, and that the new AFPDS spooled files (output from APU processing) are placed in the output queues PRT01 and PRT02, depending on where the original was.

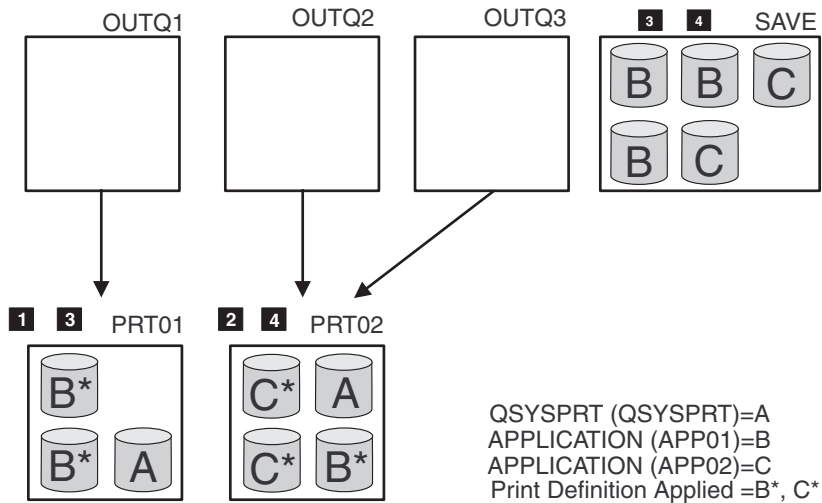


Figure 104. APU Monitor Example After Processing

Results:

1. The QSYSPRT spooled file from OUTQ1 is in the output queue PRT01.
2. All QSYSPRT spooled files from the other OUTQs are in the output queue PTR02.
3. The original application SCS spooled files from OUTQ1 are in the output queue SAVE. New AFPDS spooled files (B* and C*) have been placed in the output queue PRT01. This new spooled file is the result from APU after applying the print definition.
4. All other original application SCS spooled files from all other OUTQs are placed in the output queue SAVE. New AFPDS spooled files have been placed in the output queue PRT02. These new spooled files result from APU after applying the print definition.

Note: If the processing for one spooled file fails, the original spooled file stays in the output queue in status *HOLD as specified on the FAILURE parameter.

Chapter 16. Using Page Printer Formatting Aid

Page Printer Formatting aid (PPFA) is an IBM licensed program that helps users create their own line data (including record format line data and XML data) formatting resources, called form definitions and page definitions. For information about recent enhancements to PPFA, see “Record Format and PPFA” on page 194. Page definitions and form definitions are used to format line data independent of the application that generates it. Specifically:

Page definitions

Specify how the line data is formatted, including fonts and positioning.

Form definitions

Specify characteristics of the physical page, such as duplex printing and overlays to use.

In most cases you should use Infoprint Designer to create page definitions and form definitions. Infoprint Designer provides both a graphical interface and a programming interface, so you can use whichever you are more comfortable with. Currently, Infoprint Designer does not support 2D bar codes or record format line data, so to use those functions you must use PPFA.

Using a form definition or a page definition created by PPFA is a three step process:

1. Write a set of PPFA commands that define how to position the data or handle the physical sheets, that is, you need to write code to create a page definition or form definition. For information, refer to *Page Printer Formatting Aid: User's Guide*.
2. Run PPFA to build the specified page definition or form definition and store the output as resources in a library. See “Creating Page Definitions and Form Definitions with the PPFA Compiler.”

Note: If you already have a compiled form definition on your workstation, you can use the iSeries Navigator import function to import it to your iSeries as a form definition resource. See “Importing Resources to the OS/400” on page 273 for instructions.

3. Submit the print file, specifying the page definition and form definition. See “Using Page Definitions and Form Definitions to Format OS/400 Output” on page 189.

This chapter explains how to complete the last two steps. It also shows how you can use a page definition and form definition with the Super Sun Seeds case study (see “Super Sun Seeds Case Study” on page 191) and describes record format line data (see “Record Format and PPFA” on page 194).

Creating Page Definitions and Form Definitions with the PPFA Compiler

This section describes how to compile PPFA source code to create page definitions and form definitions. It assumes you already have the source code on your system. Although you can use them independently, we will create a form definition and page definition that work together.

Figure 105 illustrates the process of creating page definitions and form definitions using PPFA.

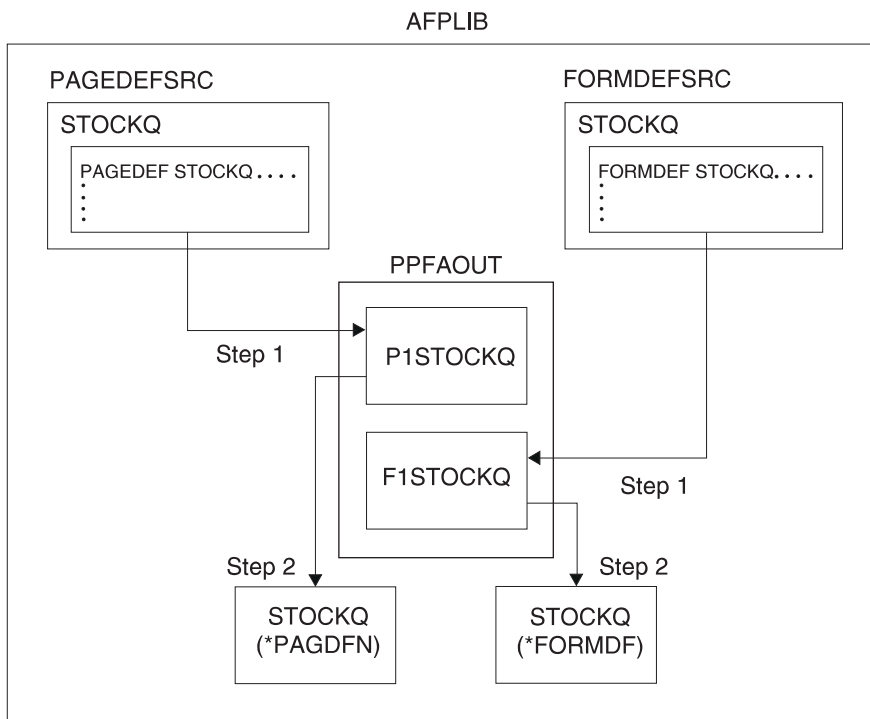


Figure 105. Creating Page Definitions and Form Definitions Using PPFA

This is the data we will format with the page definition and form definition. These records have been created by a sample invoice application:

1John Smith
690 Bean Hill Road
Louisville, KY 50301

193-45-8901

Don't forget to take advantage of an abundance of online material on our new World Wide Web homepage at <http://www.stocks-r-us.com>.

General Parts Inc.	(GPIX)	67.00	68.25	67.75
Rasterpunkt Inc.	(RPGMBH)	14.00	15.00	15.00
Ecocycle Inc.	(ECO)	5.25	5.50	5.25

1Linda Noll
14 Hillcrest Drive
Houston, TX 60443

677-12-7468

Don't forget we offer a preapproved Check payment plan. Call our toll-free number 1-800-STOCKSR to join.

Management Plus Co. (MGTPLS)	25.50	28.75	26.25
------------------------------	-------	-------	-------

We will use a page definition and form definition to format the output as shown in Figure 112 on page 190 and Figure 113 on page 191. In this section, we compile the page definition then the form definition.

Creating a Page Definition Using the PPFA Compiler

This section shows how to create a page definition on the OS/400 using the PPFA compiler and the PPFA page definition source code shown in Figure 108 on page 187.

You can create a page definition using the PPFA compiler in two ways:

- Using the command line interface
- Using the Convert PPFA Source (CVTPPFASRC) display

Both options are shown.

Before you begin:

- Make sure that the qppfa library is in your library list. You can add the library to your library list by entering `addlib qppfa`.
- Ensure that the source code for the page definition exists in the AFPLIB/page definition SRC (STOCKQ) member.

Steps to create a page definition:

Creating a page definition with PPFA takes two steps:

1. Convert the PPFA source code to a page definition output database file member. You can do this in either of two ways:

- Enter:
`cvtppfasrc file(afplib/pagdefsrc) mbr(stockq) pagdfnfile(afplib/ppfaout)`
- Enter `cvtppfasrc` and press F4 to open this display:
Specify the values you want and press Enter.

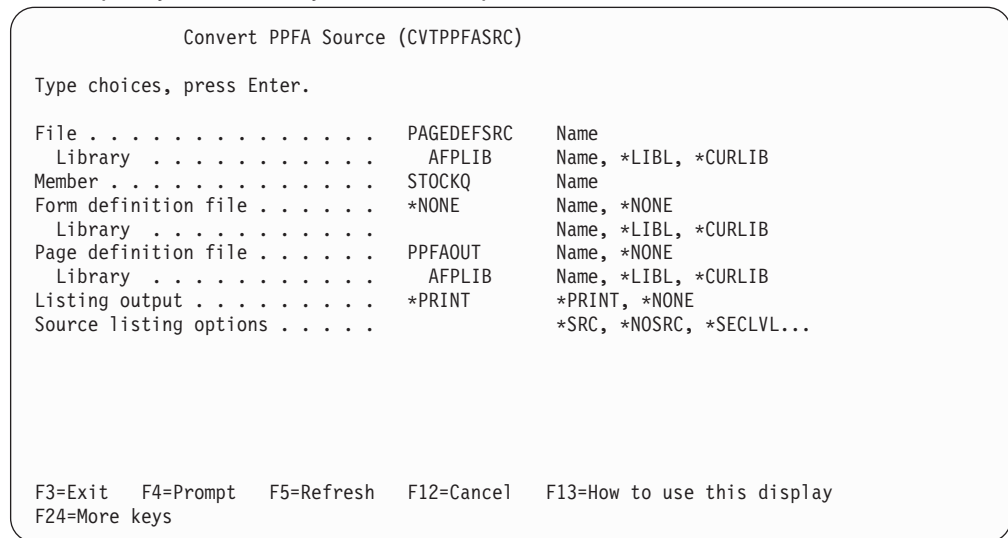


Figure 106. Convert PPFA Source Display

Note: In creating the output member, the PPFA compiler adds P1 to the front of the name you specify on the page definition statement in the PPFA source code. Adding these two characters lets you put a page definition and a form definition in the same OS/400 file under different member names.

2. Create the OS/400 page definition resource, which has a type of *PAGDFN. You can do this in either of two ways:

- Enter:
`crtpagdfn pagdfn(afplib/stockq) file(afplib/ppfaout) mbr(p1stockq)`
- Enter `crtpagdfn` and press F4 to open this display:

```

                                Create Page Definition (CRTPAGDFN)

Type choices, press Enter.

Page definition . . . . . > STOCKQ      Name
Library . . . . . > AFPLIB           Name, *CURLIB
File . . . . . PPFAOUT              Name
Library . . . . . AFPLIB           Name, *LIBL, *CURLIB
Member . . . . . P1STOCKQ          Name, *PAGDFN
Text 'description' . . . . . *MBRTXT

                                                                    Bottom
F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel
F13=How to use this display  F24=More keys

```

Figure 107. Create Page Definition (CRTPAGDFN) Display

Specify the values you want and press Enter.

The page definition source code is given in Figure 108 on page 187.


```

5722WDS V5R2M0                      SEU SOURCE LISTING
SOURCE FILE . . . . . AFPLIB/PAGEDEFSRC
MEMBER . . . . . STOCKQ
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ..
100 PAGEDEF STOCKQ                      /* CVTPPFASRC will create P1STOCKQ */
200 REPLACE YES                          /* PPFA should replace existing copy */
300 LINEONE 0.5 IN 1.0 IN;               /* Position of topmost line */
400 FONT BIG PR10;                       /* Prestige Roman 10 Char/Inch */
500 FONT NORM PR12;                      /* Prestige Roman Normal 12 Char/Inch */
600 FONT ITAL PI12;                      /* Prestige Italic Normal 12 Char/Inch */
700 SETUNITS LINESP 6 LPI;               /* 6 lines per inch */
800
900 /* Customer Address Block - 4 input lines */
1000 PRINTLINE
1100 CHANNEL 1                            /* When an input line has a first col. */
1200                                     /* (CC) '1' value, this PRINTLINE gets */
1300                                     /* control - meaning: top of a new page */
1400 POSITION 2 IN 1 IN                    /* position across 2 inches down 1 inch */
1500 FONT BIG                             /* Use the font nicknamed BIG */
1600 REPEAT 4;                            /* Do this for 4 lines - subsequent */
1700                                     /* lines are positioned down the page */
1800                                     /* at 6 lines/inch (see SETUNITS above).*/
1900
2000 /* Soc. Sec. Number - 1 input line rotated/placed in upper right */
2100 PRINTLINE
2200 POSITION 7.8 IN 0.8 IN                /* position across 7.75 in down 0.75 in */
2300 FONT NORM                             /* Use the font nicknamed NORM */
2400 DIRECTION DOWN;                      /* Print down the page (rotated 90 dgs) */
2500
2600 /* Customer-specific text - 2 input lines */
2700 PRINTLINE                             /*
2800 POSITION 1.25 IN 3.0 IN                /* position across 1.25 in down 3 in */
2900 FONT ITAL                             /* use the font nicknamed ITAL */
3000 REPEAT 2;                            /* Do this for two input lines */
3100
3200 /* Sales Quote - up to 30 lines of text */
3300 PRINTLINE
3400 POSITION 2.5 IN 4.25 IN                /* position across 2.5 in down 4.25 in */
3500 FONT NORM                             /* Use the font nicknamed NORM */
3600 REPEAT 30;                            /* Do this for up to 30 input lines */
                                     * * * * E N D O F S O U R C E * * * *

```

Figure 108. Page Definition Source Code

Creating a Form Definition using the PPFA Compiler

This section shows how to create a form definition on the OS/400 using the PPFA compiler and the PPFA form definition source code shown in Figure 108.

You can create a form definition with PPFA in two ways:

- Using the command line interface
- Using the Convert PPFA Source (CVTPPFASRC) display

Both options are shown.

Before you begin:

- Make sure that the library qppfa is in your library list. You can add the library to your library list by entering `addlib qppfa`.
- Ensure that the source code for the form definition is in member STOCKQ in the AFPLIB/form definition SRC source file.

Steps to create a form definition:

Creating a form definition with PPFA takes two steps:

1. Convert the PPFA source code to a form definition output database file member.
You can do this in either of two ways:

- Enter:

```
cvtppfasrc file(afplib/formdefsrc) mbr(stockq) formdffile(afplib/ppfaout)
```

- Enter cvtppfasrc and press F4 to open this display:

Specify the values you want and press Enter.

Convert PPFA Source (CVTPPFASRC)

Type choices, press Enter.

File	FORMDEFSRC	Name
Library	AFPLIB	Name, *LIBL, *CURLIB
Member	STOCKQ	Name
Form definition file	PPFAOUT	Name, *NONE
Library	AFPLIB	Name, *LIBL, *CURLIB
Page definition file	*NONE	Name, *NONE
Library		Name, *LIBL, *CURLIB
Listing output	*PRINT	*PRINT, *NONE
Source listing options		*SRC, *NOSRC, *SECLVL...

F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys

Figure 109. Convert PPFA Source (CVTPPFASRC) Display

Note: In creating the output member, the PPFA compiler adds F1 to the front of the name you specify on the form definition statement in the PPFA source code. Adding these two characters lets you put a page definition and a form definition in the same OS/400 file under different member names.

2. Create the OS/400 form definition object, which has a type of *FORMDF. You can do this in either of two ways:

- Enter:

```
crtformdf formdf(afplib/stockq) file(afplib/ppfaout) mbr(f1stockq)
```

- Enter: crtformdf and press F4 to open this display:

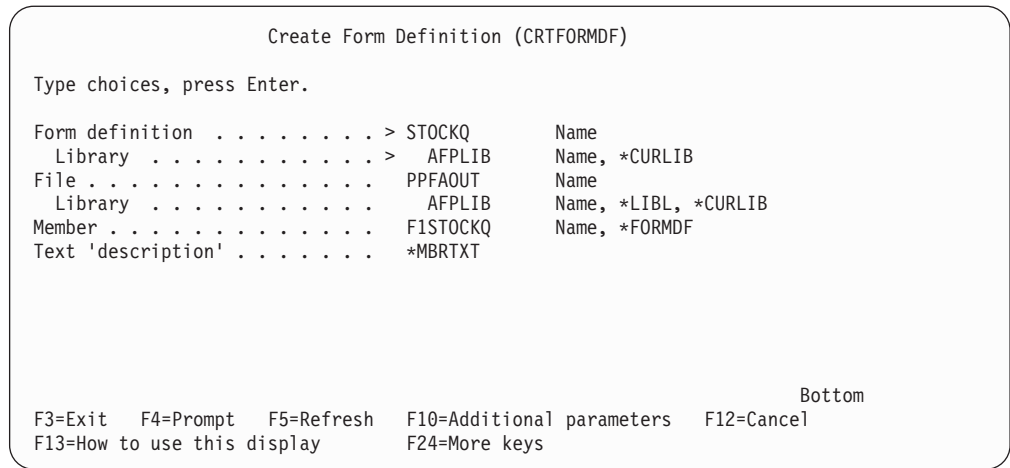


Figure 110. Create Form Definition (CRTFORMDF) Display

Specify the values you want and press Enter.

The form definition source code is given in Figure 111.

```

5722WDS V5R2M0                      SEU SOURCE LISTING
SOURCE FILE . . . . . AFPLIB/FORMDEFSRC
MEMBER . . . . . STOCKQ
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ..
100  FORMDEF STOCKQ                      /* CVTPPFASRC will create F1STOCKQ */
200  REPLACE YES;                        /* PPFA should replace existing copy */
300
400  OVERLAY QSHEET;                      /* 01QSHEET to be used - Quote Sheet */
500  OVERLAY ARCHIV;                      /* 01ARCHIV to be used - "Archive" lbl */
600  OVERLAY CUSTMR;                      /* 01CUSTMR to be used - "Customer" lbl */
700
800  /* For each page of application data, two pages will be printed: */
900  /* one labelled "Archive Copy" for archival and the other labelled*/
1000 /* "Customer Copy" to be mailed to the customer. */
1100 SUBGROUP COPIES 1 OVERLAY QSHEET ARCHIV; /* Archive Copy */
1200 SUBGROUP COPIES 1 OVERLAY QSHEET CUSTMR; /* Customer Copy */
                                     * * * * E N D O F S O U R C E * * * *

```

Figure 111. Form Definition Source Code

Using Page Definitions and Form Definitions to Format OS/400 Output

To specify a page definition and form definition to be used, specify them on the printer file Page definition (PAGDFN) and Form definition (FORMDF) parameters. The output file type must be *LINE, which is specified on the Device type (DEVTYPE) parameter. For example, to use the page definition and form definition above with the printer file PRTF, enter this command:

```
CHGPRTF FILE(PRTF) DEVTYPE(*LINE) PAGDFN(AFPLIB/P1STOCKQ) FORMDF(AFPLIB/F1STOCKQ)
```

These additional parameters might be necessary:

- Specify DUPLEX(*FORMDF) if your form definition specifies duplexing options.
- Specify DRAWER(*FORMDF) if your form definition specifies the paper source.
- Specify CVTLINDTA(*YES) if you are going to send the resulting output to an ASCII printer by Host Print Transform or if you are going to view the output with the AFP Viewer.

When you specify line data, the print lines are passed to the page definition and form definition objects in a slightly different format than SCS. Line data adds blank lines for vertical positioning and a couple of additional carriage control characters to each print record. This is transparent to the application program, but the developer of the page definition must be aware that space and skip-to lines will be passed to the page definition as blank lines. They can either be printed as blank lines (to maintain vertical positioning) or ignored, whichever is appropriate to a particular page layout.

The printed output, using the STOCKQ page definition and form definition looks like this:

John Smith 690 Bean Hill Road Louisville, KY 50301	193-45-8901	Customer Copy 193-45-8901																				
Don't forget to take advantage of an abundance of online material on our new World Wide Web homepage at http://www.stocks-r-us.com .																						
<p style="text-align: center;">Your personalized stock quote:</p> <table border="0"> <tr> <td>General Parts Inc. (GPIX)</td> <td>67.00</td> <td>68.25</td> <td>67.75</td> </tr> <tr> <td>Rasterpunkt Inc. (RPGMBH)</td> <td>14.00</td> <td>15.00</td> <td>15.00</td> </tr> <tr> <td>Ecocycle Inc. (ECO)</td> <td>5.25</td> <td>5.50</td> <td>5.25</td> </tr> </table> <p style="text-align: center;">STOCKS-R-US is the leader in convenient stock purchasing</p>	General Parts Inc. (GPIX)	67.00	68.25	67.75	Rasterpunkt Inc. (RPGMBH)	14.00	15.00	15.00	Ecocycle Inc. (ECO)	5.25	5.50	5.25	<p>ance of online material on our www.stocks-r-us.com.</p> <p>stock quote:</p> <table border="0"> <tr> <td>67.00</td> <td>68.25</td> <td>67.75</td> </tr> <tr> <td>14.00</td> <td>15.00</td> <td>15.00</td> </tr> <tr> <td>5.25</td> <td>5.50</td> <td>5.25</td> </tr> </table>	67.00	68.25	67.75	14.00	15.00	15.00	5.25	5.50	5.25
General Parts Inc. (GPIX)	67.00	68.25	67.75																			
Rasterpunkt Inc. (RPGMBH)	14.00	15.00	15.00																			
Ecocycle Inc. (ECO)	5.25	5.50	5.25																			
67.00	68.25	67.75																				
14.00	15.00	15.00																				
5.25	5.50	5.25																				
<p style="text-align: center;">STOCKS-R-US is the leader in convenient stock purchasing</p>																						

Figure 112. Formatted Output - Customer 1

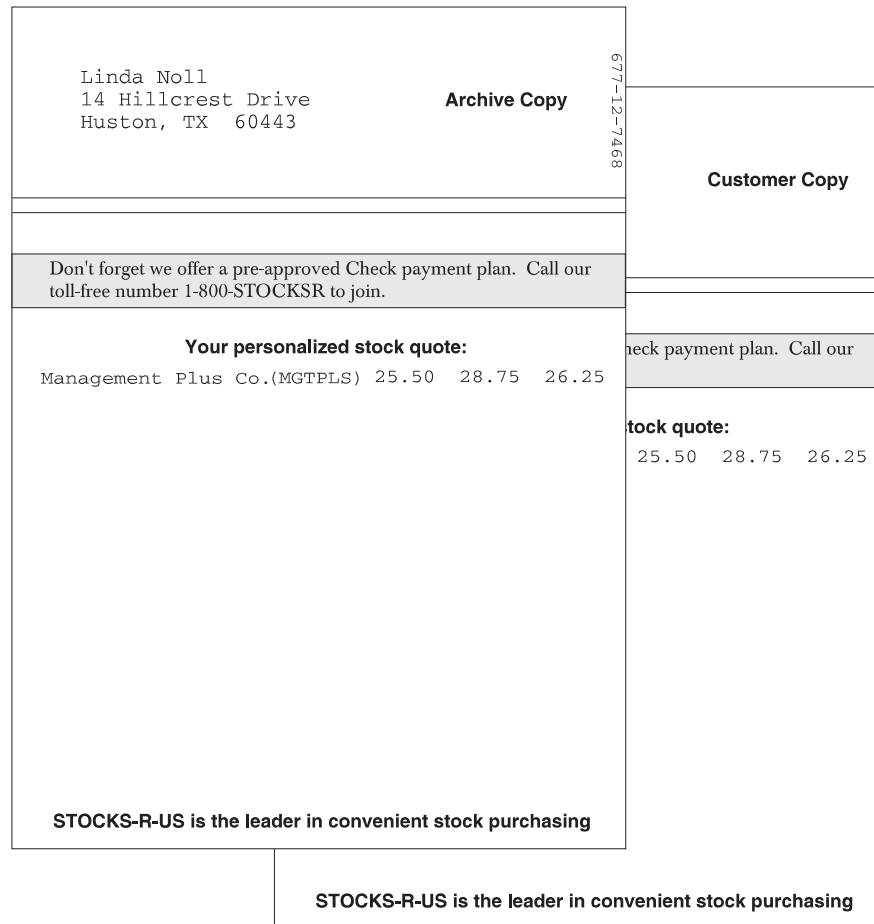


Figure 113. Formatted Output - Customer 2


Super Sun Seeds Case Study

In this section, we re-visit the Super Sun Seeds case study and show how you can use PPFA make it a fully electronic application. The invoice output, shown in Figure 114 on page 192 and Figure 115 on page 193 illustrates these capabilities:

- Dynamic page format switching based on application data content
- Constant Terms and Conditions overlay on the back sides of specified pages, without requiring the application to generate a page eject
- Producing multiple copies of specified pages with different purposes (for example, a packing slip, customer copy, and file copy) without requiring the application to generate each variation
- Suppressing price data from the packing slip copy

The page definition and form definition in Appendix C, “Source Code and Line Data for Super Sun Seeds,” on page 317 generate the output. Based on the input invoice data, one of the following three output page types would be created:

- Single page invoice
- First or interior page of multi-page invoice
- Last, or summary, page of a multi-page invoice

400 CPU Parkway Vegetation, NJ 55090 		Office: 555-499-2367 Fax: 555-415-9794	
THE LAST LEAF 340 DESPERADO COURT LONGVIEW CA 12345-6789		CAMBIUM LAYER LIMITED 2222 SAPLING CIRCLE BAKERSVILLE BC 47365-7290	
Customer Number: 154	Invoice Number: 31354	Invoice Date: 11/06/02	Payment Date: 12/06/02
Ship Via: TREE TRUCK		Ship Date: 11/06/02	Terms: NET 15
Rep: MARY PINETREE		Rep: MARY PINETREE	

Qty	UOM	Item #	Item Description	Price	Extension
9	EA	00231300	SEED ROASTER OVEN SET		
9	EA	00231300	SEED ROASTER OVEN SET		
4	PK	03698741	STRING GRAPEFRUIT		
4	PK	03698741	STRING GRAPEFRUIT		
300	EA	11000146	AZALEA, GIANT ROSE SEEDS		
300	EA	11000146	AZALEA, GIANT ROSE SEEDS		
6	CT	11005010	EARLY DWARF DANISH SEEDS		
24	DZ	11005013	MINCOR NANTES CARROT SEED		
6	DZ	11005013	MINCOR NANTES CARROT SEED		
24	PK	11005020	FRENCH PICKLING SEEDS		
12	CT	12382910	SUCCOTASH SEEDS		
12	CT	12382910	SUCCOTASH SEEDS		
55	CT	13145340	SOUR GRAPE SEED		
55	CT	13145340	SOUR GRAPE SEED		
14	BZ	32165478	BLACK EYED BANANA		
14	BZ	32165478	BLACK EYED BANANA		
600	DZ	44646510	PLUMP RED PLUMS		
600	DZ	44646510	PLUMP RED PLUMS		
40	DZ	45613712	CRANAPPLE BERRY SEEDS		
40	DZ	45613712	CRANAPPLE BERRY SEEDS		
4	PK	65412384	SEED SCRUBBER		
4	PK	65412384	SEED SCRUBBER		
1	PK	65412384	SEED SCRUBBER		
1	PK	65412384	SEED SCRUBBER		
12	DZ	84512130	SUB-ARCTIC TOMATO SEEDS		
12	DZ	84512130	SUB-ARCTIC TOMATO SEEDS		

Packing List

CONTINUED
Page 1

1	PK	65412384	SEED SCRUBBER	3.84	46.08
12	DZ	84512130	SUB-ARCTIC TOMATO SEEDS	3.84	46.08
12	DZ	84512130	SUB-ARCTIC TOMATO SEEDS	3.84	46.08


Customer Copy

CONTINUED
Page 1

1	PK	65412384	SEED SCRUBBER	3.84	46.08
12	DZ	84512130	SUB-ARCTIC TOMATO SEEDS	3.84	46.08
12	DZ	84512130	SUB-ARCTIC TOMATO SEEDS	3.84	46.08

File Copy

CONTINUED
Page 1

400 CPU Parkway Vegetation, NJ 55090 		Office: 555-499-2367 Fax: 555-415-9794	
THE LAST LEAF 340 DESPERADO COURT LONGVIEW CA 12345-6789		CAMBIUM LAYER LIMITED 2222 SAPLING CIRCLE BAKERSVILLE BC 47365-7290	
Customer Number: 154	Invoice Number: 31354	Invoice Date: 11/06/02	Payment Date: 12/06/02
Ship Via: TREE TRUCK		Ship Date: 11/06/02	Terms: NET 15
Rep: MARY PINETREE		Rep: MARY PINETREE	

Qty	UOM	Item #	Item Description	Price	Extension
951	PK	11005018	EARLY BANTAM SEEDS		
46	PK	11005018	EARLY BANTAM SEEDS		
45	PK	11005018	EARLY BANTAM SEEDS		
4	EX	11057893	AFRICAN DAISY, SEEDS		
4	EX	11057893	AFRICAN DAISY, SEEDS		
100	EA	31321655	SEMILLAS DEL SUS SOMBREROS		
100	EA	31321655	SEMILLAS DEL SUS SOMBREROS		
1000	EX	56413213	POT POT		

Packing List

Page 2

Total Due: \$26,882.54

Return this tear-off strip with your payment. Make Checks Payable to: Super Sun Seeds

Payment is due by: 12/06/02 Amount Due is: \$26,882.54

THE LAST LEAF
340 DESPERADO COURT
LONGVIEW
CA 123456789

Customer Copy

Page 2

Total Due: \$26,882.54

Return this tear-off strip with your payment. Make Checks Payable to: Super Sun Seeds

Payment is due by: 12/06/02 Amount Due is: \$26,882.54

THE LAST LEAF
340 DESPERADO COURT
LONGVIEW
CA 123456789

File Copy

Page 2

Figure 115. Three Copies of a Two-Page Customer Invoice

Record Format and PPFA

PPFA page definitions were designed to map traditional line data into new formats without changing the application. This works well for regular data in fixed formats but is not flexible enough for reports whose format varies, such as phone bills with different formats for different services or bank statements with variable lines and layouts for different types of accounts. In these situations, the application still needed to do the routine bookkeeping, such as adding the correct headers and footers, numbering the pages, and dealing with page overflow. Record format line data presents a solution to these problems. Record format line data is a form of line data that is supported by PSF and formatted by a page definition. With this format, each data record contains a 10-byte record identifier (IDs are byte strings, not text), which selects the record descriptor (RCD) in a record format page definition that is used to format the line data

PPFA has a record formatting function that lets you work with this type of data. The function automates page management; such as adding headers, footers, and page numbering. It works with grouped-item and page overflow, can add subheadings and simple graphics (lines, boxes, circles, ellipses, and filled areas). It also can add color and text enhancements, such as delimited fields on input and right and left alignment on output. For more information about using PPFA to create page definitions for record format line data, including a complete example, refer to *Page Printer Formatting Aid: User's Guide*.

Using the Record Formatting Functions

To take advantage of the record format capabilities, each print record must contain a 10 byte Record ID. Each Record ID selects a Layout command in the page definition to format it.

Example: crdata DEPOSIT 01/05/99 \$ 26.90 selects the Layout named crdata.

Specifying a Layout

You use Layout instead of Printline, not both in the same page definition. There are four Layout types:

Body The Body layout is used for most data. It produces detail lines.

Page Header

The Page Header is automatically produced on each page. You can use the default header or your own. If you use your own headers, they can contain variable data and are invoked by a Record ID.

Page Trailers

Page Trailers work like page headers.

Group Header

A Group Header precedes a group of Body records. It is continued on next page if there is an eject before the group ends. It remains in effect until a new Group is invoked or Body NOGROUP is specified.

Examples:

```
LAYOUT C'layout1' BODY GROUP DELIMITER X'05' DIRECTION ACROSS COLOR CYAN OVERLAY OVL1 0.5 0.25 OVROT
```

```
LAYOUT C'headr1' GRPHEADER XSPACE .2 IN POSITION SAME .6 IN;
```

```
LAYOUT C'pghd' PAGEHEADER NEWPAGE POSITION .6 IN ABSOLUTE .55 IN;
```


Using the Field Command

The Field command identifies a field in a data record to be formatted and placed on the page. You can specify the start position and length or a delimited field number. For the field you can specify:

- Position on the page
- Print direction
- Left or right Alignment
- Font or bar code
- Automatic page number

Examples:

```
LAYOUT C'layout1' BODY
FIELD START 1 LENGTH 20 ALIGN LEFT POSITION 0 CURRENT FONT FONT1;
layout1 field 1 data      field 2 data
```

```
LAYOUT C'layout2' BODY DELIMITER C'%'
FIELD FLDNUM 1 ALIGN RIGHT POSITION 0 NEXT FONT FONT2;
layout2 %field 1 data%field 2 data%
```

Notes:

1. You cannot use TRCs to select a font; the layout selects the fonts used to format a record.
2. Carriage controls are required for mixed-mode (X'5A') and are ignored otherwise.

Defining Color

The DEFINE command defines a color model to use. You can specify one of these values:

OCA Use for named colors

RGB Specify rvalue gvalue bvalue

CMYK Specify cvalue mvalue yvalue kvalue

CIELAB

Specify lvalue - clvalue - c2value

HIGHLIGHT

Specify hvalue

COVERAGE

Specify cvalue

BLACK

Specify bvalue

Example:

```
DEFINE puce RGB 50 40 60;
```

Drawing Graphics

You can use the Drawgraphic command to draw single or repeated objects in one command. You can create these simple graphics in various styles with this command: boxes, circles, ellipses, and lines.

Examples:

- This command draws a filled box with rounded corners. The outline is solid green.

```
DRAWGRAPHIC BOX GRAPHID 01 POSITION LPOS + 0.25 IN NEXT
          BOXSIZE 2.0 IN 1.0 IN ROUNDED MEDIUM LINEWT BOLD
          LINETYPE SOLID COLOR GREEN FILL DOT04;
```
- This command draws a dotted, magenta line that slopes from upper left to lower right.

```
DRAWGRAPHIC LINE POSITION NEXT NEXT ACROSS TO - 3 IN
          LINETYPE DOTTED COLOR MAGENTA;
```
- This command draws a vertical orange line in a dot-dash pattern. The line is then copied four times, each one .25 inches to the right of the previous one.

```
DRAWGRAPHIC LINE POSITION LPOS CPOS +1.0
DOWN 3.0 IN LINETYPE DASHDOT LINEWT MEDIUM
COLOR ORANGE COPY ACROSS 4 SPACED 0.25;
```
- This command draws a circle with a solid outline and a patterned (diagonal lines) fill.

```
DRAWGRAPHIC CIRCLE POSITION LPOS NEXT
          RADIUS 1.0 IN LINEWT MEDIUM LINETYPE SOLID
          FILL BLTR1;
```
- This command draws an ellipse with a solid outline and a patterned (vertical lines) fill.

```
DRAWGRAPHIC ELLIPSE POSITION LPOS NEXT
          AXIS1 -2 -2 AXIS2 + 2 +2
          LINEWT MEDIUM LINETYPE SOLID
          FILL VERTLN;
```

Creating Variable-Length Graphics

You can use the Endgraphic command to extend a box or line to enclose a variable number of input lines.

Example:

These commands create a variable-length box. Possible results are shown in Figure 116.

```
DRAWGRAPHIC BOX GRAPHID 01 BOXSIZE 3 IN POSITION 1.75 IN .3 IN;
ENDGRAPHIC GRAPHID 01 LPOS;
```

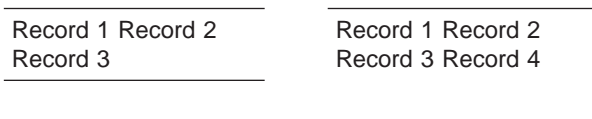


Figure 116. Variable-Length Boxes

Chapter 17. Using Infoprint Designer

Infoprint Designer (product number 5733-ID1) is an iSeries licensed product for transforming existing application output to advanced electronic documents. Infoprint Designer helps you transform applications such as those that use preprinted forms with *LINE or *SCS data to full-function electronic documents. Because Infoprint Designer builds on the integrated page architecture of the iSeries, the document design and reengineering can be accomplished without changes to the line of business application program. In this chapter we describe Infoprint Designer and its components, then show how you can use it to create an application solution to the Super Sun Seeds case study.

Overview

Infoprint Designer provides a fully-graphical document composition interface to the iSeries printing and e-output system. It supports the requirements of today's complex documents and reports, producing fully electronic documents that combine data, text, overlays, graphics, images, bar codes, and typographic fonts. You can use Infoprint Designer to design new output applications or re-engineer existing applications.

Infoprint Designer consists of three components:

- **Infoprint Overlay Editor** for design of overlays (electronic forms).
- **Infoprint Image Editor** for design of images (iSeries page segments).
- **Infoprint Layout Editor** for design and placement of application data as well as formatting the other page elements (overlays, images).

Infoprint Designer can seamlessly retrieve current application data (spooled files) for you to use in your design, then upload all component resources to the OS/400 so the application can be put into production without application changes.

In the background, Infoprint Designer defines the format of pages with page definition and form definition resource objects. Unlike DDS, these resources are application-independent, although they are compatible with applications that define output data with DDS. OS/400 will automatically write these applications as full AFP, which ensures viewing and PCL support.

Figure 117 on page 198 illustrates how Infoprint Designer integrates with the iSeries. Before implementing a new application template, the iSeries application produces a simple SCS or line data spooled file. The printer file that is used by the application determines the format of this spooled file. With Infoprint Designer, you seamlessly download application output data into Infoprint Designer, design images and overlays, and merge the data into a completed document. Next, you upload your overlays, images, bar codes, fonts and the Infoprint Designer-created page definition and form definition to OS/400. In order to put them production, you add references to the page definition and form definition to the printer file. The next time the application runs, instead of producing simple SCS or line data, it produces fully graphical AFP.

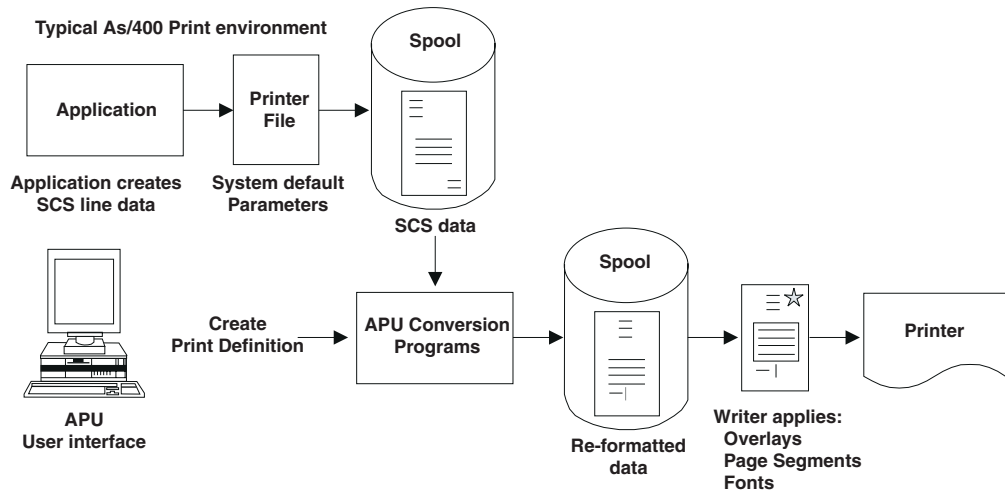


Figure 117. Infoprint Designer Application Flow

For more information about Infoprint Designer, including a detailed example and instructions about advanced use of the product, refer to the Redbook *IBM @server iSeries Printing VI: Delivering the Output of e-business* or *Infoprint Designer for iSeries: Getting Started*.

What You Can Do with Infoprint Designer

Output specifications for iSeries application programs generate SCS, line, or AFP spooled files. Infoprint Designer takes SCS or line data spooled files and creates page definitions, form definitions, and overlays for use with line data files. SCS is a line-oriented data stream that typically uses preprinted forms to create the final document. For these types of output applications, Infoprint Designer eliminates the need for preprinted forms. With Infoprint Designer, you can:

- Create multi-copy documents, with each page or copy customized. Infoprint Designer provides a Wizard to make this simple.
- Use data that is contained within a page to determine page formatting.
- Re-map any field that the input pages contain. For example, you can change the position, font, orientation, and color.
- Add and edit images to application documents.
- Print application data in any of the standard bar code symbologies.
- Add document elements such as overlays, images, lines, boxes, and constant text.
- Place a new application into production for automatic processing without changes to the underlying application.
- Create complex, fully graphical documents that can be used in “downstream” processes such as PDF presentation, e-mail, fax, and printing.

Many of these capabilities are illustrated in “Super Sun Seeds Case Study” on page 201.

Overlay Editor

The Infoprint Overlay Editor is a component of Infoprint Designer for graphically designing electronic forms (overlays). The Overlay Editor includes these functions:

- Display and edit multiple overlays per project
- Create lines, boxes, circles in multiple styles
- Create shaded or rounded box elements
- Place text standalone or within a defined region with justification options
- Duplicate and repeat functions; undo and redo changes
- Import TIFF images “on-the-fly” into the document (and into page segment format)
- Import existing AFP overlays
- Scan an image of an existing preprinted form to use as a template for design
- Align overlay elements with the autoalign feature
- Work without worry by using the autosave feature

Figure 118 shows the Overlay Editor with an overlay opened in it.

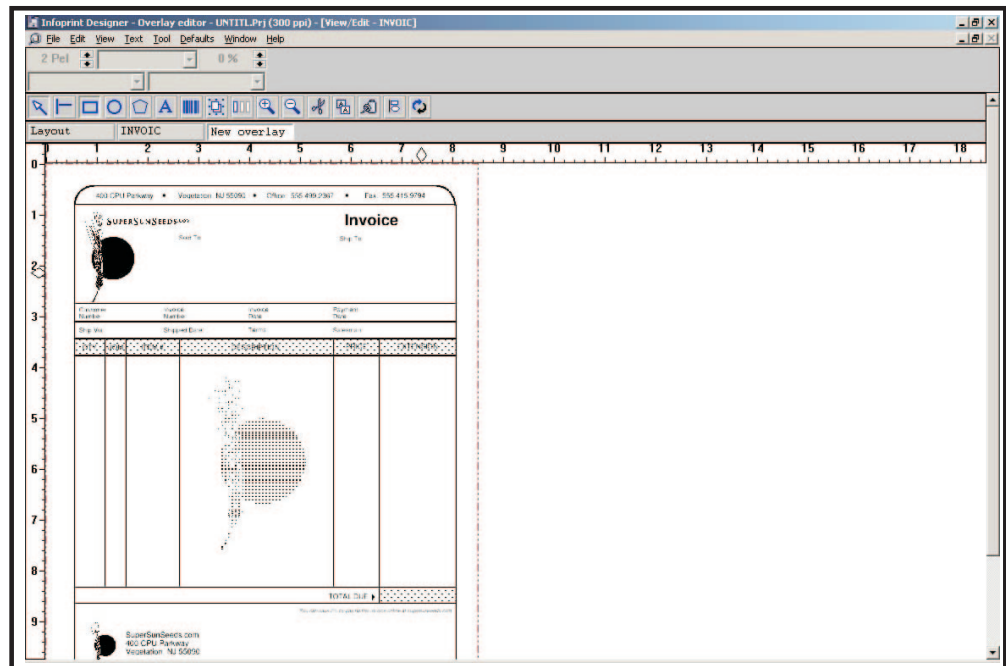


Figure 118. Overlay Editor

Image Editor

The Infoprint Image Editor is the component of Infoprint Designer for image design and conversion. With the Image Editor you can:

- Create sharp, high-fidelity images
- Edit images in native iSeries format
- Import TIFF images
- Touch-up images professionally using the toolkit of image editing tools
- Edit image a pel level for precise control
- Enter text within the image

- Rotate image in 1-degree increments
- Rescale, resize, cut, paste, crop, invert, shade, flip, and mirror the image
- Preview the image
- Create or edit an image in full color, including the latest FS45 color image specification

Figure 119 shows the Image Editor with the Super Sun Seeds logo displayed.

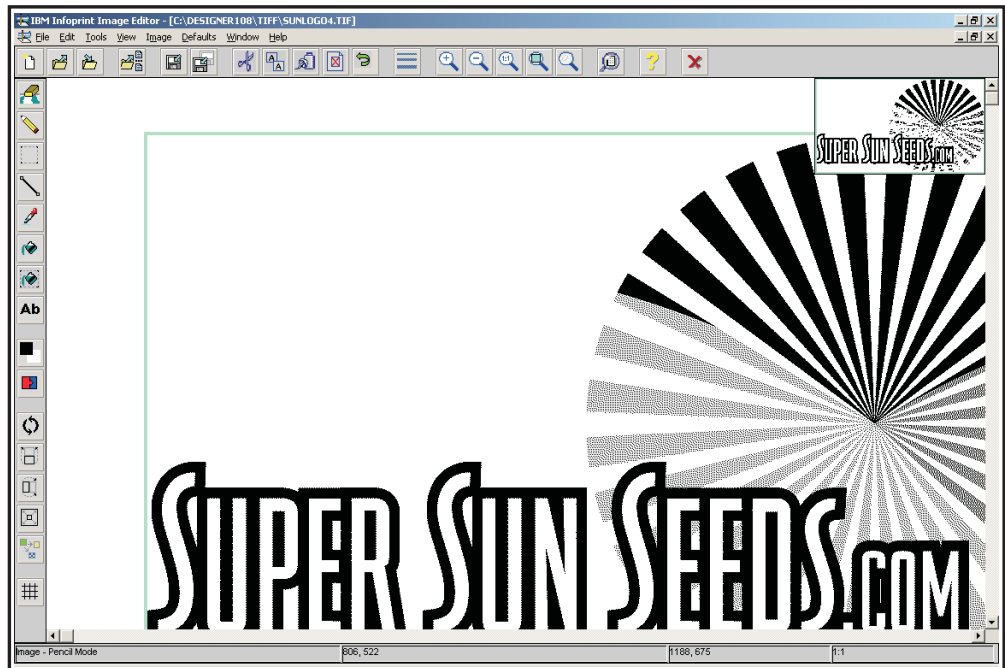


Figure 119. Super Sun Seeds Logo Scanned into the Image Editor

Layout Editor

The Layout Editor is the central design component of Infoprint Designer. It lets you merge the data from OS/400 spooled files with overlays, images, bar codes, and fonts - creating the iSeries template for the desired fully graphical document. You can then upload the whole project to the OS/400 and use it in production. The Layout Editor includes these functions:

- iSeries Access integration for download of application output data and upload of completed projects
- Easy drag-and-drop manipulation of application data
- Fully graphical page preview, you see what will print
- Ability to browse and select target print file
- Separate graphical windows for programmer designers
- Full conditional processing – page layout is based on test of page data
- Design wizard for ease in designing more complex document applications
- Full implementation of iSeries page (AFP) architecture
- Seamless, full integration into iSeries output architecture

Figure 120 shows the Layout Editor window. An overlay with no data on it is in the left pane. The data that is going to be placed on the overlay is displayed in the right pane.

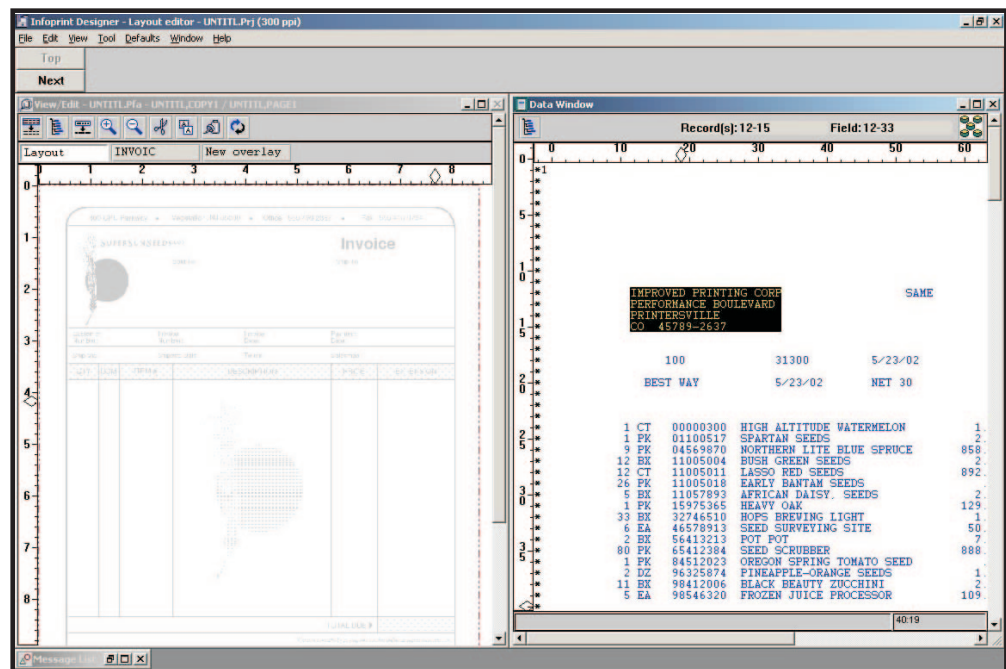



Figure 120. Layout Editor Window

Starting in version 1.11 (provided with PTFs SF67603 and SF67604), Infoprint Designer comes with a Layout wizard, which makes it easier for you to create complex layouts in the Layout Editor. To use the wizard, design your first output

page then select the wizard icon:  from the toolbar. The wizard lets you specify how the next output page is different from the first one, helping you create a modified copy of the first page (with different overlays or suppressed data, for example) or to change layouts based on a trigger in the data file. You can use the wizard as many times as you need to in a project to design all of the different layouts you need for a print job.

Super Sun Seeds Case Study

In this section we apply Infoprint Designer to the Super Sun Seeds case study. We will use Infoprint Designer to create a redesigned output application using standard iSeries page resources (overlays, page segments, fonts, page definitions, and form definitions).

Applying Infoprint Designer to the Super Sun Seeds case study is a simple, straightforward application. As explained in Chapter 12, "Introduction to the Super Sun Seeds Case Study," on page 113, the existing Super Sun Seeds invoicing application prints on a preprinted invoice. The data to be printed, shown in Figure 121 on page 202, is produced by program INVSCS.

IMPROVED PRINTING CORP		SAME	
PERFORMANCE BOULEVARD			
PRINTERSVILLE			
CO 45789-2637			
100	31300	1/26/02	2/26/02
BEST WAY	1/26/02	NET 30	YOUR PRINTER REP
1 CT	00000300	HIGH ALTITUDE WATERMELON	1.01 1.01
1 PK	01100517	SPARTAN SEEDS	2.39 2.39
9 PK	04569870	NORTHERN LITE BLUE SPRUCE	858.32 7,724.88
12 BX	11005004	BUSH GREEN SEEDS	2.50 30.00
12 CT	11005011	LASSO RED SEEDS	892.23 10,706.76
26 PK	11005018	EARLY BANTAM SEEDS	.38 9.88
5 BX	11057893	AFRICAN DAISY, SEEDS	2.35 11.75
1 PK	15975365	HEAVY OAK	129.09 129.09
33 BX	32746510	HOPS BREWING LIGHT	1.20 39.60
6 EA	46578913	SEED SURVEYING SITE	50.00 300.00
2 BX	56413213	POT POT	7.65 15.30
80 PK	65412384	SEED SCRUBBER	888.79 71,103.20
1 PK	84512023	OREGON SPRING TOMATO SEED	.97 .97
2 DZ	96325874	PINEAPPLE-ORANGE SEEDS	1.29 2.58
11 BX	98412006	BLACK BEAUTY ZUCCHINI	2.30 25.30
5 EA	98546320	FROZEN JUICE PROCESSOR	109.90 549.50

Thank You
 Because you have ordered
 over \$500 of seeds this
 year, on your next seed
 order you will receive
 a 10% discount.

\$90,652.21


2/26/02
 IMPROVED PRINTING CORP
 PERFORMANCE BOULEVARD
 PRINTERSVILLE
 CO 457892637

\$90,652.21

Figure 121. The SCS File to be Formatted

Infoprint Designer is used to transform the output of INVSCS to an electronic invoice. The formatted output that we want to produce is shown in Figure 122 on page 203.

400 CPU Parkway Vegetation, NJ 55090 Office: 555-499-2367 Fax: 555-415-9794



SUPER SUN SEEDS.com

Invoice


Sold To
IMPROVED PRINTING CORP
 PERFORMANCE BOULEVARD
 PRINTERSVILLE
 CO 45789-2637

Ship To
SAME

Customer Number: 100	Invoice Number: 31300	Invoice Date: 7/17/02	Payment Date: 8/17/02
Ship Via: BEST WAY	Shipped Date: 7/17/02	Terms: NET 30	Rep: YOUR PRINTER REP

Qty	UOM	Item	Item Description	Price	Extension
1	CT	00000300	HIGH ALTITUDE WATERMELON	1.01	1.01
1	PK	01100517	SPARTAN SEEDS	2.39	2.39
9	PK	04569870	NORTHERN LITE BLUE SPRUCE	858.32	7,724.88
12	BX	11005004	BUSH GREEN SEEDS	2.50	30.00
12	CT	11005011	LASSO RED SEEDS	892.23	10,706.76
26	PK	11005018	EARLY BANTAM SEEDS	.38	9.88
5	BX	11057893	AFRICAN DAISY, SEEDS	2.35	11.75
1	PK	15975365	HEAVY OAK	129.09	129.09
33	BX	32746510	HOPS BREWING LIGHT	1.20	39.60
6	EA	46578913	SEED SURVEYING SITE	50.00	300.00
2	BX	56413213	POT POT	7.65	15.30
80	PK	65412384	SEED SCRUBBER	888.79	71,103.20
1	PK	84512023	OREGON SPRING TOMATO SEED	.97	.97
2	DZ	96325874	PINEAPPLE-ORANGE SEEDS	1.29	2.58
11	BX	98412006	BLACK BEAUTY ZUCCHINI	2.30	25.30
5	EA	98546320	FROZEN JUICE PROCESSOR	109.90	549.50
<small>Thank You Because you have ordered over \$500 of seeds this year, on your next seed order you will receive a 10% discount.</small>					
Total Due					\$90,652.21

You can save 2% by paying this invoice online at superseeds.com



SuperSunSeeds.com
 400 CPU Parkway
 Vegetation, NJ 55090

Page 1

Figure 122. Super Sun Seeds Invoice

To do this, we will break down the design project into these tasks:

- Specify only one page format, the basic invoice page.
- Select a sample spooled file.
- Specify general page layout options, using data in the spooled file.
- Place the INVOIC overlay on front.

Before you begin - optional:

If you have Infoprint Designer installed, you can duplicate the case study solution that is illustrated in this chapter. In order to do this, make sure that these setup steps are done:

- Ensure that you have version 1.11 of Infoprint Designer.
- Infoprint Designer comes with a demonstration library called IPDATA that includes sample tools to use on your iSeries server. If you want to use the sample data, follow these steps:
 1. Install the IPDATA library on your iSeries.
 2. Add IPDATA to your library list using the command: ADDLIB IPDATA.
 3. Issue the command INV (accept the default parameters) to create a spooled file called INVSCS on your current output queue for use with this example.

For more information about installing and using the IPDATA sample application library, refer to *Infoprint Designer for iSeries: Getting Started*.

Steps to create the Super Sun Seeds electronic invoice:

1. When you start Infoprint Designer, it opens with the Overlay Editor started so you can begin creating an overlay. You will draw your overlay in the **View/Edit** window, shown in Figure 123.

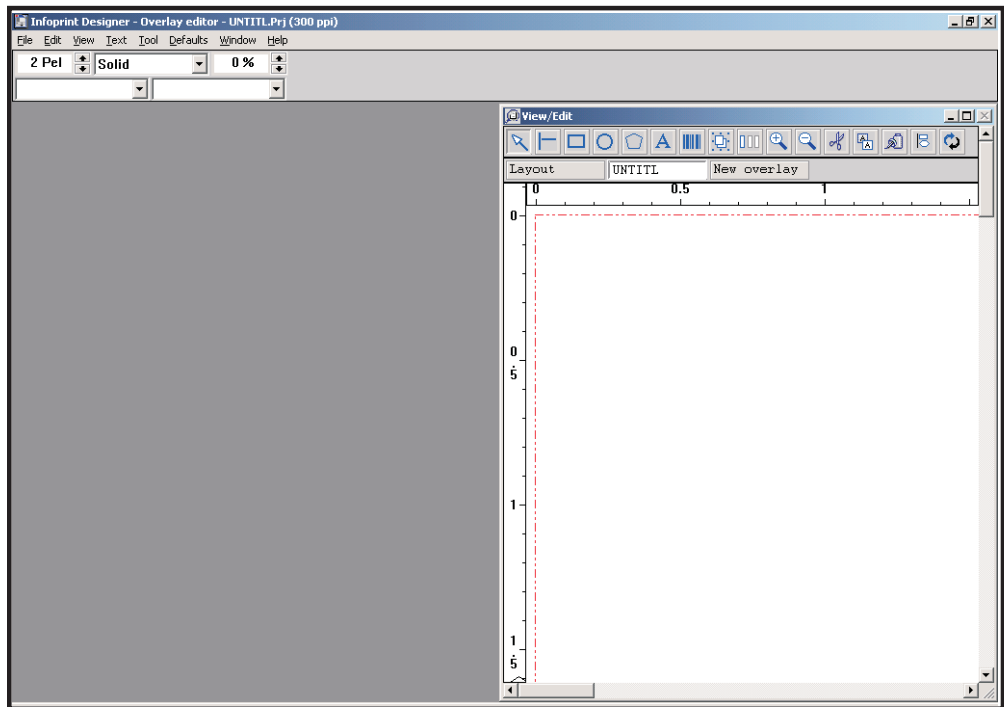


Figure 123. New Project

To help you place overlay elements, ensure that you have an appropriate grid defined and that Snap to grid is on.

- To specify the grid size and units, from the **Edit** menu, select **Preferences**. Use the **Units and Grid** page to specify how you want your grid.
- To turn on and off the Snap to grid feature, from the **View** menu, select or deselect **Snap to Grid**.

The default values for the grid are acceptable for the Super Sun Seeds sample project, so we will not change them.

2. We will start by drawing the outermost box in the overlay. Select the Box



drawing tool: Draw a box with the upper left-hand corner .4 in down from the top and .5 in from the left edge. The **X** and **Y** fields in the toolbar show the distance from the left and top edges, respectively. Make the box 7.5 in wide and 10.25 in. tall. The **W** and **H** fields in the toolbar show the width and height of the box you are creating.

Note: The units shown in the toolbar are, by default, the units specified for your grid. You can change these units from the **Preferences** dialog. Alternatively, you can change the units on the toolbar after drawing something on the page.

You should see a large rectangle in the overlay.

3. To round the corners of the box, right-click anywhere on the border of the box. The **Box properties** dialog appears. Select the **Rounding** page, as shown in Figure 124 on page 205. Under Rounded corners, select **All** and make sure the

Rounding size is set to MEDIUM. Click **OK** and the box now has rounded corners on the overlay.

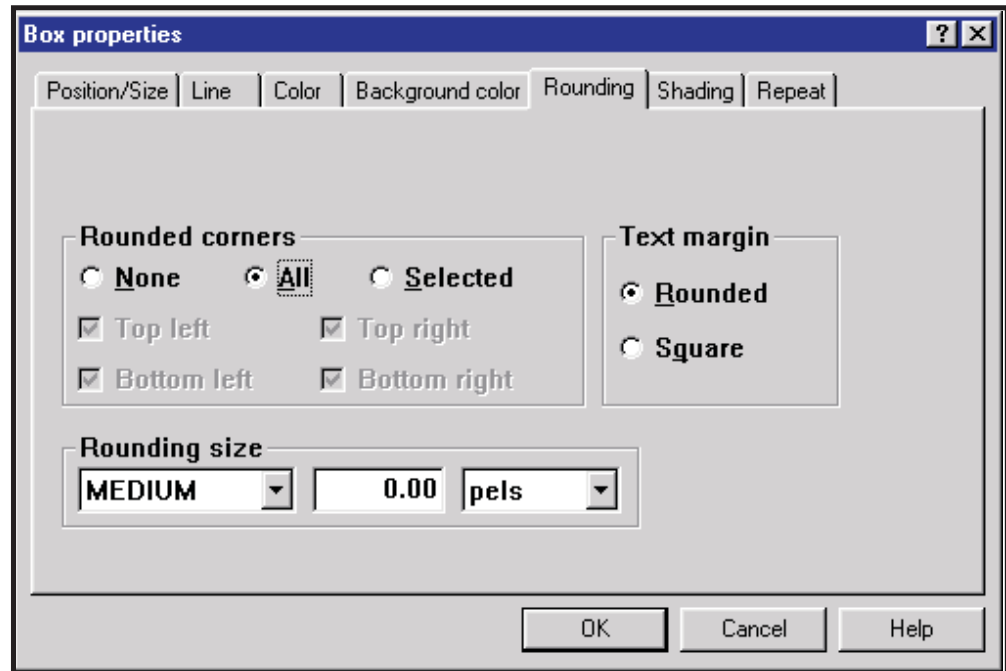


Figure 124. Box Properties Dialog - Rounding Page

- Next we will add a line inside the box. We will add the one that appears beneath the company logo and above the row starting with Customer Number.



Select the Line drawing tool: Draw a line starting at X = .55 inches and Y = 2.75 inches and connecting to the right side of the box. The L field in the toolbar should show that the line is 7.5 inches long.

To make sure that the line is attached to the left edge of the box, select the line and the box border. To select multiple objects, select the first one then hold down **Shift** while you select the rest. Both elements should now be surrounded



with a thin blue rectangle. Next, select the **Align** tool: The **Align objects** dialog is displayed. Select the picture that shows left alignment and click **OK**.

- Notice there is another line beneath the text "Customer Number". Create that line by repeating the line you just drew. Select the line and then click the



Repeat tool: . On the **Repeat** page, select **Repeat manually** then select **Place**. Place the new line at X = .55 in. and Y = 3.25 in.



- To enter the text block "Customer Number," select the Text tool: Starting at X = .65 inches and Y = 2.90 inches, draw a box with a width of 1.2 inches (use the ruler at the top to approximate the width) and a height tall enough to fit between the two lines you drew. When you release the mouse button, the **Select font** dialog is displayed, as shown in Figure 125 on page 206.

After selecting a typeface (such as Helvetica), choices are made available for size, weight/width/design, and code page. Select a value for each of these font attributes; for this example use “Helvetica 10 point”, “Medium/medium/normal” and “USA/Canada - CECP”. Click **OK**.

Type **Customer Number**. When you type the N in Number, the text wraps onto the next line. Press **Esc** when you are done entering text.

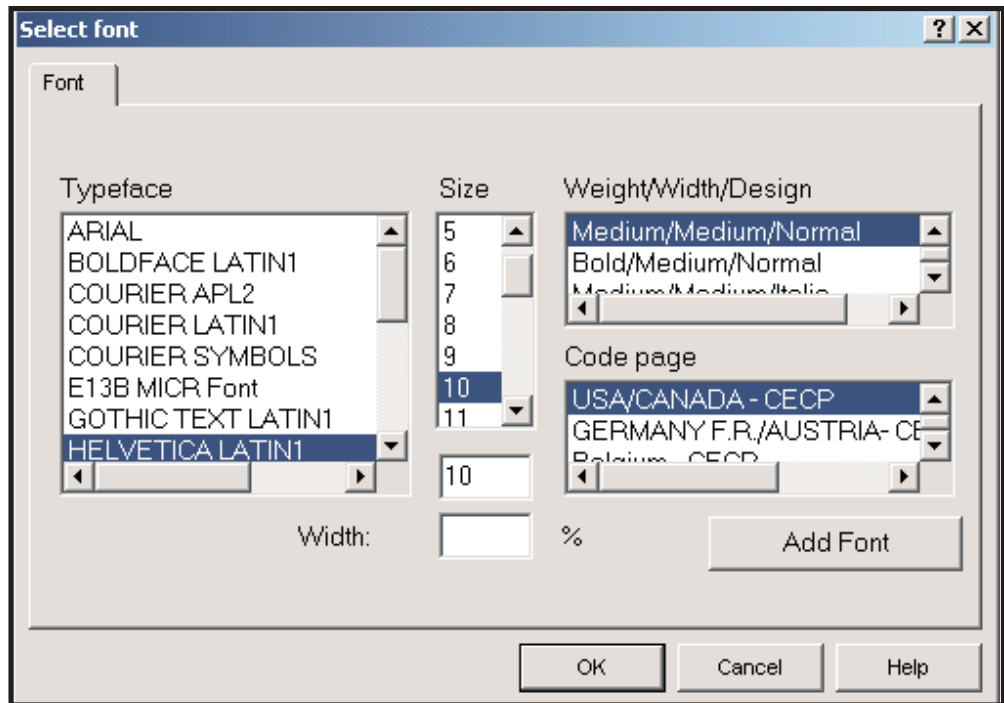


Figure 125. Select Font Dialog

7. Now we will retrieve the Super Sun Seeds logo, which is stored as an AFP page segment. Click **File** and select **Get image**. Browse to the Designer\PSEG300 directory and select the file named **SS2TOP.300** and click **OK**. The selected image appears at the upper left corner. You can drag the image anywhere you want it in the overlay. We placed the logo at X=.5 inches, Y=.7 inches.
8. You can create the remaining objects in the overlay using the techniques described in steps 2 - 7. To save your work on the overlay, click **File** and select **Save as** and name the project. The same name will be given to the overlay file by default.
9. Now you are ready to begin mapping data. To open the Layout Editor, click the **Layout** button on the View/Edit toolbar. The **Get sample data - Select source** dialog opens. If you installed the IPDATA INV invoice application on your iSeries and created a sample spooled file, select **From AS/400**. If you want to use the PC version of the sample file shipped with Infoprint Designer, select **From PC**. Click **OK**.

The **Get sample data** dialog is displayed where you can select the name of the output queue containing the spooled file. The spooled files in that output queue are then displayed so you can select the desired spooled file, as shown in Figure 126 on page 207. Select the file named INVSCS that you created earlier and then provide a name for the backup data file on the PC.

If you are getting the data from your PC, in the **Get sample data** dialog, browse to **Designer\Data** and open **Invscs.asc**.

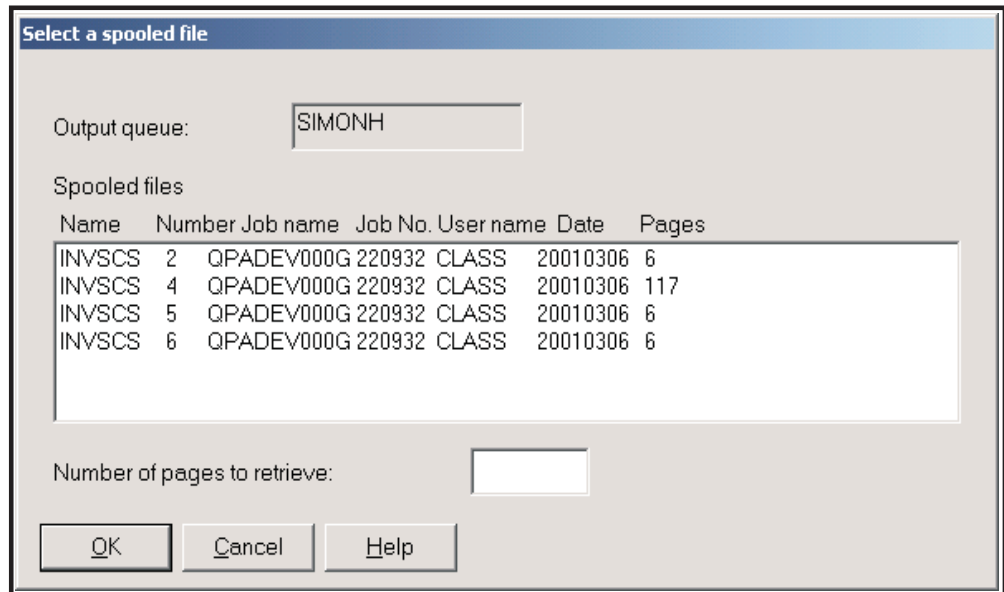


Figure 126. Selecting the Spooled File

10. After the spooled file is selected, the **Layout properties** dialog opens, as shown below. Here you can specify the page size (Letter, in this case), orientation (portrait), and sides (simplex).

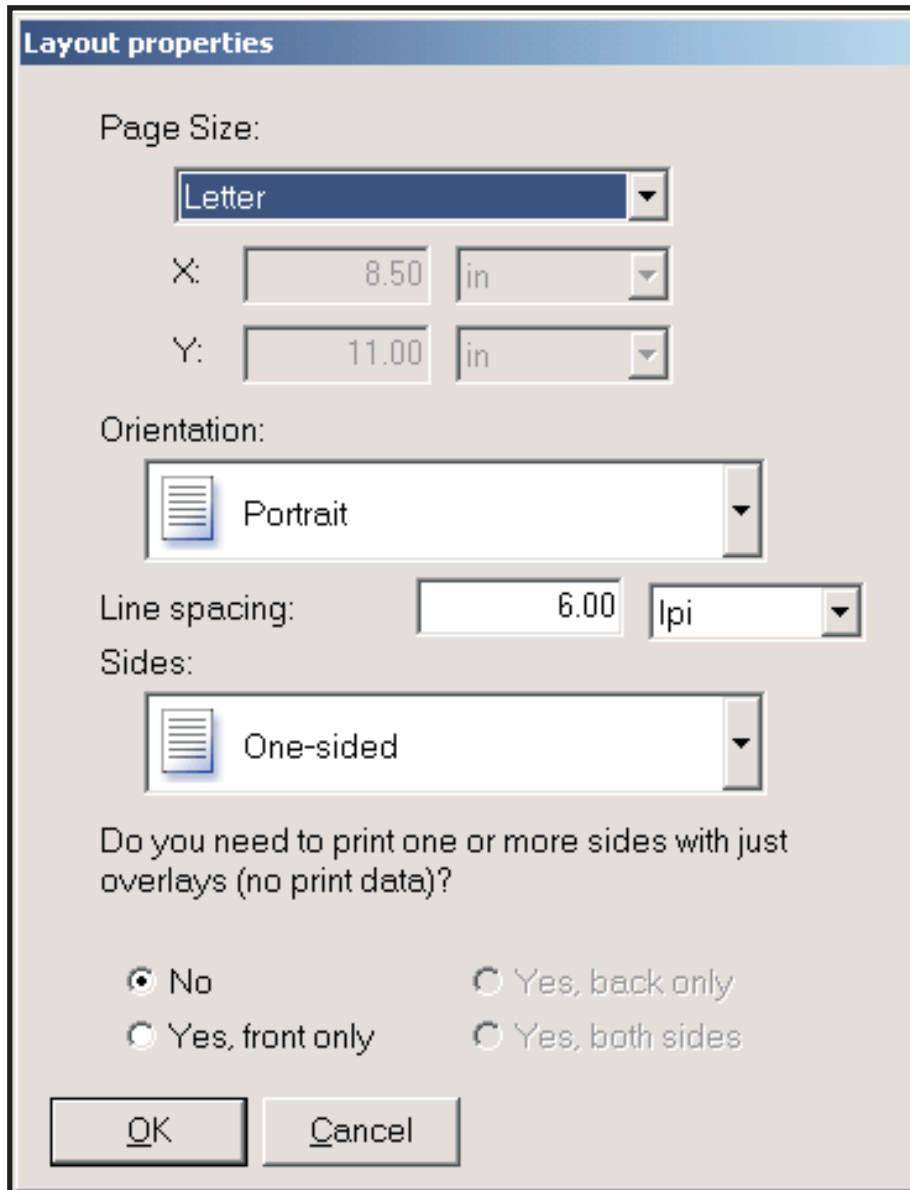


Figure 127. Layout Properties Dialog

Click **OK** and your Designer window should look like Figure 128 on page 209. You might need to click **Window** and select **Data** to open the Data window.

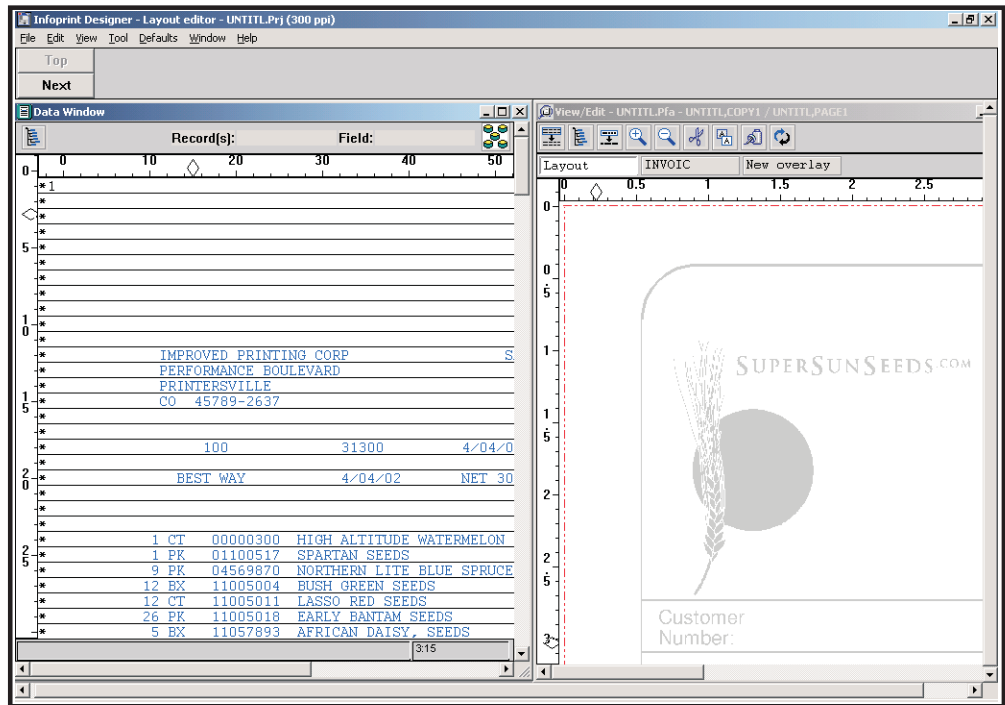


Figure 128. Spooled File Displayed in the Data Window

11. To begin mapping data, hold down the left mouse button and select the entire address block. Right-click (and hold down) anywhere inside the selected block and drag the text over to the **View/Edit** window until you have the address positioned at X = 2.5 in. (or 600 pels), Y = -.25 in. (or -60 pels). Figure 129 on page 210 shows the data being positioned in the **View/Edit** window. You can see the X and Y coordinates at the top, right-hand corner of the screen.

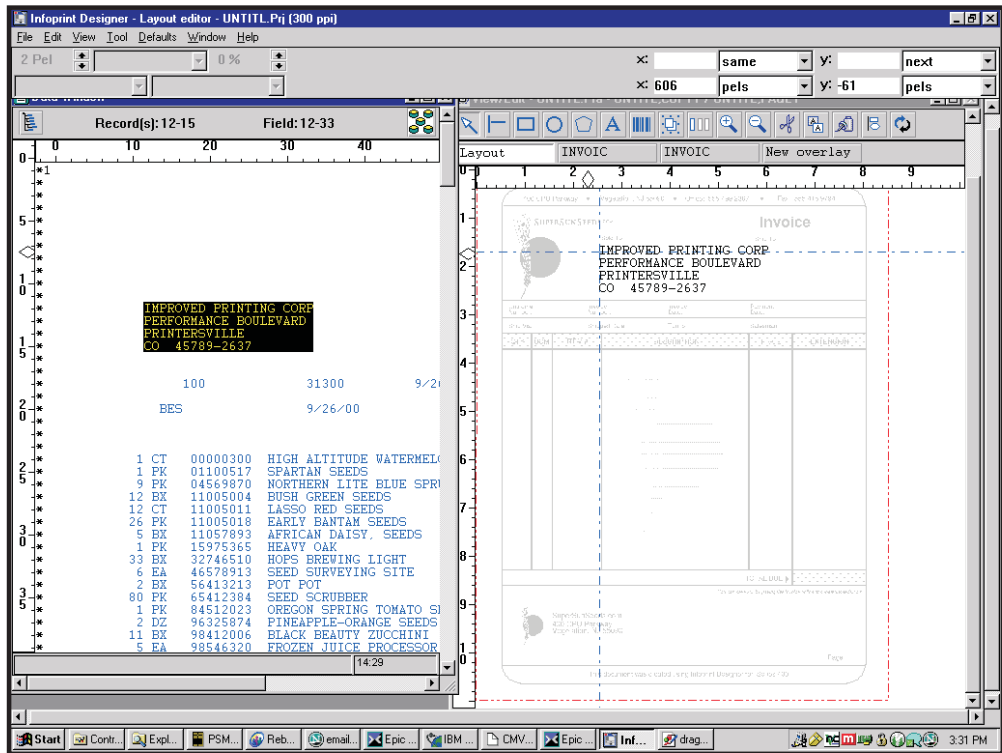


Figure 129. Positioning Text in the View/Edit Window

To change the font for the address block, right-click anywhere in the field. The **Field properties** dialog opens. On the **Font** page, select the same font attributes you used for the text in the overlay (10-point Helvetica, medium, USA/CANADA). Click **OK**. The font is changed in the **View/Edit** window.

- To map the zip code as a bar code, select the first five digits of the zip code in the data window. Drag the data onto the **View/Edit** window and place it beneath the address block. Right-click it to open the **Field properties** dialog. Click the **Bar code** button on the **General** page to open the **Bar code options** dialog, as shown in Figure 130 on page 211. Enter a name for this type of bar code (for example, POST), then select POSTNET as the bar code type and click **OK**.

Click **OK** on the **Field properties** dialog and you should see the zip code shown in Postnet bar code in the **View/Edit** window.

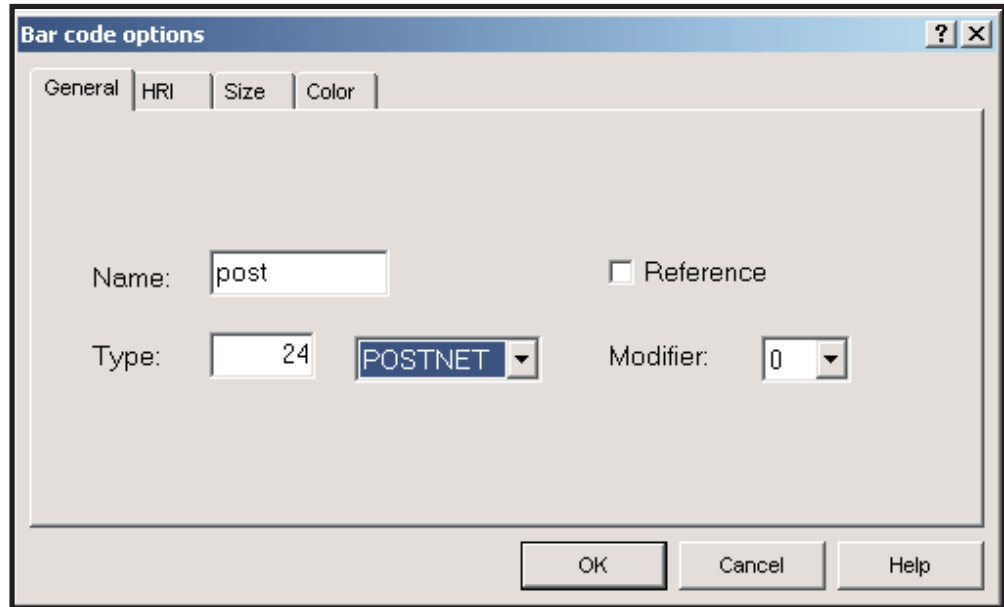


Figure 130. Bar Code Options Dialog

13. Although we did not complete the column headings in the overlay, experiment with mapping the invoice detail lines as a block. In the **Data** window, select columns 10 through 80 on records 24 through 42 (starting at the top left corner of the data you want to select). Drag the selected block to the **View/Edit** window and drop it under the second line we drew on the overlay. Notice that you can select the first record in the group and move the entire group of records for finer positioning. You can also select the bottom row of the group and then drag it down to change the interline spacing of all of the lines in that group.
14. If you find that the overlay does not allow enough room to place some data, click on the name of the overlay in the tabs in the **View/Edit** window to toggle from the Layout Editor to the Overlay Editor. Notice that the overlay comes into the foreground (text and graphical elements turn black) and the mapped data goes into the background (turns gray). You can now edit the overlay; lengthening a line or stretching a box with the mapped data there to guide you. When the overlay has been corrected, click the **Layout** tab and you toggle back to the Layout Editor.
15. After you map all of the data onto the overlay, as shown in Figure 131 on page 212, save the project and upload it to the iSeries. To save the project, click **File** and select **Save project**.

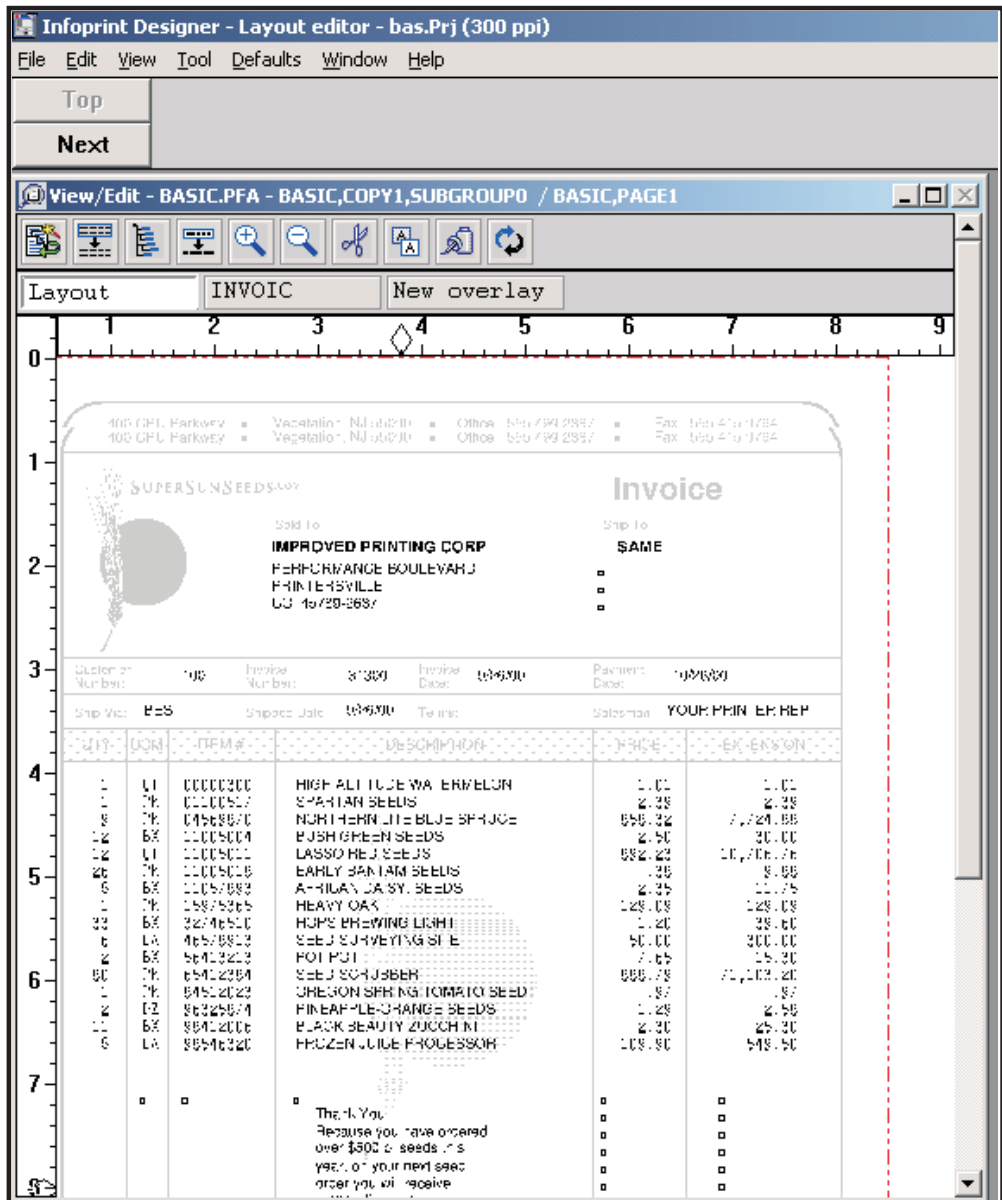


Figure 131. Finished Overlay with Data

- To upload the project to the iSeries, click **File** and select **Upload to AS/400**. The **Upload to AS/400** dialog opens. On the **Contents** page, select **Whole project** so that all the project resources (overlays, images, page definition, and form definition) that have been created based on your design will be uploaded and compiled. Open the **AS/400** page to verify your connection information, as shown in Figure 132 on page 213. Ensure that **Page segments** is selected so the Super Sun Seeds logo file is sent to your system as well as the resources you created. Click **OK**. A series of messages show the file transfer status.

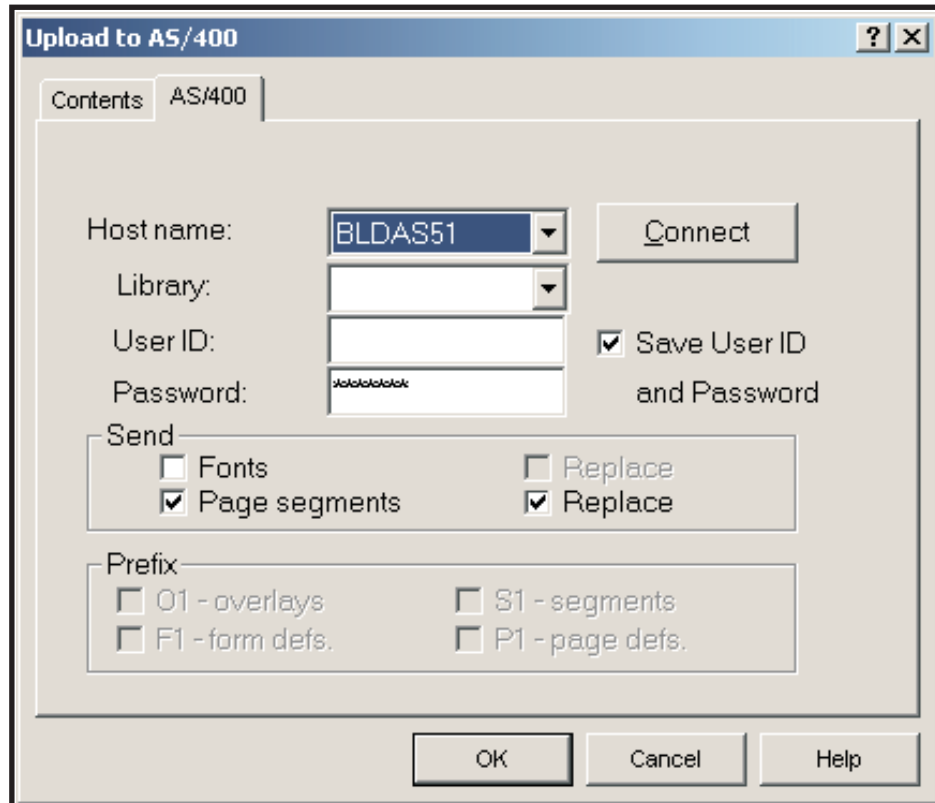


Figure 132. Upload to AS/400 Dialog, AS/400 Connection Page

17. Override your printer file to specify the form definition and page definition and specify a device type of *LINE. To change your printer file for the current session and user, enter this command, where *pagedef* and *formdef* are the page definition and form definition you created with Infoprint Designer (they have the same name as your project). *printer-file* is the printer file your application uses.

```
OVRPRTF FILE(printer-file) DEVTYPE(*LINE) PAGDFN(pagedef) FORMDF(formdef)
```

To use the OS/400 panels, enter OVRPRTF and press F4. Specify the printer file's name and specify *LINE for Printer device type, as shown below:

```

Override with Printer File (OVRPRTF)

Type choices, press Enter.

File being overridden . . . . . > seeds      Name, *PRTF
Overriding to printer file . . . *FILE      Name, *FILE
Library . . . . .                      Name, *LIBL, *CURLIB
Device:
Printer . . . . .                      Name, *SYSVAL, *JOB
Printer device type . . . . . *LINE    *SCS, *IPDS, *USERASCII...
Page size:
Page length . . . . .                .001-255.000
Page width . . . . .                 .001-378.000
Measurement method . . . . . *ROWCOL    *ROWCOL, *UOM
Lines per inch . . . . .              3, 4, 6, 7.5, 7.5, 8, 9, 12
Characters per inch . . . . .        5, 10, 12, 13.3, 13.3, 15...
Front margin:
Offset down . . . . .                0-57.790, *DEVD
Offset across . . . . .              0-57.790

More...
F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel
F13=How to use this display  F24=More keys

```

Figure 133. Override with Printer File - First Screen

Page through the screens until you can specify the page definition and form definition:

```

Override with Printer File (OVRPRTF)

Type choices, press Enter.

Font character set:
Character set . . . . .                Name, *FONT
Library . . . . . *LIBL              Name, *LIBL, *CURLIB
Code page . . . . .                    Name
Library . . . . . *LIBL              Name, *LIBL, *CURLIB
Point size . . . . . *NONE           0.1-999.9, *NONE
Coded font:
Coded font . . . . .                Name, *FNCHRSET
Library . . . . . *LIBL              Name, *LIBL, *CURLIB
Point size . . . . . *NONE           0.1-999.9, *NONE
Page definition . . . . . sunseeds    Name, *NONE
Library . . . . . *LIBL              Name, *LIBL, *CURLIB
Form definition . . . . . sunseeds    Name, *NONE, *DEVD
Library . . . . . *LIBL              Name, *LIBL, *CURLIB
AFP Characters . . . . .                Name, *NONE
+ for more values

More...
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

Figure 134. Override with Printer File - Specifying the Page Definition and Form Definition

For the IPDATA sample application, you can alternatively use the INV command instead of the Override with Printer File (OVRPRTF) commands above. On a command line, enter INV and press F4 to view the options. You can enter your project name for the page definition and form definition. The INV command will then execute the Override with Printer File (OVRPRTF) command and rerun the invoice program with the new printer file settings.

- 18. Respool your file.
- 19. Print your spooled file in one of these ways:
 - From the OS/400, enter WRKSPLF. Make sure the device you want to print to is started, then specify option 6 (Release) by your new spooled file.
 - To use iSeries Navigator, follow these steps:

- a. Double-click the iSeries Access icon.
- b. Double-click the iSeries Navigator icon.
- c. Open a connection to an OS/400.
- d. Expand **Basic Operations** then **Printer Output**. A list of spooled files is displayed in the right-hand pane.
- e. Right-click the spooled file you just created and select **Move**.
- f. In the **Move** dialog, specify **Printer**. You can specify a printer name or click **Browse** to search for one. The printers that are ready have a status of **Waiting for printer output**.

Chapter 18. Using IBM AFP Toolbox

IBM AFP Toolbox, hereafter referred to as AFP Toolbox, provides a high-level, application programming interface that enables programs written in ILE C, COBOL, and RPG programming languages to directly produce an AFP document that is composed of individual pages. Within each page, you can define page elements such as text, images, overlays, lines, boxes, tables, and bar codes. AFP Toolbox is very powerful, but is also more complex than other tools you can use to generate AFP. As such, AFP Toolbox should be used where its specific image manipulation or text formatting capabilities are required.

The MO:DCA-P documents created with AFP Toolbox are platform-independent and can be printed or viewed by the IBM AFP printer drivers on all supported platforms, such as z/OS, OS/400, RISC/6000, and AFP Workbench for Windows. AFP Toolbox output can be written to a file or returned to the application for further processing.

What Can I Do with AFP Toolbox?

Page definitions provide powerful functions for formatting line data, but might not provide enough flexibility or formatting capabilities for every application. Using AFP Toolbox is similar to, though more complex than, using DDS. If you need more control over the appearance of your output than you can get with page definitions or if you have specific application requirements, such as fully aligned text, that DDS cannot provide, you should consider using AFP Toolbox.

AFP Toolbox lets you create new print applications with formatting tailored to each page of data. For example, if you create the Super Sun Seeds case study example using AFP Toolbox, you can position rules and fixed or variable depth boxes depending on whether the customer's last order falls on the page or whether the customer has one or many orders. You also can use a variety of typographic fonts and include graphics or special messages at different points on each page according to the characteristics of the data. Each statement is designed to effectively communicate the information needed by the individual customer.

AFP Toolbox also lets the programmer add indexing tags to the output document. These tags can be used by the AFP Viewer to navigate through the document and locate specific pages or groups of pages. The indexing tags also can be used by other applications, such as archival and retrieval applications and the Infoprint Server AFP to PDF transform.

With AFP Toolbox, programmers familiar with the ILE C, COBOL, or RPG programming languages can invoke powerful AFP functions using familiar data structures, function calls, and objects that are easy to learn and understand. These functions let the programmer specify and change such AFP formatting options such as:

- Text position and orientation
- Fonts used for printing lines and fields
- Rules and boxes of varying sizes and positions
- Color for rules, boxes, and fonts
- Page segments, overlays, images, graphics, and bar codes
- Centering and aligning lines of text using typographic fonts
- Flowing text into a paragraph, which can be left-aligned, right-aligned, centered, or fully justified.

- Querying the width of a string of text before it is actually placed on the page to help the application make decisions about when to break a line or page
- Underlining text
- Creating bar code objects (BCOCA)
- Creating graphic objects (GOCA), including partial circles and ellipses
- Creating tables

AFP Toolbox Document and Page Structure

Applications that use the AFP Toolbox library functions need to follow the AFP hierarchy when building compound documents. Documents are made up of pages and the pages are made up of objects, as shown in Figure 135.

Refer to *AFP Toolbox for Multiple Operating Systems: User's Guide* for a list showing a typical sequence of calls for an application using the AFP Toolbox Library functions.

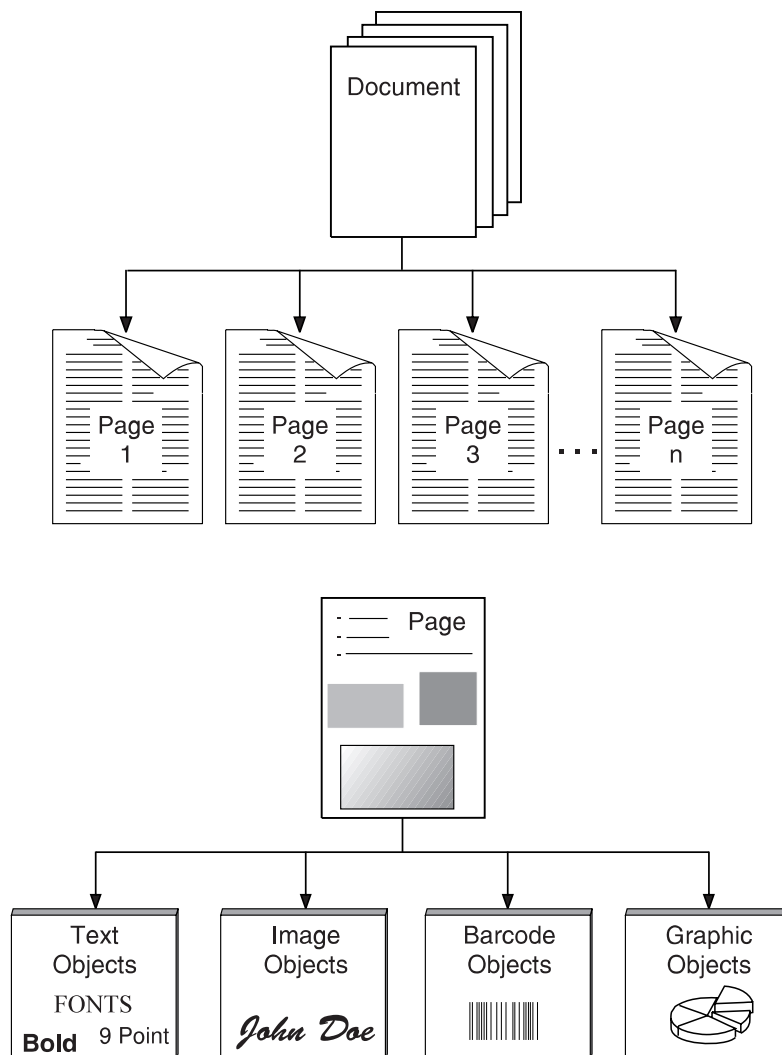


Figure 135. AFP Toolbox Document and Page Structure

As an illustration of creating this structure, we will look at the overall structure of a COBOL program that uses AFP Toolbox subroutines to generate an AFP

application. The program flow is:

<i>Set up general parameters</i>	
Perform AFPINIT	(Initialize)
<i>Set up document parameters</i>	
Perform AFPBDOC	(Begin document)
<i>Set up page parameters</i>	
Perform AFPBPAGE	(Begin page)
<i>Set up fonts to be used</i>	
Perform AFPDFNAT	(Define font)
<i>Set up text to be printed</i>	
Perform AFPNLINE	(Line advance)
Perform AFPWRITE	(Write line)
<i>Set up and perform rest of page</i>	
Perform AFPEPAGE	(End Page)
<i>Repeat for all pages</i>	
Perform AFPEDOC	(End Document)
Perform AFPEND	(End Application)

RPG programs that use AFP Toolbox are coded in a similar manner to COBOL. "Sample RPG Program using AFP Toolbox" on page 343 contains a simple RPG example. Refer to *AFP Toolbox for Multiple Operating Systems: User's Guide* for other examples.

Super Sun Seeds Case Study

Appendix D, "Sample Toolbox Source Code," on page 331 contains the entire COBOL program using AFP Toolbox subroutines to produce the Super Sun Seeds invoice application. This example illustrates many AFP Toolbox document formatting capabilities, including:

- Typographic fonts can be used throughout, because AFP Toolbox supports left, center, and right alignment of text.
- Boxes surrounding transaction information can be drawn dynamically.
- Paragraphs of text can be defined and dynamically formatted. The text is flowed and aligned based on the area of the page allotted for it.
- Page groups (such as each customer invoice) can be numbered with pages "n of m."

Let's look at several segments of the this Super Sun Seeds program. After initialization, the program sets up overall document information (including the output file name) and begins the document:

```
017200000000      MOVE "SUPER SUN SEEDS" TO AFP-DOC-NAME.
017400000000      MOVE "iSeries COBOL Program" TO AFP-DOC-COMMENT.
017600000000      MOVE FILED TO AFP-OUTPUT-TYPE.
017700960322      MOVE "/QSYS.LIB/QAOCL.LIB/QAYTBRESCLS.FILE/SUNSEEDS.MBR"
017800000000      TO AFP-OUTPUT-FILENAME.
018000000000      PERFORM AFPBDOC.
```

Next, the page numbering fields are initialized and the "Begin Page" subroutine is called:

```
028300000000      MOVE "Page 1" TO AFP-PAGE-NAME.
028500000000      PERFORM AFPBPAGE.
```

Various page elements are composed and placed on the page. An overlay comprising the top part of the invoice page (INVHEAD1) is positioned at a 0,0 offset from the page origin and written to the page:

```

030300000000 *-----*
030400000000 * INCLUDE THE OVERLAY *
030500000000 *-----*
030600000000     MOVE 0 TO AFP-X-COORDINATE.
030700000000     MOVE ABSOLUTE-POS TO AFP-X-REF-COORD-SYS.
030800000000     MOVE 0 TO AFP-Y-COORDINATE.
030900000000     MOVE ABSOLUTE-POS TO AFP-Y-REF-COORD-SYS.
031000000000     PERFORM AFPSPPOS.
031100000000
031200000000     MOVE "INVHEAD1" TO AFP-OVLY-NAME.
031300000000     PERFORM AFPIOVL.
031400000000
031500000000     PERFORM PROCESS-THE-ADDRESS.
031600000000
031700000000 *-----*
031800000000 * Start vertical rules under the shaded box *
031900000000 *-----*
032000000000     MOVE 10 TO AFP-RULE-FAT-THICKNESS.
032100000000     MOVE BASELINE-DIR TO AFP-DIRECTION.
032200000000
032300000000     MOVE 793 TO AFP-X-COORDINATE.
032400000000     MOVE ABSOLUTE-POS TO AFP-X-REF-COORD-SYS.
032500000000     MOVE 5442 TO AFP-Y-COORDINATE.
032600000000     MOVE ABSOLUTE-POS TO AFP-Y-REF-COORD-SYS.
032700000000     PERFORM AFPSPPOS.
032800000000     PERFORM AFPBRULE.
032900000000     SET RULEHNDL1 TO AFP-RULE-HANDLE.
033000000000     MOVE 850 TO AFP-X-COORDINATE.
033100000000     PERFORM AFPHMOVE.
033200000000     PERFORM AFPBRULE.
033300000000     SET RULEHNDL2 TO AFP-RULE-HANDLE.
033400000000     MOVE 567 TO AFP-X-COORDINATE.
033500000000     PERFORM AFPHMOVE.
033600000000     PERFORM AFPBRULE.
033700000000     SET RULEHNDL3 TO AFP-RULE-HANDLE.
033800000000     MOVE 1474 TO AFP-X-COORDINATE.
033900000000     PERFORM AFPHMOVE.
034000000000     PERFORM AFPBRULE.
034100000000     SET RULEHNDL4 TO AFP-RULE-HANDLE.
034200000000     MOVE 4505 TO AFP-X-COORDINATE.
034300000000     PERFORM AFPHMOVE.
034400000000     PERFORM AFPBRULE.
034500000000     SET RULEHNDL5 TO AFP-RULE-HANDLE.
034600000000     MOVE 1304 TO AFP-X-COORDINATE.
034700000000     PERFORM AFPHMOVE.
034800000000     PERFORM AFPBRULE.
034900000000     SET RULEHNDL6 TO AFP-RULE-HANDLE.
035000000000     MOVE 1500 TO AFP-X-COORDINATE.
035100000000     PERFORM AFPHMOVE.
035200000000     PERFORM AFPBRULE.
035300000000     SET RULEHNDL7 TO AFP-RULE-HANDLE.

```

The program reads data from the input file and begins filling in the page. The customer name and address is written using a series of new lines (AFPINLINE) and writes (AFPWRITE). This text is left aligned:

```

048400000000     PERFORM AFPINLINE.
048500000000     CALL "TRIM" USING CUST-NAME,
048600000000         BY CONTENT LENGTH OF CUST-NAME,
048700000000         BY REFERENCE AFP-CHARACTER-STRING,
048800000000         AFP-STRING-LENGTH.
048900000000
049000000000     ADD 1 TO AFP-STRING-LENGTH.

```

```

049300000000      MOVE LEFT-ALIGN TO AFP-ALIGNMENT-OPTION.
049400000000      MOVE TRU TO AFP-POSITION-OPTION.
049500000000      PERFORM AFPWRITE.
049600000000
049700000000
049800000000      *-----*
049900000000      * Write the customer address.                                *
050000000000      *-----*
050100000000      PERFORM AFPNLIN.
050200000000
050300000000      CALL "TRIM" USING CUST-ST-ADDR,
050400000000          BY CONTENT LENGTH OF CUST-ST-ADDR,
050500000000          BY REFERENCE AFP-CHARACTER-STRING,
050600000000          AFP-STRING-LENGTH.
050700000000
051000000000      ADD 1 TO AFP-STRING-LENGTH.
051100000000      MOVE LEFT-ALIGN TO AFP-ALIGNMENT-OPTION.
051200000000      MOVE TRU TO AFP-POSITION-OPTION.
051300000000      PERFORM AFPWRITE.
051400000000      PERFORM AFPNLIN.
051500000000
051600000000      CALL "TRIM" USING CUST-CITY-STATE,
051700000000          BY CONTENT LENGTH OF CUST-CITY-STATE,
051800000000          BY REFERENCE AFP-CHARACTER-STRING,
051900000000          AFP-STRING-LENGTH.
052000000000
052100000000      ADD 1 TO AFP-STRING-LENGTH.
052300000000      MOVE LEFT-ALIGN TO AFP-ALIGNMENT-OPTION.
052400000000      MOVE TRU TO AFP-POSITION-OPTION.
052500000000      PERFORM AFPWRITE.
052600000000      PERFORM AFPNLIN.
052700000000

```

Other invoice data, such as account number and invoice number is positioned and right-aligned on the page:

```

038000000000      MOVE 2254 TO AFP-X-COORDINATE.
038100000000      MOVE ABSOLUTE-POS TO AFP-X-REF-COORD-SYS.
038200000000      MOVE 4335 TO AFP-Y-COORDINATE.
038300000000      MOVE ABSOLUTE-POS TO AFP-Y-REF-COORD-SYS.
038400000000      PERFORM AFPSPPOS.
038500000000
038600000000      MOVE ACCOUNT-NUM-IN TO ACCOUNT-NUM-OUT.
038700000000      MOVE 4 TO AFP-STRING-LENGTH.
038800000000      MOVE ACCOUNT-NUM-OUT TO AFP-CHARACTER-STRING.
039000000000      MOVE RIGHT-ALIGN TO AFP-ALIGNMENT-OPTION.
039100000000      MOVE FALS TO AFP-POSITION-OPTION.
039200000000      PERFORM AFPWRITE.
039300000000
039400000000      MOVE 3121 TO AFP-X-COORDINATE.
039500000000      PERFORM AFPHMOVE.
039600000000
039700000000      MOVE INVOICE-NUM-IN TO INVOICE-NUM-OUT.
039800000000      MOVE 8 TO AFP-STRING-LENGTH.
039900000000      MOVE INVOICE-NUM-OUT TO AFP-CHARACTER-STRING.
040100000000      MOVE RIGHT-ALIGN TO AFP-ALIGNMENT-OPTION.
040200000000      MOVE TRU TO AFP-POSITION-OPTION.
040300000000      PERFORM AFPWRITE.

```

As the invoice transaction data is placed down the page, the "frame" of lines surrounding that information is progressively extended down the page:

```

065300000000      MOVE TRANS-UOM TO AFP-CHARACTER-STRING.
065400000000      MOVE 2 TO AFP-STRING-LENGTH.
065600000000      MOVE CENTER-ALIGN TO AFP-ALIGNMENT-OPTION.
065700000000      MOVE TRU TO AFP-POSITION-OPTION.
065800000000      PERFORM AFPWRITE.

```

One of AFP Toolbox's key capabilities is being able to flow text like a word processing program. The space for a paragraph is defined and the text is positioned and aligned within that space as shown in the program statement below:

```

057900000000 *-----*
058000000000 * If there is room, build a personalized paragraph. *
058100000000 *-----*
058200000000 PERFORM AFPQUERY.
058300960322 IF AFP-Y-COORDINATE < 8000
058400000000
058500960322 MOVE 4040 TO AFP-PARA-WIDTH
058600960322 MOVE JUSTIFY-ALIGN TO AFP-FORMAT-OPTION
058700960322 MOVE DEFAULT-LSP TO AFP-LINE-SPACING
058800960322 MOVE 10000 TO AFP-PARA-MAXDEPTH
058900960322 PERFORM AFPBPARA
059000000000
059100960322 MOVE LOW-VALUES TO AFP-CHARACTER-STRING
059200960322 STRING AFP-STRING-1
059300960322 DELIMITED BY SIZE INTO AFP-CHARACTER-STRING
059400960322 CALL "STRING-LENGTH" USING AFP-CHARACTER-STRING,
059500000000 BY CONTENT LENGTH OF AFP-CHARACTER-STRING,
059600960322 BY REFERENCE AFP-STRING-LENGTH
059700960322 ADD 1 TO AFP-STRING-LENGTH
060000960322 PERFORM AFPPTTEXT
060100000000
060200960322 CALL "TRIM" USING CUST-NAME,
060300000000 BY CONTENT LENGTH OF CUST-NAME,
060400000000 BY REFERENCE AFP-CHARACTER-STRING,
060500960322 AFP-STRING-LENGTH
060600000000
060700960322 ADD 1 TO AFP-STRING-LENGTH
061000960322 PERFORM AFPPTTEXT
061100000000
061200960322 MOVE LOW-VALUES TO AFP-CHARACTER-STRING
061300960322 STRING AFP-STRING-2
061400960322 DELIMITED BY SIZE INTO AFP-CHARACTER-STRING
061500960322 CALL "STRING-LENGTH" USING AFP-CHARACTER-STRING,
061600960322 BY CONTENT LENGTH OF AFP-CHARACTER-STRING,
061700960322 BY REFERENCE AFP-STRING-LENGTH
061800960322 ADD 1 TO AFP-STRING-LENGTH
062100000000
062200960322 PERFORM AFPPTTEXT
062300960322 PERFORM AFPEPARA
062400960322 END-IF.

```

As you can see with this simple example, using AFP Toolbox is a much lower-level page interface than DDS. It should be used when other tools such as Infoprint Designer or DDS cannot provide the specific capability required.

Chapter 19. Using the Print Format Utility

Print Format Utility, hereafter referred to as PFU, is part of the IBM AFP Utilities (Product Number 5722-AF1). This chapter introduces PFU and describes how to use it.

Introduction to PFU

PFU lets you format data directly from iSeries database files. It is analogous to a document version of the Query product. PFU is well-suited to either quick applications or applications requiring multiple-up pages (such as labels) with bar coding. The forms and formats for printing are designed by defining a printout format definition (PFD definition) instead of writing your own application programs.

With PFU, you can:

- Design a record layout interactively.
You can define headings, boxes, or logos that contain text, lines, images, bar codes, and graphics to be printed in addition to the data in the database file member.
- Design the page layout interactively.
You can define headings, boxes, or logos that contain text, lines, boxes, images, bar codes, and graphics to be printed.
- Save the record layout and page layout as a printout format definition in your file.
- Print a database file member according to the printout format definition.
- Print data summary information in the database member, such as the total, average, or maximum.

PFU also lets you create various kinds of output from one database file member. For example, you can print a list of products, product descriptions, or even delivery labels from one database file member using different printout format definitions.

You can use element selection to specify selective printing of any element in a record layout. Selection is determined by the values of one to five variable fields in the database file being printed. To use the function, press F9=Element selection on any of the following screens and specify the conditions under which the element should be printed:

- Define text
- Change text
- Define line
- Change line
- Define box
- Change box
- Define bar code
- Change bar code
- Place graphics
- Change graphics
- Place page segment
- Change page segment

Super Sun Seeds Packing List Labels

PFU is best suited to multiple-up label applications, so labels with bar codes that support the Super Sun Seeds packing list will be the task at hand. This PFU application creates a label for each line item in a Super Sun Seeds customer order. Then labels are peeled off when the order is filled and the bar code is scanned to confirm the item selection. The labels are printed 3-up and up to 30 per 8 1/2 by 11 inch page. A sample of the labels is shown below:

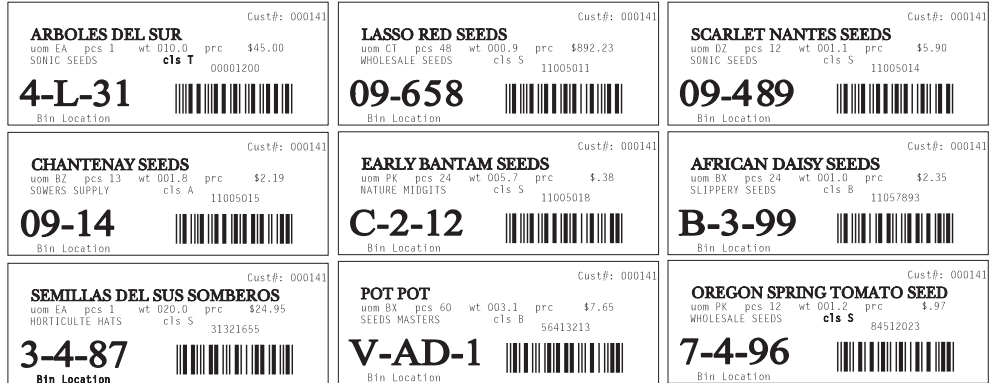


Figure 136. Packing List Label Printout

Since PFU works directly with the iSeries database, we have created an expanded customer invoice file that joins the invoice file (SEEDDETL) and the item master file (SEEDITEM). This lets both invoice and item master information to be passed to PFU with each record processed. The specification for this joined file (SEEDJOIN) is shown below:

```

SOURCE FILE . . . . . SAMPLER/QDDSSRC
MEMBER . . . . . SEEDJOIN
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
100      A* Super Sun Seeds Invoice/Item Join LF
200      A          R ITMALL          JFILE(SEEDDETP SEEDITEM)
300      A          J                JOIN(SEEDDETP SEEDITEM)
400      A                                JFLD(ITM1 ITEM#)
500      A* SEEDDETP FIELDS
600      A          RECNO          4S 0    COLHDG('Record' 'Number')
700      A                                TEXT('Record Number')
800      A          DELETE          1A      COLHDG('Delete')
900      A                                TEXT('Delete Code (D)')
1000     A          CUST#           6S 0    COLHDG('Customer' 'Number')
1100     A                                TEXT('Customer Number')
1200     A          STNAME          25A     COLHDG('SHIP-TO NAME')
1300     A                                TEXT('SHIP-TO-NAME')
1400     A          STSTRT          25A     COLHDG('SHIP STREET')
1500     A                                TEXT('SHIP STREET')
1600     A          STCITY          25A     COLHDG('SHIP CITY')
1700     A                                TEXT('SHIP CITY')
1800     A          STSTE           2A      COLHDG('ST')
1900     A                                TEXT('STATE ADDR')
2000     A          STZIP           9S      COLHDG('ZIP')
2100     A                                TEXT('ZIP+4')
2200     A                                EDTWRD(' - ')
2300     A          SHPVIA          10A     COLHDG('SHIP VIA')
2400     A                                TEXT('SHIP VIA')
2500     A          TERMS           10A     COLHDG('TERMS ')
2600     A                                TEXT('TERMS ')
2700     A          QTY1            4S 0    COLHDG('QTY01')
2800     A                                TEXT('QTY01')
  
```

```

2900      A          ITM1          8S 0      COLHDG('ITM01')
3000      A                                     TEXT('ITM01')
3100      A* SEEDITEM FIELDS
3200      A          ITMDEL        1A          COLHDG('DELETE CODE')
3300      A                                     TEXT('DELETE CODE')
3400      A          ITMDES        25A         COLHDG('ITEM DESCRIPTION')
3500      A                                     TEXT('ITEM DESCRIPTION')
3600      A          ITMCLS        1A          COLHDG('ITEM 'CLASS')
3700      A                                     TEXT('ITEM CLASS')
3800      A          SHPCDE        1A          COLHDG('SHIP 'CODE')
3900      A                                     TEXT('SHIP CODE')
4000      A          UOM           2A          COLHDG('UNIT OF' 'MEAS')
4100      A                                     TEXT('UNIT OF MEASURE')
4200      A          PCSUOM        2A          COLHDG('PCS/' 'UOM')
4300      A                                     TEXT('PIECES/UOM')
4400      A          WHSLOC        6A          COLHDG('WHSE' 'LOC')
4500      A                                     TEXT('WAREHSE LOC')
4600      A          WEIGHT        4S 1        COLHDG('#WEIGHT')
4700      A                                     TEXT('#WEIGHT')
4800      A          COST          6S 2        COLHDG('COST')
4900      A                                     TEXT('ITEM COST')
5000      A          SELPRC        6S 2        COLHDG('SELL' 'PRICE')
5100      A                                     TEXT('SELLING PRICE')
5200      A                                     EDTWRD(' , $0. ')
5300      A          UTSLY         6S 0        COLHDG('UNITS' 'LST YR')
5400      A                                     TEXT('UNITS LAST YEAR')
5500      A          $YTDLY        7S 2        COLHDG('SALES' 'LAST YEAR')
5600      A                                     TEXT('SALES LAST YEAR')
5700      A          UTYTD         6S 0        COLHDG('UNITS' 'YTD')
5800      A                                     TEXT('UNITS YR-TO-DTE')
5900      A          $YTD          7S 2        COLHDG('SALES' 'YTD')
6000      A                                     TEXT('SALES YR-TO-DATE')
6100      A          SUPLR1        15A         COLHDG('SUPPLIER 1')
6200      A                                     TEXT('SUPPLIER 1')
6300      A          SUPLR2        15A         COLHDG('SUPPLIER 2')
6400      A                                     TEXT('SUPPLIER 2')
6500      A          DTESAL        6S          COLHDG('LSTSAL')
6600      A                                     TEXT('LAST SALE')
6700      A          DTESHP        6S          COLHDG('LSTSHP')
6800      A                                     TEXT('LAST SHIP')
6900      A          K CUST#
7000      A          K ITM1

```

* * * * E N D O F S O U R C E * * * *

PFU builds a print format definition that defines how information from the database file is merged with other output elements (such as text, page segments, overlays, and bar codes). PFU defines a record layout and a page layout. The record layout defines the composition of one database record. In this example, each line item in an invoice is a record. The page layout defines how one or more record layouts are positioned on the page. This can be done through a one-to-one relationship, or as in this case, we can place up to 30 records or labels on a page.

To create the labels, we first start AFP Utilities:

```

AFPUI      IBM Advanced Function Printing Utilities for iSeries

Select one of the following:

Overlay Utility
  1. Work with source overlays
  2. Work with source overlay files

Print Format Utility
  11. Work with PFD definitions
  12. Work with PFD definition files
  13. Print database file member
  14. Print AFP Utilities tutorial

Resource Management Utility
  21. Convert to page segment
  22. Work with overlays
  23. Work with page segments

Selection or command
===> 11
F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel  F16=System main menu

```

Figure 137. AFP Utilities Main Menu

Select option 11 to build a print definition. This display opens:

```

                                Work with PFD Definitions

File . . . . . QPFDDEF      Name, F4 for list
Library . . . . . SAMPLER   Name, *LIBL, *CURLIB
PFD definition . . . . . *ALL  Name, generic*, *ALL
Position to . . . . .      Starting characters

Type options, press Enter.
  1=Create  2=Change  3=Copy  4=Delete  6=Print PFD definition  7=Rename
  9=Print database file

      PFD
Opt Definition Text                               Changed
  BILL1
  2  ITEMLBL   Order Labels for Super Sun Seeds   03/01/96
      SHLFLBL   Shelf Label for Seed Items       03/01/96
                                           Bottom

Parameters or command
===>
F3=Exit  F4=Prompt  F5=Refresh  F9=Retrieve  F11=Display names only
F12=Cancel

```

Figure 138. Work with PFD Definitions Display

Select option 2 to change a definition called "ITEMLBL" for the Super Sun Seeds item labels.


```

Change PFD Definition

File . . . . . : QPFDDEF
Library . . . . . : SAMPLER
PFD definition . . . . . : ITEMLBL

Type options, press Enter.
1=Select

Opt   Action
      Define PFD specifications
      Work with PFD definition fonts
1     Specify database file
1     Specify break fields
1     Design record layout
1     Design page layout
      Specify record selection
      Define printout specifications
      Specify mapping object name

F3=Exit  F5=Refresh  F9=Select all  F12=Cancel

```

Figure 139. Change PFD Definition Display

The main selection screen shows the options that can be used to build the application. We have selected the options to specify the database file, define break fields, design record layout, and design page layout. Pressing Enter brings up the display for the first option, Specify Database File:

```

Specify Database File

Type choices, press Enter.
Database file . . . . . SEEDJOIN      Name, F4 for list
Library . . . . . SAMPLER           Name, *LIBL, *CURLIB
Record format . . . . . ITMALL      Name, *FIRST, F4 for list

F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel

```

Figure 140. Specify Database File Display

Select SEEDJOIN as the target database file for this application.

The Design Record Layout display shows the layout of a record (a label) in screen view mode. Design elements include text, bar codes, lines, boxes, page segments, and database fields. Fields from the database are specified by using an ampersand (“&”) prefix and “.” suffix. For example, the specification “&WEIGHT.” is the field WEIGHT.


```

                                Design Record Layout
Control . . . . . PFD Definition . . . . . ITEM LBL
NBR ID NAME M ACROSS DOWN Parameters
001 B      2  0.000  0.000  END ACROSS= 2.660  END DOWN  = 1.000
002 T      1    3      2  FONT = 0  &ITMDES.
003 C      2  1.400  0.660  &ITM1.
004 T      2  0.100  0.850  FONT=0  &WHSLOC.
005 T      2  0.200  0.400  FONT=0  uom &UOM. pcs &PCSUOM. wt &WEIGHT. prc
006 T      2  1.700  0.570  FONT=0  &ITM1.
007 T      2  0.200  0.500  FONT=0  &SUPLR1;
008 T      2  0.250  0.980  FONT=0  Bin Location
009 T      2  1.300  0.500  FONT=0  c\&ITMCLS.
010 T      2  2.000  0.150  FONT=0  Cust#: &CUST.#;
011
012
013
014
015
016
                                More...

F3=Exit      F6=Text      F9=Line      F10=Box
F11=Bar code F13=Place   F14=Change  F24=More keys

```

Figure 143. Design Record Layout Display

With the “Design Page Layout” display, you can place one or more copies of the record layout on the page. Each record layout is filled with variable data from a record in the selected database file. With the Super Sun Seeds item labels, we want to use 8-1/2 by 11 inch label stock and position labels three across and ten deep. If we select the first record layout, we see the record positioning:

```

                                Design Page Layout          Columns:  1- 74
Control . . . . . PFD Definition . . . . . ITEM LBL
*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...
001 *R001-001          *R001-002          *R001-003
002
003
004
005
006
007 *R001-004          *R001-005          *R001-006
008
009
010
011
012
013 *R001-007          *R001-008          *R001-009
014
015
016
017
                                More...

F3=Exit      F6=Text      F9=Line      F10=Box
F11=Bar code F21=Element edit F22=Block edit F24=More keys

```

Figure 144. Design Page Layout Display

This display shows that the record layout is replicated three across and ten down.

Part 3. Working with AFP Output

Part I introduced you to the architecture of output on iSeries. In Part II, we examined a number of options to develop output applications. We used a standard case study, Super Sun Seeds.com invoicing, with each tool. In Part III, we will assume that some output-enabling approach has been adopted and that a document or report has been generated onto an iSeries output queue. Part III focuses on the options to deliver that document or report to the “consumer” of the information.

Prior to the availability of the Internet, most documents and reports flowed within an enterprise as hard copy pages. Computing platforms such as OS/400 provided a set of print services that enabled the printing of these documents and reports to actual physical printers. Printing is certainly not going away, but the Internet and enterprise intranets offer a path to more timely and efficient delivery of the information. iSeries enterprises are reengineering or deploying e-business models to reduce costs, improve efficiency, and remain competitive. Generally, if a business process is made an e-business process, then the communications (documents and reports) associated with that process should also be e-business-enabled. For example, converting order entry from a batch to a web-based process is not really complete until the principal output of order entry - an order confirmation document - is also converted. Generally, this means that the output is delivered electronically.

Part III looks at the “back-end” of the iSeries output architecture, discussing options for the electronic delivery of documents and reports. This includes fax, outbound electronic delivery by e-mail, and inbound electronic delivery by client or browser access to the output.

Chapter 20. Viewing Files

The AFP Viewer provides the ability to resolve and view AFP documents.² It can also view other printed formats, such as SCS and ASCII, as well as a number of image file formats. From the viewer, a document or image can be annotated, reprinted, or faxed. There are two versions of the AFP Viewer: the AFP Workbench Viewer and the AFP Viewer Plug-in. The AFP Workbench Viewer is integrated into iSeries Access (specifically, the iSeries Navigator component). You can download the AFP Viewer Plug-in free from IBM at: from <http://www.printers.ibm.com/R5PSC.NSF/web/afpviewer>.

Benefits of Viewing Your Output

The ability to view a fully composed AFP document on a PC screen offers many advantages:

- Increased end-user productivity
 - Lets you proof your output before printing it
 - Lets you view information online instead of printing it
 - Indexed documents can be quickly navigated
- Reduced paper use
 - Saves money by purchasing less paper
 - Is environmentally friendly
- Enhanced customer service
 - Allows a customer service representative to immediately access a customer statement online
 - Provides a view of the complete customer statement just as it was printed
 - Indexing statements helps customer service representative easily locate a specific statement

Many OS/400 applications take advantage of the AFP Viewer so their output can easily be viewed by users at intelligent workstations.

- iSeries Access includes the AFP Viewer. It also provides all users with a graphical interface that shows them a list of OS/400 spooled files. Double-clicking on any spooled file in the list invokes the AFP Viewer, and brings up an image of that document on the PC display. As you will see in Chapter 22, “Working with Infoprint Server,” on page 245, this graphical interface to your spooled files can also provide interactive access to PDF and e-mail functions
- Facsimile Support for iSeries uses the AFP Viewer to display incoming fax documents.
- Content Manager OnDemand for iSeries (OnDemand) supports the archival and retrieval of a wide variety of data types, including AFP. OnDemand can be used to selectively search for and then display, print, or fax documents that have been retrieved.

OnDemand provides hierarchical storage management across magnetic, optical, and tape devices. The OnDemand server allows indexes, such as customer name or account number, to be created and stored. This gives the user quick and simple navigation through thousands of statements.

- ImagePlus, a Content Management component, generates image documents on the OS/400. These documents can be viewed using the AFP Viewer.

2. BCOCA is not supported by the AFP Viewer

- Content Manager OnDemand for AIX provides archival and retrieval of AFP formatted documents, as well as line-mode and image documents, much like OnDemand for iSeries.

iSeries Access Viewer Details

AFP Viewer functions are available to all iSeries Access clients.

You can view iSeries spooled files, as well as documents in shared folders, on a LAN, or on a PC. The standard PC image file types that are also supported are TIFF, PCX, DCX, DIB, and GIF. Also, you can print the document on a locally attached printer, clip portions of a document to create new documents, use the IBM AFP Printer Driver for Windows to create AFP documents from PC applications, and annotate those documents.

The following functions are part of the AFP Viewer within iSeries Navigator:

- Client management of iSeries spooled files
- View AFP, SCS, and ASCII files
- View TIFF, PCX, DCX, and DIB image files
- Attach notes (annotation) to a displayed document
- Copy selected pages to a file
- Create iSeries overlays and page segments from any Windows application
- Print viewed documents or images
- Search a spooled file for target keywords
- Select and copy data to clipboard
- Print (or any print driver operation, such as fax)
- View controls, such as zoom, page rotation, and so on

AFP External Resources

A typical AFP document is “architected” with references to external page elements such as fonts, overlays, and page segments. The iSeries Access AFP Viewer retrieves those page elements and displays the entire document, including these external elements.

Overlays and Page Segments

Viewing AS/400 Spooled Files with Client Access/400

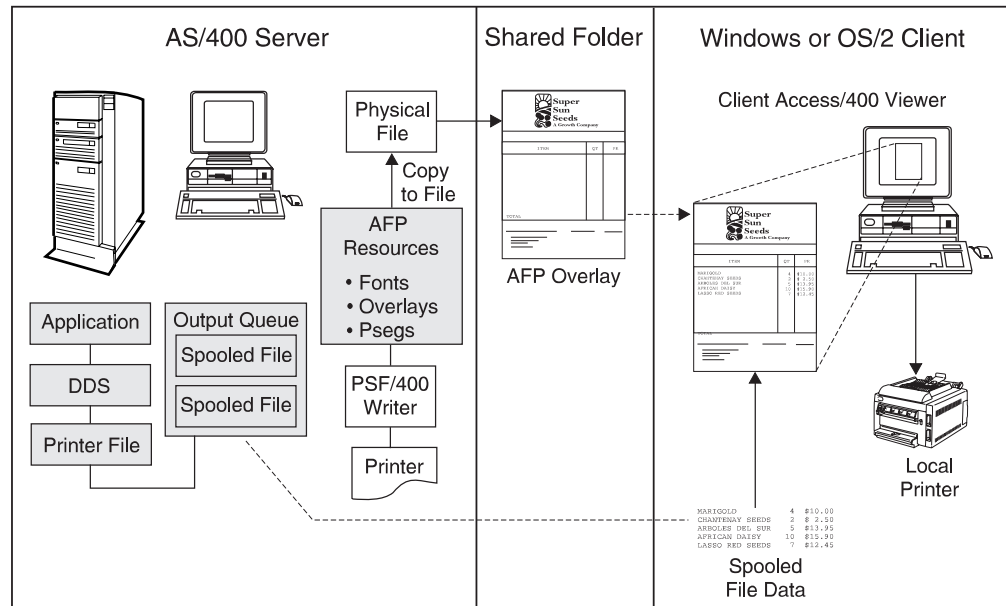


Figure 146. Viewing OS/400 Output Using iSeries Access

In order to understand this process, consider the case study of the Super Sun Seeds Invoice. When that job is a spooled file in an output queue, it contains references to page segments (Super Sun Seeds logo, and others like the strawberry, tree, and flower). It also contains references to several different AFP overlays (the different variations of the invoice form). These resources are not “inline” with the spooled file; they are simply referenced by the spooled file. If this job is released to an IPDS, AFP(*YES) printer, PSF gathers all the resources and sends them to the printer along with the spooled file.

Fonts

The AFP Viewer uses PC-resident fonts to display documents. This means that the font used in the iSeries document will be matched as closely as possible with a font available on the PC when the document is displayed with the AFP Viewer.

By default, the PC uses TrueType fonts, which are included with Windows. If the PC also has Adobe Type Manager (ATM) fonts installed and active, the AFP Viewer uses these fonts to display documents. ATM must be purchased separately from the AFP Viewer.

For best fidelity when viewing AFP documents, you should use IBM fonts that have Type 1 equivalents (such as the outline fonts that come with AFP Font Collection).

Viewing Spooled Files with iSeries Access

Follow these steps to use iSeries Access to view spooled files on your PC.

1. Double-click the iSeries Access icon. This opens the iSeries Access Window.

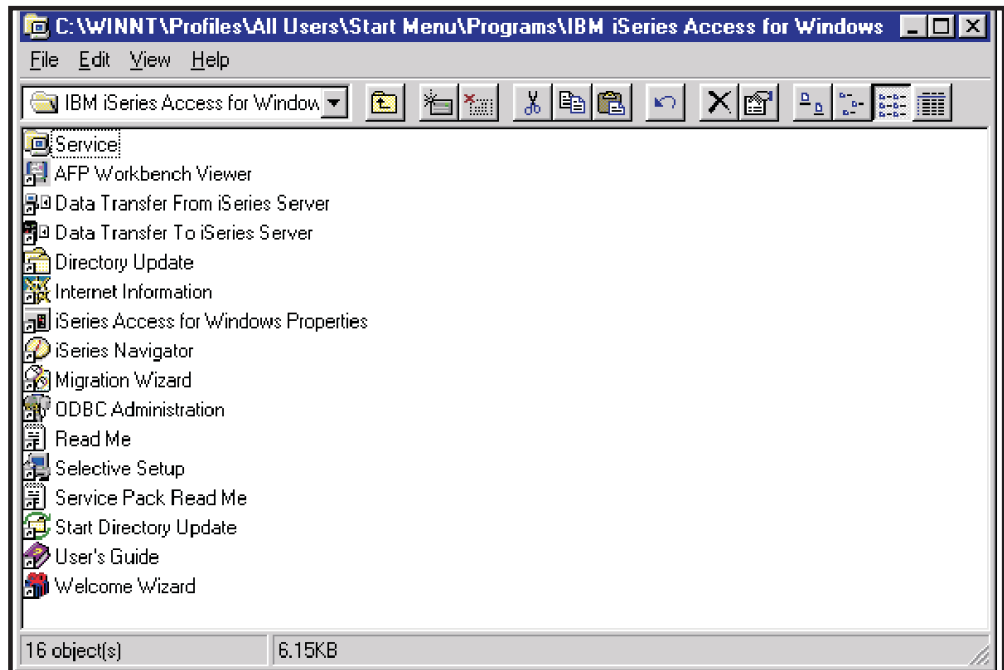


Figure 147. The iSeries Access Window

2. Double-click the iSeries Navigator icon. The iSeries Navigator window opens:

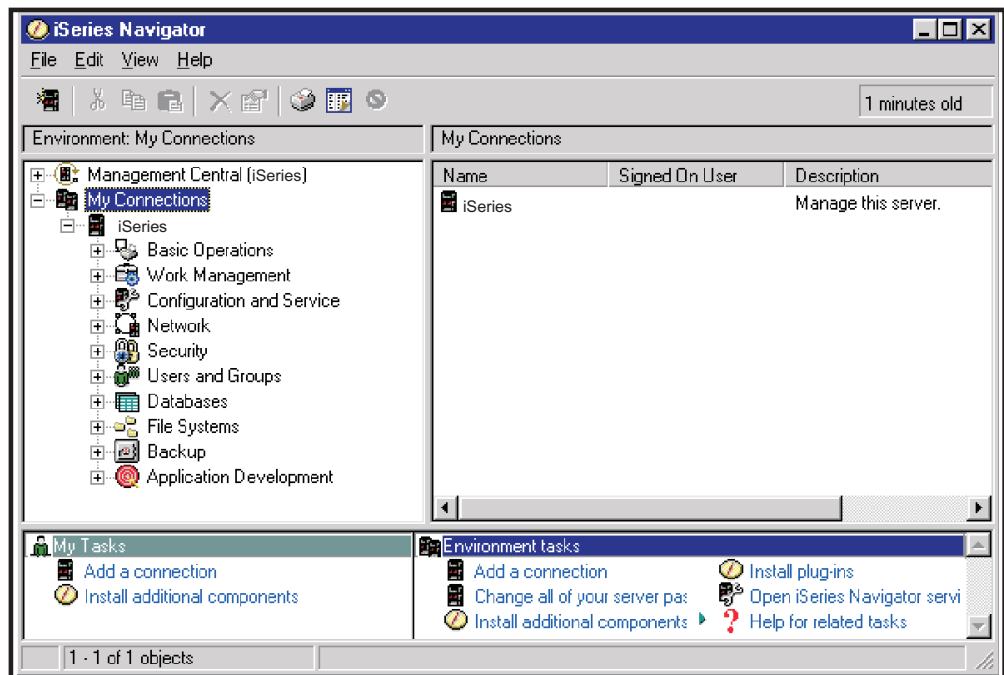


Figure 148. iSeries Navigator Window

3. Expand **Basic Operations**.
4. Open **Printer Output**.
5. From the You can customize the information that appears in the Output List and which printer output you want to list, by User, Printer, Output Queue, and so on. Refer to the online help for instructions.

6. Once you have the Output List organized to your liking, double-click on any spooled file in the list. This automatically invokes the AFP Workbench for Windows Viewer, which lets you do the following:
 - See your spooled file on the display (see Figure 149).
 - Use the AFP Viewer functions such as zooming and rotating.
 - Print the viewed spooled file to a locally attached printer (if function is available).

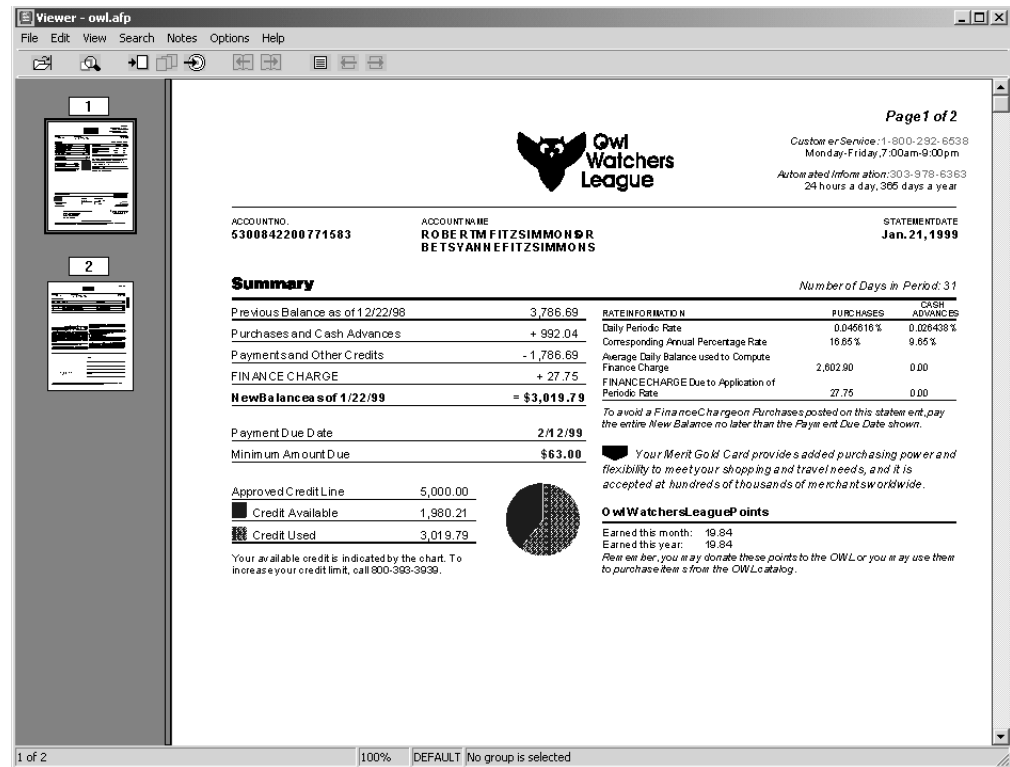


Figure 149. Spooled File Displayed


Viewing OS/400 Stream Files with iSeries Access

To view a stream file (it must be an appropriate type, such as AFP, SCS, and ASCII) in the integrated file system on the OS/400, you must first get the file to your workstation.

To copy a file from the Integrated File System to your workstation, follow these steps:

1. Double-click the iSeries Access icon.
2. Double-click the iSeries Navigator icon.
3. Open a connection to an OS/400.
4. Expand **File Systems** then **Integrated File System**. Continue to expand until the stream file you want to copy is visible.
5. Right-click on the stream file, and select **Copy**.
6. On your workstation, in the directory you want to store the stream file, right-click and select **Paste**.

To view an AFP stream file on your PC with the AFP Viewer, follow these steps:

1. Double-click the iSeries Access shortcut.
2. Move the AFP Workbench Viewer icon to your desktop (optional).
3. Double click the **AFP Workbench Viewer** icon.
4. Click **File** and select **Open**.
5. Specify the stream file to open.
6. If you indexed the stream file, certain pages are grouped together. To select a group to view, click the group icon: 

Viewing Spooled Files with iSeries Access for Web

V5R2 iSeries Access for Web (5722 XH2) supports viewing spooled files in PDF format. The functions described below are all available if Infoprint Server is installed on the iSeries.

To view a spooled file in PDF, follow these steps:

1. Start an iSeries Access for Web session. The address is in the format `http://myiseries/webaccess/iWAHome`, where *myiseries* is your system's name.
2. Click **Printer Output**. This displays your spooled files.
3. To view a spooled file in PDF format, click the PDF link for the desired spooled file. This brings up a PDF configuration form.
4. For **Destination**, specify **Browser**.
5. The spooled file displays in your browser with the Acrobat plug-in:

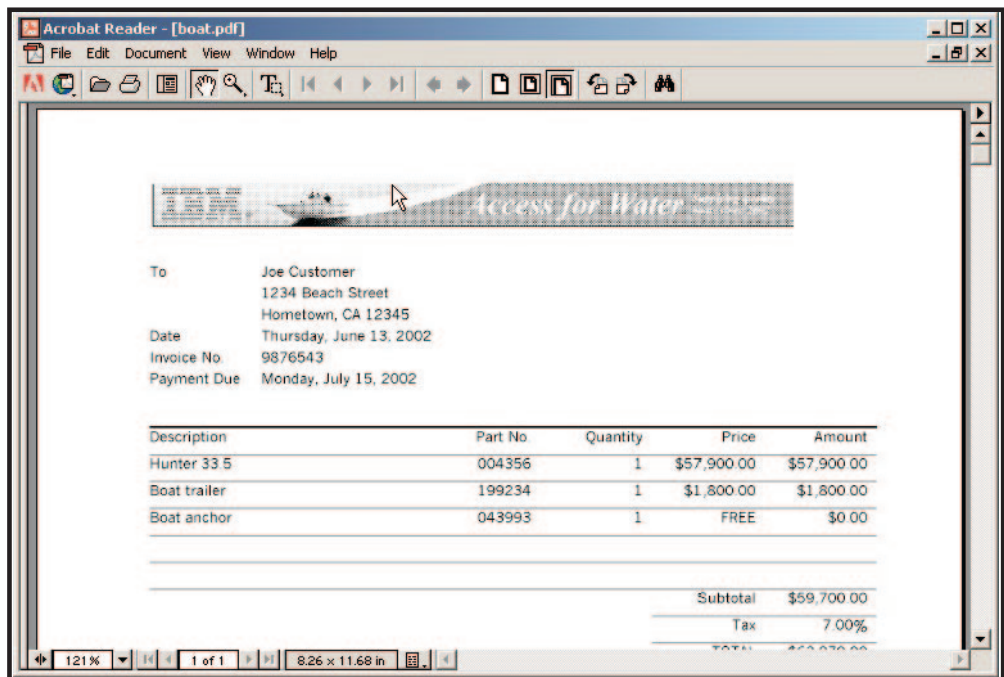


Figure 150. File Displayed in Acrobat Reader

Viewing AFP Files with the AFP Viewer Plug-In

IBM provides a free version of the AFP Workbench Viewer, called the AFP Viewer Plug-In. It is a plug-in for Web browsers (such as Internet Explorer and Netscape Navigator) on Windows systems. To download the viewer, follow the instructions on <http://www.printers.ibm.com/R5PSC.NSF/web/afpviewer>. To view an AFP file on your workstation with the AFP Viewer Plug-In, follow these steps:

1. On your workstation, navigate to the file you want to view. Double-click the file.
2. If the Open With window opens, select the browser to which you have the AFP Viewer Plug-In installed.
3. The file displays in the specified browser.

Calling the AFP Viewer Directly from Your Application

The AFP Viewer can also be called directly from your application program. You can use the Start PC Command (STRPCCMD) to invoke the Viewer program (CWBVIEWR). The program needs to be passed the complete ID of the target spooled file.

Chapter 21. Sending Your OS/400 Output by Fax

Facsimile Support for iSeries provides comprehensive facilities to manage fax within your organization. It can handle the receiving, managing, viewing, and printing of your inbound faxes. It can send any *SCS or *AFPDS spooled file on your iSeries outbound, including setting up and managing the transmission. It can also send data from a PC client that is connected to the iSeries server by iSeries Access and using the IBM AFP Facsimile Support driver. This PC client can be a Windows NT, Windows 2000, Windows 95, or Windows 3.1 client. If you have Infoprint Server installed, you can use it to respool any standard iSeries spooled file as an *AFPDS spooled file. You can then use Facsimile Support for iSeries to fax the *AFPDS spooled file.

Super Sun Seeds Case Study

Facsimile Support for iSeries works well with the Super Sun Seeds case study by providing the ability to fax a copy of the Super Sun Seeds invoice to a customer. Assuming that the information being faxed, an invoice or set of invoices for a specific customer, resides in an individual spooled file. Faxing is done most easily by providing a reprint program that selects and reprints specific invoices.

This section shows how the invoice can be faxed.

From the Facsimile Support for iSeries main menu, select option 5, "Outbound Fax". The Outbound Fax Commands Display appears.

```
FAXOUT                               Outbound Fax Commands                               System:  BLDSYS1
Select one of the following:
    1. Work with Fax Output Queue
    2. Work with Fax File Attributes
    3. Change Fax File Attributes
    4. Delete Fax File
    5. Send Fax

Enhanced Services:
    21. Create Fax Note
    22. Check Fax Status
    23. Print Fax Status
    24. Submit Fax

Selection or command
===> 1
F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel  F13=Information Assistant
F16=System main menu
```

Figure 151. Outbound Fax Commands

Select option "1" to work with a fax output queue.

You are prompted for output queue information. Select output queue SUPER in library QGPL.

```

Work with Fax Output Queue (WRKFAXQ)

Type choices, press Enter.

Output queue . . . . . super      Name, *SNDQ, *SAVQ, *ERRQ
Library . . . . . qgp1          Name, *LIBL, *CURLIB
Select files for:
User . . . . . *CURRENT        Name, *CURRENT, *ALL
User data . . . . . *ALL        *ALL, *CURRENT, *TIMED
                                                    Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

Figure 152. Work with Fax Output Queue (WRKFAXQ)

The spooled files in the SUPER output queue are displayed. Select “1” (Send fax) to send the INVNEW2 spooled file as a fax. If you have set up a fax profile that has the environment of 2 = Enhanced, you will see the Submit Fax (SBMFAX) command.

```

Work with Fax Output Queue

Queue:  SUPER      Library:  QGPL

Type options, press Enter.
 1=Send fax  3=Hold  4=Delete  5=Display  6=Release

Opt File          FNbr Job      User      Number  User Data
 1  INVNEW2        1  QPADEV0003 WCSHAFF  104946  INVNEW2
   QSYSPRT        2  QPADEV0003 WCSHAFF  104946
   QSYSPRT        3  QPADEV0003 WCSHAFF  104946
                                                    Bottom

Parameters or command
===>
F3=Exit    F4=Prompt  F5=Refresh  F9=Retrieve  F12=Cancel  F17=Top
F18=Bottom F20=QFAXOPR F22=Printers
(C) COPYRIGHT IBM CORP. 1991, 1995.

```

Figure 153. Work with Fax Output Queue

Specify the fax information, including “send to” information, phone number, and company name. Specify “*YES” in the “Create Cover Page” field to have Facsimile Support for iSeries generate a cover sheet for the fax.

```

Send Fax (SNDFAX)

Type choices, press Enter.

Send to:
Telephone number . . . . . > 3039246300
To line 1 for cover page . . . > 'Improved Printing Corp'
To line 2 for cover page . . . > 'Performance Boulevard'
To line 3 for cover page . . . > PRINTERSVILLE
      + for more values
Spooled file . . . . . > INVNEW2      Name
Job name . . . . . > QPADEV0003      Name, *
User . . . . . > WCSHAFF           Name
Number . . . . . > 104946          000000-999999
Spooled file number . . . . . > 1      1-999999, *ONLY, *LAST
Transmission mode . . . . . *FINE     *FINE, *NORMAL
Create cover page . . . . . > *YES     *NO, *YES
                                                    More...

F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

Figure 154. Send Fax Display

On this continuation display for Send Fax, specify the printer file name for the cover page and the "To" information. The printer file QPFFCVP is an AFP data stream printer file used to support the fax transmission. The Super Sun Seeds fax cover page overlay is shown in Figure 156. To use a cover page overlay, specify its name as the front overlay name in the QPFFCVP printer file.

```

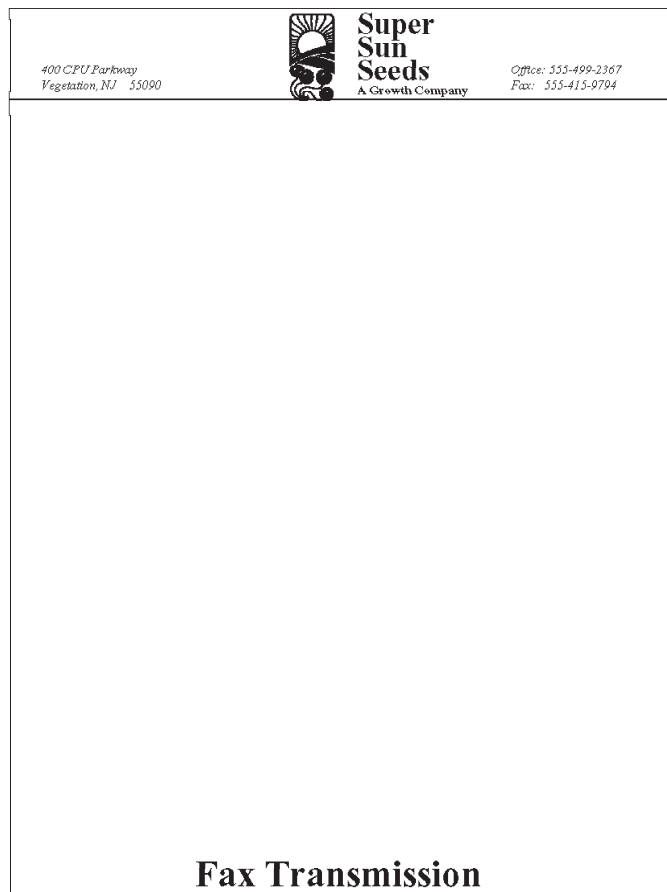
Send Fax (SND FAX)

Type choices, press Enter.

Cover page printer file . . . . QPFFCVP      Name, *DFT
Library . . . . .                Name, *CURLIB
Title line for cover page . . . Invoice Confirmation
From lines:
  From line 1 for cover page . . Super Sun Seeds
  From line 2 for cover page . . *BLANK
  From line 3 for cover page . . *BLANK
Comment for cover page . . . . . Please review
                                                    More...
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

Figure 155. Send Fax Display (continued)



This invoice overlay designed using IBM AFP Utilities/400

Figure 156. Sample Fax Overlay (SSSFAX)

Once the requested fax has been sent, it is listed on the outbound fax queue. The outbound fax (INVNEW2) is listed on this queue.

```

Work with Fax Output Queue

Queue:  QFFSNDFAX      Library:  QUSRSYS

Type options, press Enter.
  2=Change  3=Hold  4=Delete  5=Display  6=Release  8=Attributes

Opt File          FNbr Job          User          Number  User Data
  INVNEW2          1  QPADEV0003  WCSHAFF      104946  *CURRENT

Parameters or command
===>
F3=Exit  F4=Prompt  F5=Refresh  F9=Retrieve  F11=View 2  F12=Cancel
F17=Top   F18=Bottom  F20=QFAXOPR F22=Printers
Bottom

```

Figure 157. Work with Fax Output Queue

When the fax is physically sent, the Super Sun Seeds cover page with the fax information also is sent, followed by the Super Sun Seeds invoice for Improved Printing.

From this display you can also select option “5” to display the spooled file being sent (the invoice). This displays a non-graphical version of the invoice. With iSeries Access viewing support, you can display a completely graphical version of the invoice.

```

Display Spooled File

File . . . . . :  INVNEW2                      Page/Line  1/12
Control . . . . .      Columns  1 - 78
Find . . . . .
*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...+...
      IMPROVED PRINTING CORP
      PERFORMANCE BOULEVARD
      PRINTERSVILLE
      CO 45789-2637
          100          31300          3/09/02          4/09/02
      BEST WAY          3/09/02  NET 30          YOUR PRINTER RE
1 PK  01100517  SPARTAN SEEDS          2.39          2.39
9 PK  04569870  NORTHERN LITE BLUE SPRUCE  858.32      7,724.88
12 BX 11005004  BUSH GREEN SEEDS        2.50          30.00
12 CT 11005011  LASSO RED SEEDS        892.23      10,706.76
26 PK 11005018  EARLY BANTAM SEEDS      .38           9.88
5 BX  11057893  AFRICAN DAISY, SEEDS    2.35          11.75
1 PK  15975365  HEAVY OAK              129.09       129.09
33 BX 32746510  HOPS BREWING LIGHT      1.20          39.60
6 EA  46578913  SEED SURVEYING SITE     50.00        300.00
2 BX  56413213  POT POT                 7.65          15.30
More...

F3=Exit  F12=Cancel  F19=Left  F20=Right  F24=More keys
Bar codes not displayed.

```

Figure 158. Display Spooled File

Chapter 22. Working with Infoprint Server

Infoprint Server significantly expands the possibilities of print and electronic output in an iSeries environment, integrating new capabilities within the existing print framework. This chapter first illustrates how you can use Infoprint Server to accomplish various tasks in “How Can I Use Infoprint Server?” then describes how each Infoprint Server component works in these subsequent sections:

- “Transforming Your OS/400 Output to PDF” on page 254
- “E-Mail Support” on page 256
- “Indexing Output with the Create AFP Data Command” on page 257
- “Transforming ASCII Image Data to AFP” on page 259
- “Printing PCL, PDF, and PostScript Data on IPDS Printers” on page 259

For instructions about using any of these components, refer to *Infoprint Server for iSeries: User's Guide* and *IBM @server iSeries Printing VI: Delivering the Output of e-business*, SG24-6250.

How Can I Use Infoprint Server?

Let's first look at a few application scenarios to see how the functions fit in with various installations. We will examine the current application process and then how Infoprint Server functions could be used to transform that process.

Publish Output for Electronic Access

In this scenario, sales and financial reports are currently printed in hardcopy format and distributed to end users. In many cases, the recipient immediately skips to the last page to get a grand total and then discards the report. Furthermore, because we are working with hardcopy, the output is not timely and the production process of printing incurs operational costs.

Using Infoprint Server's PDF subsystem, the reports can be transformed into PDF format and stored for "inbound" access within the iSeries database, specifically the integrated file system. Recipients would use client PCs or browsers to access their reports. The reports could also be stored in the integrated file system and e-mailed to each recipient. The reports would be available immediately and printing costs would be eliminated. This takes advantage of Infoprint Server's intelligent routing function.

One advantage of the integration of the PDF subsystem into the iSeries output architecture is that standard functions that normally relate to print easily translate into desired PDF functions. For example, in the above scenario, assume that the data of interest is on pages 21-50 of the report. If you wanted to print this, you would simply override the printer file attributes to request this page range. The same function works with PDF because it is implemented as a virtual printer. Specify the page range, then pass the spooled file to the PDF subsystem. The resulting PDF contains only pages 21-50 of the report.

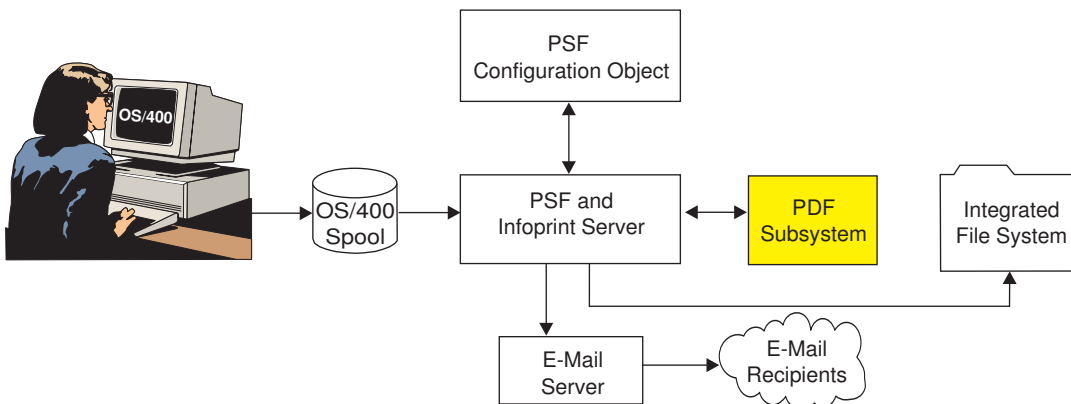


Figure 159. Publishing iSeries Output for Electronic Access

Electronic Report Distribution Using Segmentation

Here's a similar example to the one above, but with a different approach. In this case, sales reports are being produced into iSeries output queues and printed. However, as the application run represents multiple sales regions, the hardcopy output must be separated into regional reports and each regional report is physically sent to that region.

We want to transform this process in two important ways. First, we need to break up the overall report into regional "sub-reports" in order to deliver the segments to the sales regions. For example, we might have a 1000-page sales report comprised of ten different sales regions. Thus, the goal is to create ten regional reports. Second, we want to deliver the reports electronically, not in hardcopy. We also want to "push" the reports out to the regional recipients instead of having the users come get them as in the previous example.

We'll use the segmentation support in OS/400 and Infoprint Server's Create AFP Data command to insert electronic triggers within the 1000-page spooled file, delineating the ten regional report segments. Next, as we pass the spooled file for PDF processing, each regional report is transformed to PDF. At this point, we use the electronic triggers to supply the e-mail IDs for that report's distribution. Either the trigger is an actual e-mail address, or the electronic triggers (such as a region ID) are matched to actual e-mail addresses in the PDF mapping program. The integrated e-mail function sends an e-mail to those addresses with the regional PDF attached. Using the intelligent routing function, each report can also be stored as PDF in a specified location in the integrated file system for archival purposes. This "outbound" electronic process, coupled with segmentation, ensures timely delivery of the right information to the right people, with substantially reduced operational costs.

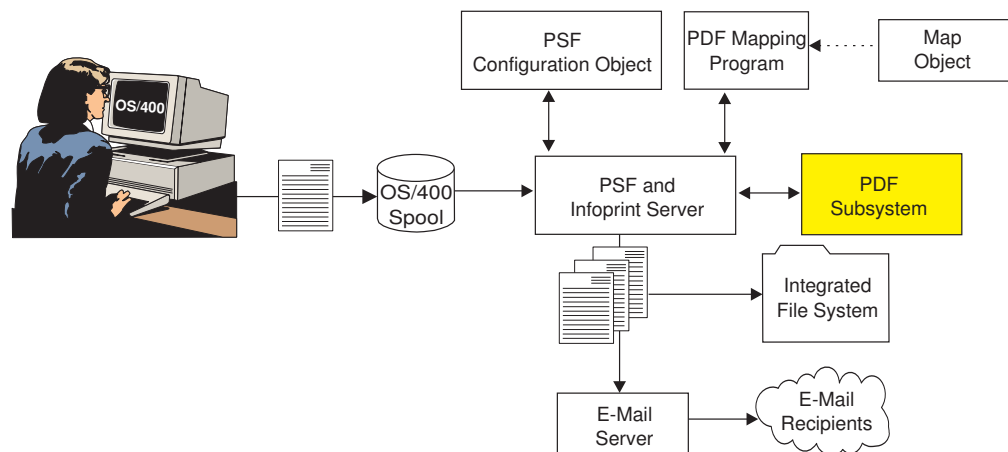


Figure 160. Electronic Report Distribution with Segmentation

E-Business Customer Documents

In a growing number of e-business processes, electronic delivery is a hard requirement, not simply a means to reduce costs and improve timeliness. Let's take an order entry application. Order entry was a batch function and has now been re-engineered as an Internet application. This lets customers enter orders directly. However, the back end of the order entry process - the order confirmation document - is still printed and sent to each customer in hardcopy. This significantly reduces the benefits of using the Internet for this function. The logical requirement is to produce the order confirmation immediately and provide it to the customer electronically.

If we assume that the order confirmation program remains essentially the same, we can create the same spooled file. With the Copy Spooled File API, we can easily create a duplicate of the order confirmation spooled file, add PDF and e-mail parameters and put it on a PDF queue for processing. The order confirmation is converted to PDF and distributed to the customer.

Alternatively, with some Internet programming, the order confirmation can be produced and then passed back to the customer's browser, which could automatically invoke Adobe Acrobat to display the PDF.

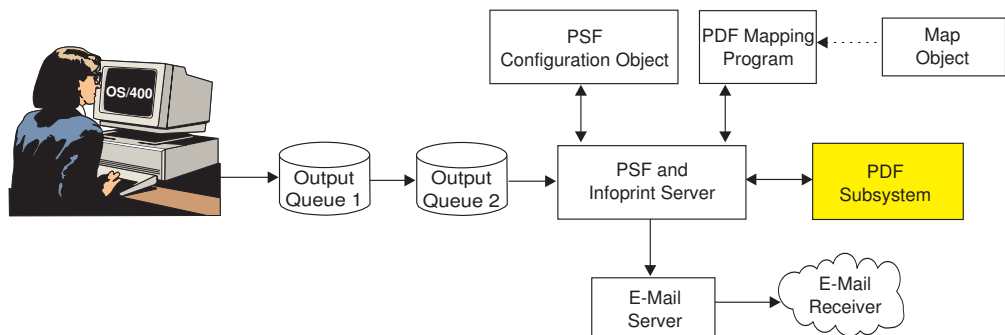


Figure 161. E-Business Customer Documents

Segmenting Customer Documents

In a variation of the sales report example, we have an application that produces customer invoices. Instead of printing the invoices and mailing them in hardcopy format, we want to provide it to the customer in whatever format they choose; hardcopy, e-mail, Web access, and links to fax as options. Additionally, a given customer could select multiple options. For example, Customer A might choose to have a copy of the order confirmation faxed and another copy placed where they can access it electronically with a browser. Customer B might choose to have their order confirmation e-mailed to three different e-mail IDs. Customer C might still want the order confirmation mailed in hardcopy format. The ability to route one spooled file or one segment of a spooled file in a combination of ways is called *intelligent routing*.

Again, we use the electronic segmentation support (Create AFP Data command) to insert triggers between customer invoices. When the entire spooled file of invoices is passed to the PDF queue, each set of pages between the triggers is converted to a PDF or AFP file³. At this point, the PDF subsystem can look to a PDF mapping program to provide additional information about document distribution. Certain parameters, for example, a customer number and spooled file information, can be passed to the PDF mapping program. Based on this data, the e-mail IDs (triggers) can be matched to e-mail addresses and passed back for processing. In addition, other information such as e-mail content (including a password for the PDF attachments) or attachments, spooling information, and the location in the integrated file system can be defined on a per customer basis. With this information, the output is processed. This “outbound” electronic process, coupled with electronic segmentation, ensures timely delivery of customer invoices.

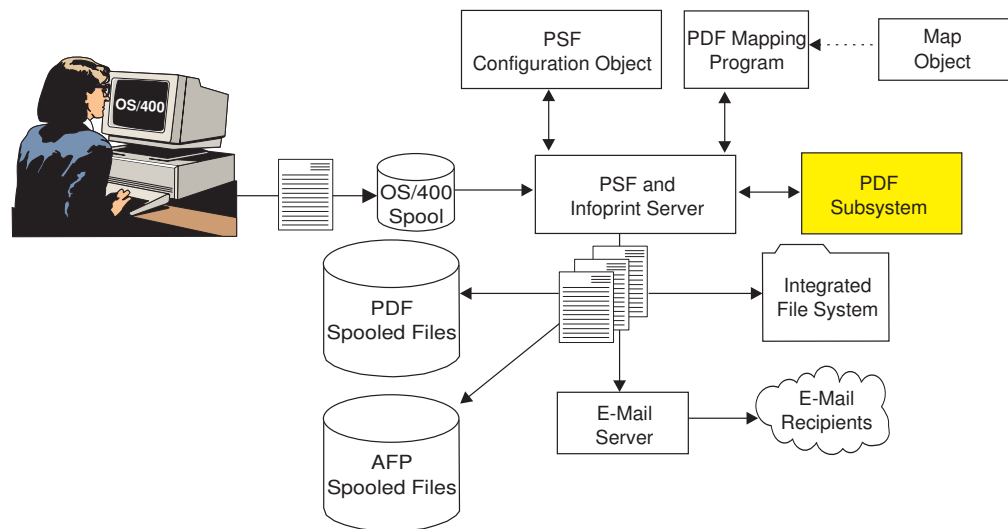


Figure 162. Segmenting Customer Documents

3. When you respool the action is respool as AFP, then that order confirmation can be printed, faxed (with link to a fax program), added to an archive, or similar “downstream” function. However, respooling as AFP requires PSF for OS/400.

Consolidating Enterprise Printing

As your company has re-engineered its network topology to take advantage of intranets and the Internet, printing and printers quickly followed suit. Initially, this decentralization of printing was viewed as a good thing. However, as applications and print volumes increased, this decentralized structure could become a liability. The network-decentralized applications produce a range of ASCII print data streams (PCL, PostScript, and PDF). These data streams were designed for personal and workgroup printing, not business printing.

In addition, companies realized that having a large number of decentralized printers could add significant operational costs, system complexities, and end user frustrations to enterprise printing.

The iSeries has very good support for business printing. We can use the ASCII print data stream transforms of Infoprint Server to consolidate client and network printing on iSeries, applying its superior print management capabilities. The Infoprint Server print data stream transforms will convert PCL, PostScript, and PDF print jobs into AFP image for printing on iSeries-connected IPDS printers. This enables the application of the efficiencies and reliability of iSeries-integrated printing to your decentralized printing - saving costs and improving service levels.

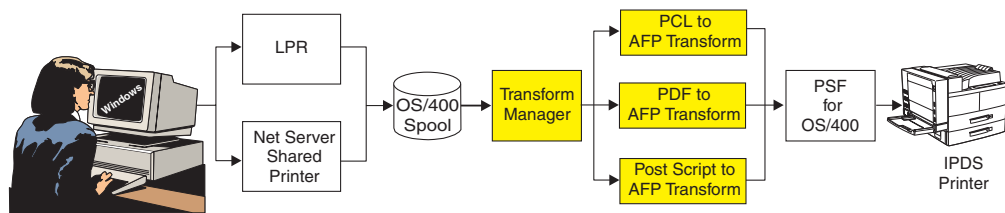


Figure 163. Consolidating Enterprise Printing

Delivering Documents and Reports over Intranet

In the first several application scenarios, we used PDF as the electronic document delivery format. PDF is the standard for Internet document delivery. But in an intranet environment, AFP is an option that offers a number of advantages.

AFP is the standard data stream within iSeries for graphical pages. As we have seen, AFP architecture structures each page by a combination of application data and references to external page resources such as overlays, images, and fonts. When AFP is sent to another system, client PC, or a browser, those external resources need to be present in order to fully resolve each page. Infoprint Server provides the function to convert a standard AFP spooled file into a fully portable file, Internet-enabled AFP. Infoprint Server's Create AFP Data (CRTAFPDTA) command can produce this fully encapsulated AFP. The command will query the AFP data stream, identify the external references, and pull those externally-referenced resources into the fully encapsulated file. This file can be accessed "inbound" off the integrated file system or sent over the company intranet. There are AFP plug-ins for Internet Explorer and Netscape to view Web AFP files in the same manner that Adobe Acrobat provides for viewing PDF files.

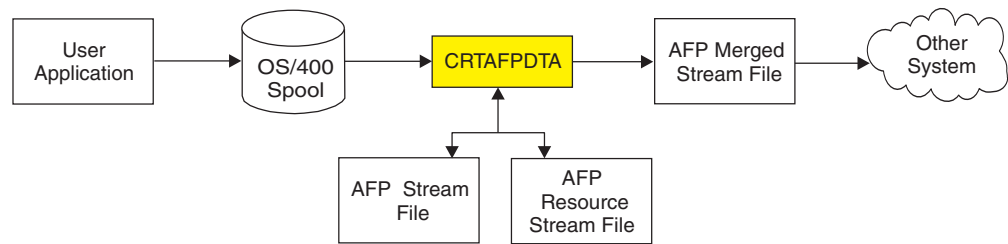


Figure 164. Delivering Documents and Reports over Intranet

Creating Electronic Documents Interactively

Frequently, there are interactive or “ad hoc” requests for documents or reports. These types of requests are the ones that need to be delivered quickly to be effective so they are prime candidates for electronic delivery.

Both iSeries Navigator and iSeries Access for Web provide for interactive access to PDF and e-mail functions, enabling electronic presentation or delivery of the information. Once a required document or report is in an iSeries output queue, either the iSeries Navigator or Web Access graphical interfaces can be used to browse and select the target spooled file. With iSeries Navigator and Infoprint Server, you can define similar PDF and e-mail functions as you can with the standard batch process. Target spooled files can be converted to PDF and written to the integrated file system, to an output queue, or sent by e-mail. With iSeries Access for Web, a selected spooled file is sent back to the browser in PDF format and automatically opened with Adobe Acrobat.

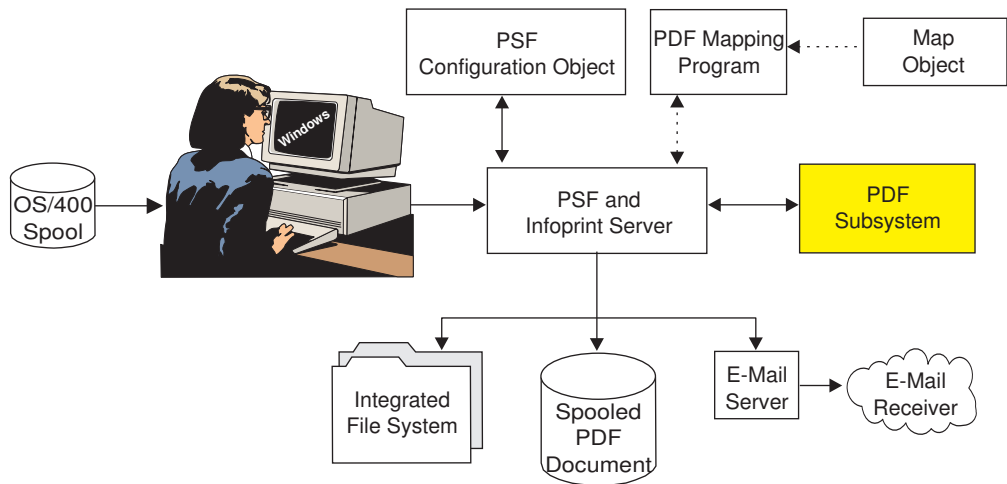


Figure 165. Creating Electronic Documents Interactively

Integrating Web Images

As your enterprise deploys Internet applications, those applications use images in standard formats - typically GIF, TIFF, and JPEG. You might need to use those images in iSeries-based line of business functions that produce output. For example, you might have an engineering workstation that produces manufacturing drawings. The work order produced by the iSeries should include that drawing.

Infoprint Server provides a set of image transforms - GIF, TIFF, and JPEG to iSeries IOCA format. The transform can create the IOCA image as an iSeries page segment object so you can reference it on a document page.

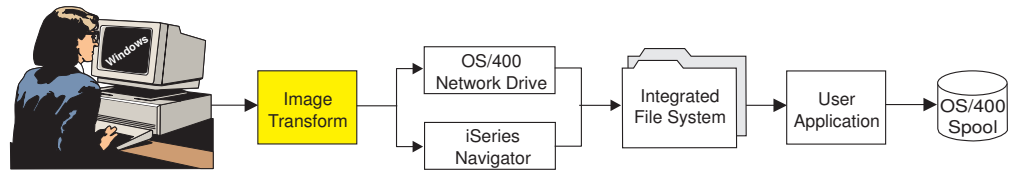


Figure 166. Integrating Web Images

Transforming Your OS/400 Output to PDF

With Infoprint Server, you get an integrated PDF subsystem that can be accessed through OS/400 CL commands or iSeries Navigator. The PDF subsystem lets you transform a spooled file or segment of a spooled file to PDF then send it as e-mail, store it in the Integrated File System, put it on an output queue, or any combination of those. In addition, you can use it with PSF and respool the input spooled file or a segment of the input as AFP. If you choose to work with segments instead of the whole input spooled file, the output is multiple PDF or AFP files, each of which you can distribute in whatever way is appropriate. This process of transforming an input spooled file to one or more output files and routing them the way you want is called *intelligent routing*. The input spooled file can be any type of data that PSF for OS/400 can print.

Notes:

1. The PDF subsystem interacts with PSF to process data. However, you do not need a license for PSF to use the PDF subsystem except to respool data as AFP.
2. You cannot use the intelligent routing function with iSeries Navigator. With iSeries Navigator, you can transform a spooled file to PDF and route it one way.

Before you can invoke the PDF subsystem, you need these:

- A PSF configuration object that specifies that data is to be transformed to PDF. The PSF configuration object also specifies what is done with the PDF output. If your spooled file contains groups, use the PSF configuration object to specify whether the transform generates a separate PDF file for each group or one PDF file containing all of the data and index tags at group boundaries.
- A properly configured device that specifies the PSF configuration object on the User defined object parameter.
- (optional) PDF mapping program to further customize the output. For example, you can use it to specify the e-mail's subject text, the output file's name, and the output file's location in the integrated file system.
- (optional) A map object to specify values to an IBM-supplied PDF mapping program. The map object acts like a database of values that the IBM-supplied PDF mapping program reads. You can create a map object by using CL commands or the PDF map APIs. Any time a PDF mapping program is used, you could use a map object and the IBM-supplied PDF mapping program instead. You would use either a user-created PDF mapping program or a map object, not both.

The PDF subsystem looks to the OS/400 like a printer, but instead of actually printing data, it transforms the spooled file then works with PSF to distribute the input. This is how the PDF subsystem works:

1. A spooled file is sent to a device that specifies a PSF configuration object. The PSF configuration object specifies that data is to be transformed to PDF.
2. The PDF subsystem transforms the spooled file and works with PSF to put the PDF file in the specified location in the Integrated File System, spools it as a PDF or AFP file for printing, e-mails it, or any combination of those. It can optionally use information from a PDF mapping program to customize the output.

Figure 167 on page 255 illustrates this process.

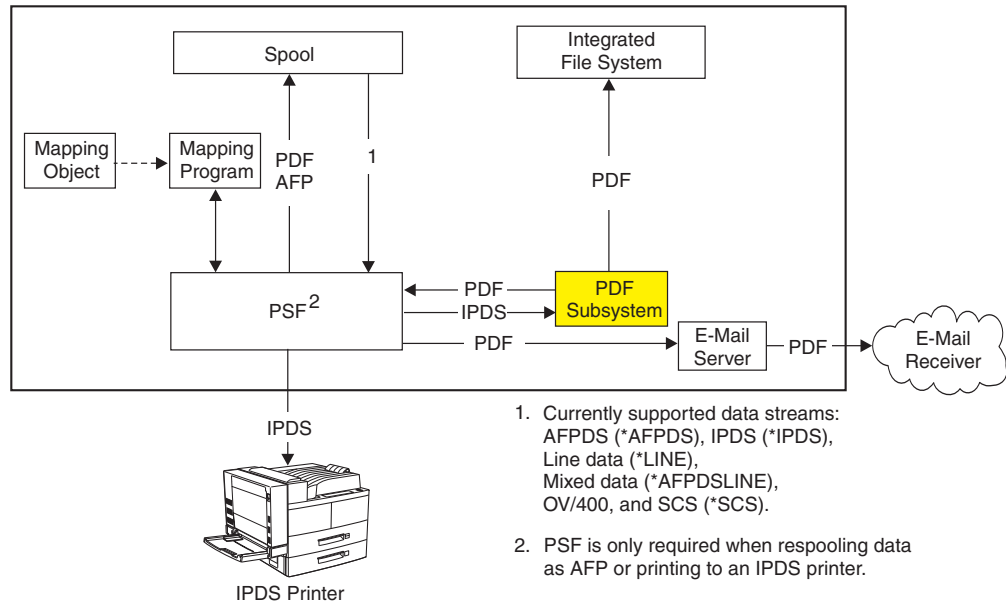


Figure 167. Transforming a Spooled File to PDF

Accessing the PDF Subsystem

While the normal access to PDF functions is through PDF queues, there are several additional ways to access the PDF subsystem:

- iSeries Navigator
- Copy Spooled File OS/400 API
- CL commands

Before you can invoke the PDF subsystem, you need to ensure that you have met all of the requirements listed on page 254.

To access the PDF subsystem from iSeries Navigator:

First, ensure that AFP Manager is installed. Then you simply right-click the printer output (spooled file), select **Convert to PDF...**, and fill in the appropriate values on the Convert to PDF window. See “Converting a Spooled File to PDF for E-mailing, Spooling, or Storing” on page 269 for more details. You cannot use the intelligent routing function with iSeries Navigator. With iSeries Navigator, you can transform a spooled file to PDF and route it one way.

Accessing the PDF subsystem from the Copy Spooled File API:

The Copy Spooled File API can help you automate the process of accessing the PDF subsystem. Assume, for example, that you have a Web application that processes orders and generates a spooled file from the order. You can use the Copy Spooled File API to make a copy of the spooled file and put it on an output queue that is set up to send data to the PDF subsystem. If the writer associated with that queue is started and the spooled file is in ready state, the spooled file is processed without user intervention.

To access the PDF subsystem from the OS/400 command line:

Use one of the OS/400 output queue commands to move the target spooled files to the queue of an appropriately configured PDF device.

For more information about using the Infoprint Server PDF subsystem, including detailed setup information, instructions about using iSeries Navigator, and instructions about using the command line, refer to *Infoprint Server for iSeries: User's Guide* and *IBM@server iSeries Printing VI: Delivering the Output of e-business*, SG24-6250.

Intelligent Routing

Intelligent routing is the ability to send a spooled file to the PDF subsystem and specify that the output be distributed in any combination of these ways:

- spooled to an output queue as AFP or PDF data
- stored as a PDF stream file in the integrated file system
- sent as a PDF file attached to an e-mail

When you use intelligent routing, you might want to use a *PDF mapping program*. A PDF mapping program is used to specify what you want done with the output files and lets you specify some options (such as PDF encryption for e-mail) that you cannot specify with a PSF configuration object. You can either write a PDF mapping program or you can use the IBM-supplied PDF mapping program. If you use the IBM-supplied PDF mapping program, you must use a *map object* with it. A map object passes values to the IBM-supplied PDF mapping program. You use OS/400 panels or the PDF map APIs to create map object.

Intelligent routing can be applied to files generated from group tags in the input spooled file as well. If your input spooled file has group tags and you specify PDFMULT(*YES *SPLIT) on the PSF configuration object, by default each generated output file is distributed the same way. For example, if you specified PDFGEN(*MAIL), by default each PDF file is e-mailed. If you do not want them all distributed the same way, use a PDF mapping program to specify how to distribute each generated file.

For more information about using the PDF subsystem, including information about the PDF mapping program and map object, setting up your device and PSF configuration object, and descriptions of the PSF configuration object parameters related to the PDF subsystem, refer to *Infoprint Server for iSeries: User's Guide* or the Redbook *IBM @server iSeries Printing VI: Delivering the Output of e-business*.

E-Mail Support

To send a spooled file as e-mail, PSF receives a spooled file that has been sent to an appropriate device (described in the above section). PSF then sends the spooled file to the PDF subsystem. The PDF subsystem transforms the spooled file and sends the PDF file back to PSF. PSF e-mails the PDF file as an attachment to the specified receiver.

The simplest way to send e-mail is to accept default text for the e-mail subject and body and specify a valid e-mail addresses on the User defined data (USRDFNDDTA) field in the printer file or the spooled file, or in the data. Alternatively, you can use DDS keywords, the Infoprint Server CRTAFPDTA command, or AFP Toolbox to specify the e-mail address in the data.

PSF can optionally use information from a user-defined or IBM-supplied PDF mapping program (hereafter, both are referred to as the PDF mapping program) to customize the e-mail. A PDF mapping program is a versatile user exit program. You can use it to interpret your mail tags, specify the e-mail subject, and add text to the beginning of each e-mail. You can also use it to specify the PDF attachment's name, encryption options, and more. If you use an SMTP server to send the e-mail, you can also use the PDF mapping program to specify carbon copy (cc), blind carbon copy (bcc), and reply to addresses, files to use as the e-mail body, and attachments, and more. You do not have to use a PDF mapping program unless you want to specify options that are not available on the PSF configuration object.

PSF uses the OS/400 SNDDST command or a specified SMTP server to send the e-mail. The mail server to use is specified on the PSF configuration object.

Indexing Output with the Create AFP Data Command

Indexing your output lets you quickly navigate through large files to find the information you need. For example, a telephone company might have one file that contains all of their bills for the month. If it is properly indexed, a customer service representative can open the file for the appropriate month in the AFP Viewer and use the index to search for a customer's bill based on the name, phone number, or other identifier.

The Infoprint Server Create AFP Data (CRTAFPDTA) command transforms a line (*LINE) or mixed (*AFPDSL) spooled file into AFP (*AFPDS). It can also take an AFP spooled file as input and output an AFP stream file. It indexes a line, mixed, or AFP document for viewing, archiving, or document retrieval, and optionally retrieves and packages the AFP resources that are needed for printing or viewing.

CRTAFPDTA takes input data from the spool and creates up to four stream files in the Integrated File System:

AFP stream file

The AFP document. This file is always created. Specify a value for **To stream file** (TOSTMF) to generate this file.

Resource stream file

A stream file that contains all the resources needed for printing or viewing. This file is optional, but is useful for archiving and viewing the document. Specify a value for **To resource stream file** (TORSCSTMF) to generate this file.

Index stream file

A stream file that contains indexing information. This file is optional, but is useful for viewing the document. Specify a value for **To index stream file** (TOIDXSTMF) to generate this file.

Merged stream file

A stream file that merges the AFP file with any of the other above files that are created. This file is optional, but is useful for archiving and viewing the document. Specify a value for **To merged stream file** (TOMRGSTMF) to generate this file.

After these files are created, you can route them to an archive system, re-queue them for printing, or view them using viewers such as the AFP Viewer or the AFP Viewer plug-in. Figure 168 on page 258 illustrates this process:

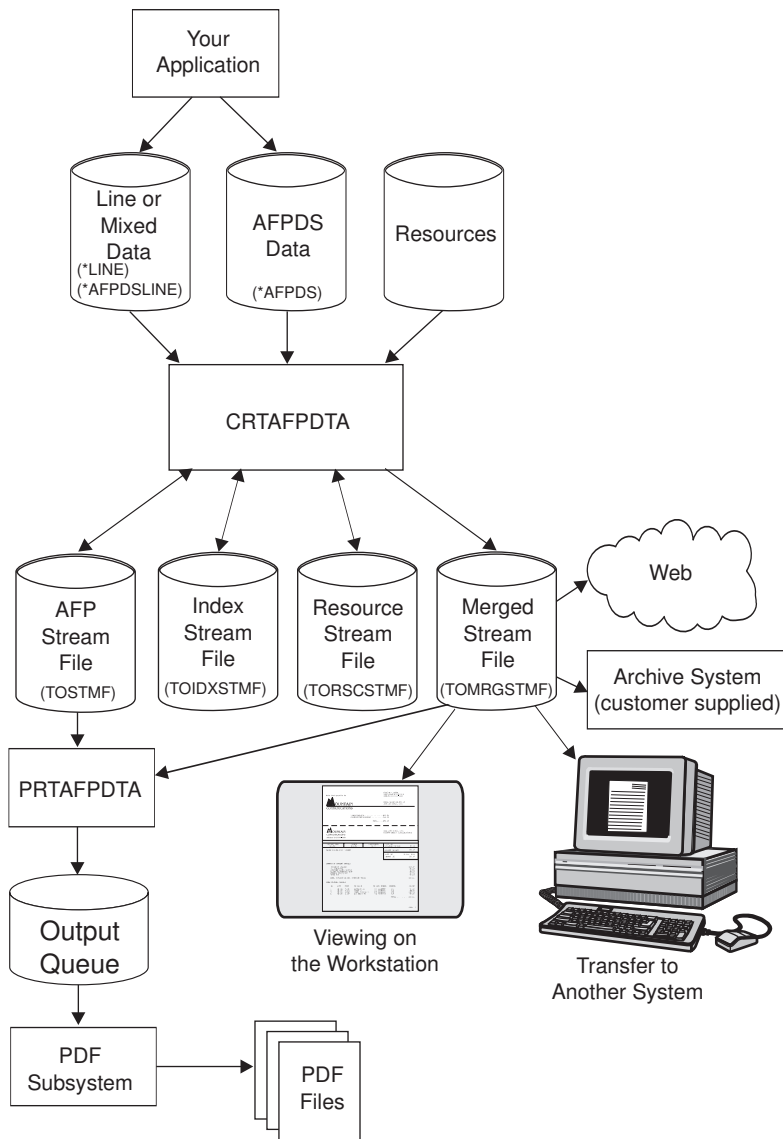


Figure 168. How the Infoprint Server Create AFP Data Command Works

The CRTAFPDTA command is going to read in line, mixed, or AFP print data, then retrieve the external page resources (such as fonts, images, and overlays) that are referenced in that print data. The output can include an index stream file or a resources stream file. Or, in the most likely scenario, a merged stream file is created with the index and resource data embedded.

CRTAFPDTA implements functionality available to other platforms with AFP Conversion and Indexing Facility (ACIF). ACIF is part of PSF on MVS, VM, and VSE and is also part of Infoprint Manager on AIX and Windows NT. You can find more information about using the functions available through the CRTAFPDTA command in *AFP Conversion and Indexing Facility: User's Guide*, S544-5285.

Transforming ASCII Image Data to AFP

Infoprint Server comes with three Windows-based image transforms that can convert GIF, TIFF, and JPEG images to AFP documents, page segments, and overlays. The transform processes bilevel (black and white), grayscale and color images. The output is bilevel, 4-bit or 8-bit grayscale, or 24-bit YCbCr color.

These are command-driven transforms so you can control nearly every aspect of the transformation, including output type, output image dimensions, and rotation. To transform an image, enter *type2afp* along with any parameters you want, on a DOS prompt, where *type* is gif, tiff, or jpeg.

Example:

In this example, we transform a TIFF file called MyOverlay.tiff into an AFP overlay called MyOverlay.afp. It is stored in the specified directory so it can easily be imported to the system MyIseries. Enter this command at a DOS prompt:

```
tiff2afp MyOverlay.tiff -pagetype overlay -o MyOverlay.afp \\MyIseries\QDLS\
```

After the AFP resource (page segment or overlay) is created, you can use iSeries Navigator or CL commands to create it on your iSeries as a resource.

Printing PCL, PDF, and PostScript Data on IPDS Printers

Infoprint Server lets you send PCL, PDF, and PostScript data to print on IPDS printers. This gives more applications access to the speed, power, throughput, and IPDS error recovery of your iSeries printers.

Notes:

1. Printing PDF and PostScript data to an IPDS printer requires priced feature 5101.
2. Printing on IPDS printers requires PSF for OS/400.

When a spooled file is released and has one of the data formats for which a transform is provided, an OS/400 AFP printer writer can automatically call image print transform. Next, image print transform calls the appropriate transform program. The PDF, PCL, or PostScript data is then converted to an AFP image to preserve fidelity and printed on the IPDS printer.

If you want a printer writer to automatically use a transform, you must specify appropriate values in the configuration file `qxtrtfmmgr.cfg` and you must specify an appropriate image configuration object in the printer device description. Refer to *Infoprint Server for iSeries: User's Guide* for information about setting up your system to use these transforms. In the configuration file, you can specify how many transforms of each type to initialize upon startup and how many can run at once.

These transforms are managed by Transform Manager. Enter `STRTFMMGR` on the command prompt to start Transform Manager. This starts the minimum number of each type of transform specified in the configuration file. Enter `ENDTFMMGR` on the command prompt to end Transform Manager. This command immediately ends all transform jobs started by Transform Manager.

Chapter 23. Working with Your iSeries from a Windows Client

IBM provides two tools that you can use to control your iSeries from a workstation: iSeries Access and iSeries Access for Web. This chapter describes both of those tools and contains instructions to use iSeries Navigator to complete these tasks:

- “Converting a Spooled File to PDF for E-mailing, Spooling, or Storing” on page 269
- “Creating a PSF Configuration Object” on page 271
- “Changing a PSF Configuration Object” on page 272
- “Importing Resources to the OS/400” on page 273
- “Creating a Printer Share” on page 274
- “Specifying which Libraries to Display” on page 277

Many of these tasks require AFP Manager, a component of iSeries Navigator.

iSeries Access

iSeries Access for Windows delivers TCP/IP connectivity to users running a variety of Microsoft Windows operating systems. iSeries Access offers an all-inclusive client solution for accessing and using resources from your Windows desktop. It includes 5250 emulation, access to DB2 Universal Database (UDB) for iSeries through its Data Transfer, and uses AS/400 NetServer for working with the OS/400 Integrated File System and printers. It also has a variety of middleware for using and developing client applications to access OS/400 resources; and Operations Navigator, the OS/400 GUI, for administering iSeries and AS/400 servers.

The window shown in Figure 169 opens when you double-click the iSeries Access icon. Double-click any of the components, such as iSeries Navigator, to open them.

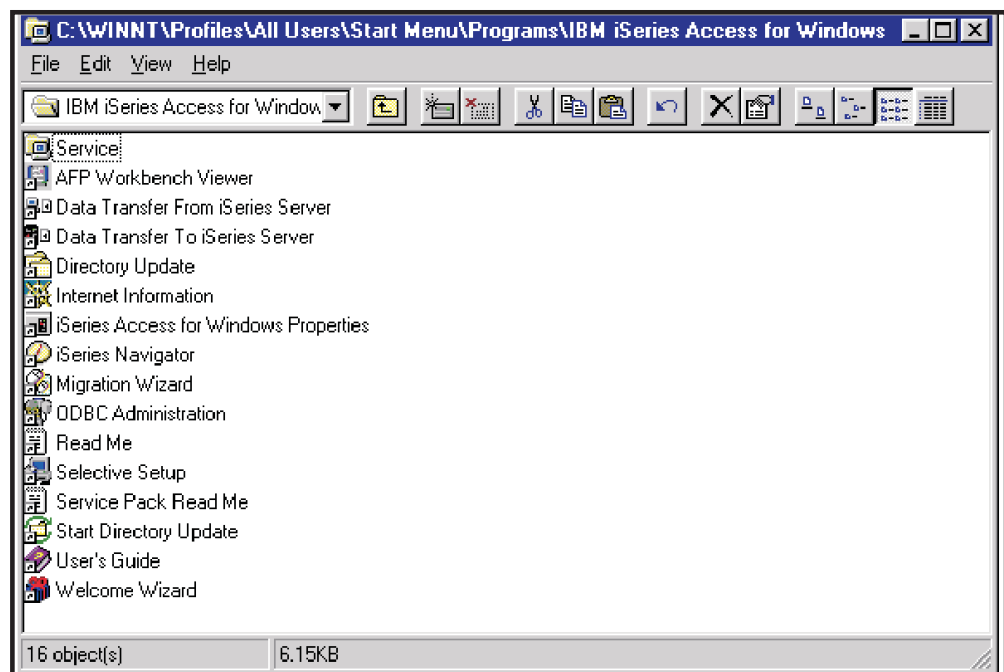


Figure 169. iSeries Access Components

iSeries Navigator

iSeries Navigator, formerly known as Operations Navigator and Management Central, is the strategic user interface for your iSeries servers. It provides integrated systems management with an easy-to-use graphical user interface. It is a component of iSeries Access. When you start iSeries Navigator and expand an iSeries connection, the window shown in Figure 170 opens. It varies according to the components you install.

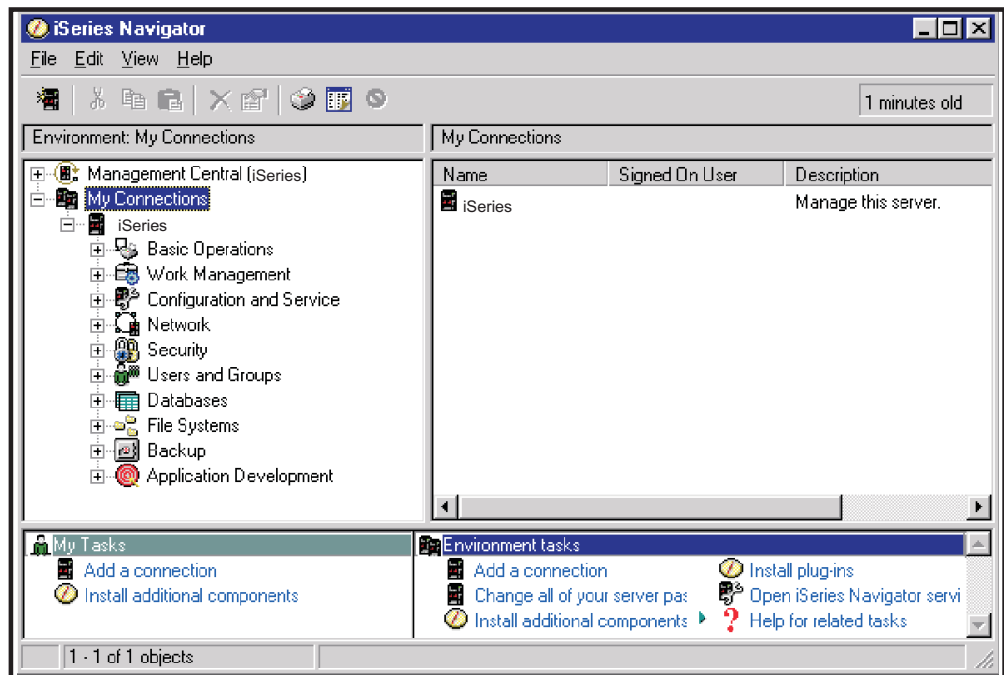


Figure 170. iSeries Navigator Window

AFP Manager

AFP Manager is the component of iSeries Navigator that gives you access to AFP print objects, including resources, PSF configuration objects, and font mapping tables. These objects are shown in Figure 171 on page 263. If you have AFP Manager installed, you can transform a spooled file to PDF and send it as e-mail, store it on an output queue, or save it in the Integrated File System in one step.

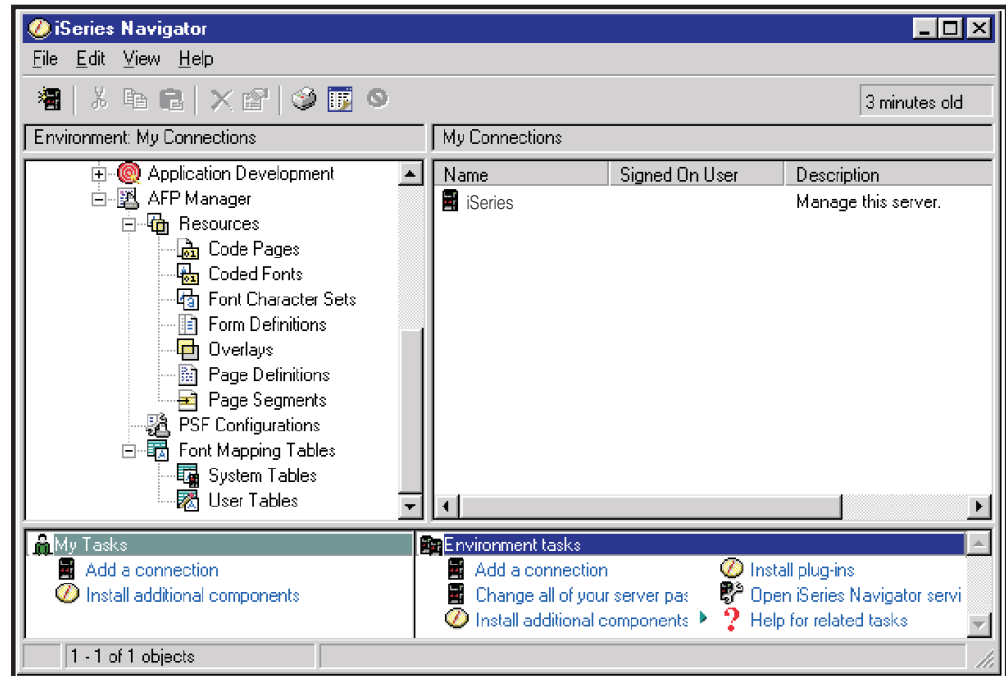


Figure 171. AFP Manager

iSeries Access for Web

IBM iSeries Access for Web (5722-XH1) offers browser-based access to iSeries servers. It enables end users to leverage business information, applications, and resources across an enterprise by extending the iSeries resources to the client desktop through a Web browser. iSeries Access for Web implements many of the functions available through iSeries Access, including the ability to work with Infoprint Server to display and print iSeries spooled files in PDF format. It has these print-related capabilities:

- **Printers** - It lists all of the printers on the system. You can work with your printers from either of two views. The basic view requires little understanding of iSeries print internals, while the advanced view lets users control writers, printers, and their output queues individually.
- **Printer Shares** - It lists all of the printer shares on the system. If a printer is associated with the share, you can control it like any other printer.
- **Internet Printers** - It lists the internet printers on the system. If the internet printer is associated with an actual printer, you can control it like any other printer.
- **Output Queues** - It lists the output queues on the system and lets you hold or release an individual queue.
- **Spooled files** - By default iSeries Access for Web lists all of the spooled files for the user that is signed on. For individual spooled files, these actions are supported, depending on the current status of the spooled file:
 - Display message
 - Reply to message
 - Print next
 - Hold

- Release
- Delete
- PDF
- Preview options: GIF, TIFF, PCL, and AFP Viewer.

iSeries Access for Web is installed on your OS/400 server and only requires a Web browser on the client machine. Once installed, the link to iSeries Access for Web is `http://myiseries:optionalport#/webaccess/iWAHome`, where `myiseries` is the system name. If `http` is configured for something other than the default port, the `optionalport#` must be specified. This is the iSeries Access for Web home page:



Figure 172. iSeries Access for Web Home Page

For more information about iSeries Access for Web, refer to this Web page:
<http://www-1.ibm.com/servers/eserver/iseries/access/web/>

Using iSeries Access for Web to Convert and View a Spooled File in PDF

You can use iSeries Access for Web to convert a spooled file to PDF and view it, place it in a folder on the iSeries, or e-mail it. This requires Infoprint Server. Follow these steps to convert a spooled file to PDF:

1. Start a V5R2 iSeries Access for Web session.
2. Click **Printer Output**. This displays your spooled files, as shown below:

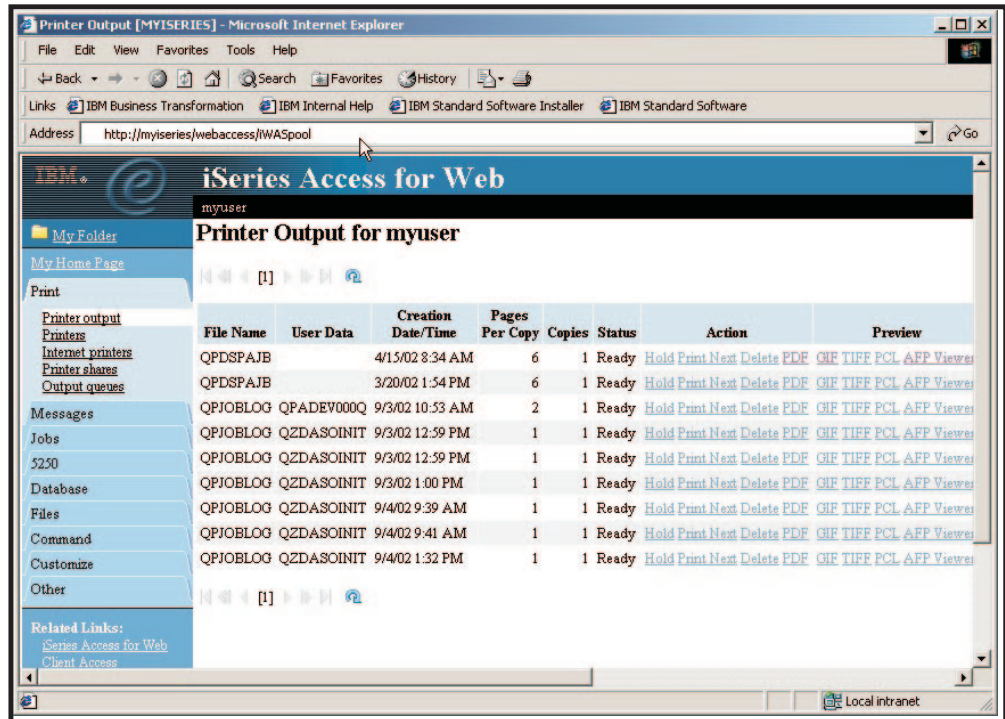


Figure 173. iSeries Access for Web Home Page

- To view a spooled file in PDF format, click the PDF link for the desired spooled file. This brings up a PDF configuration form, as shown below:

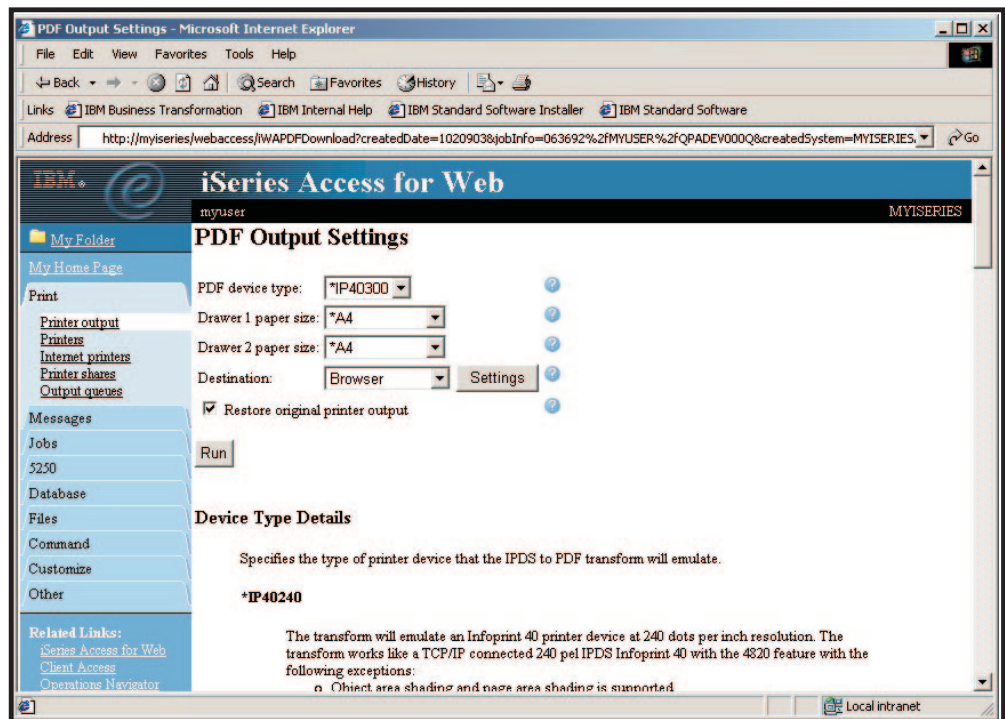


Figure 174. PDF Output Settings Page

- For **Destination**, specify one of these options:

- **Browser** displays the PDF file in your Web browser using the Acrobat Reader plug-in, as shown below:

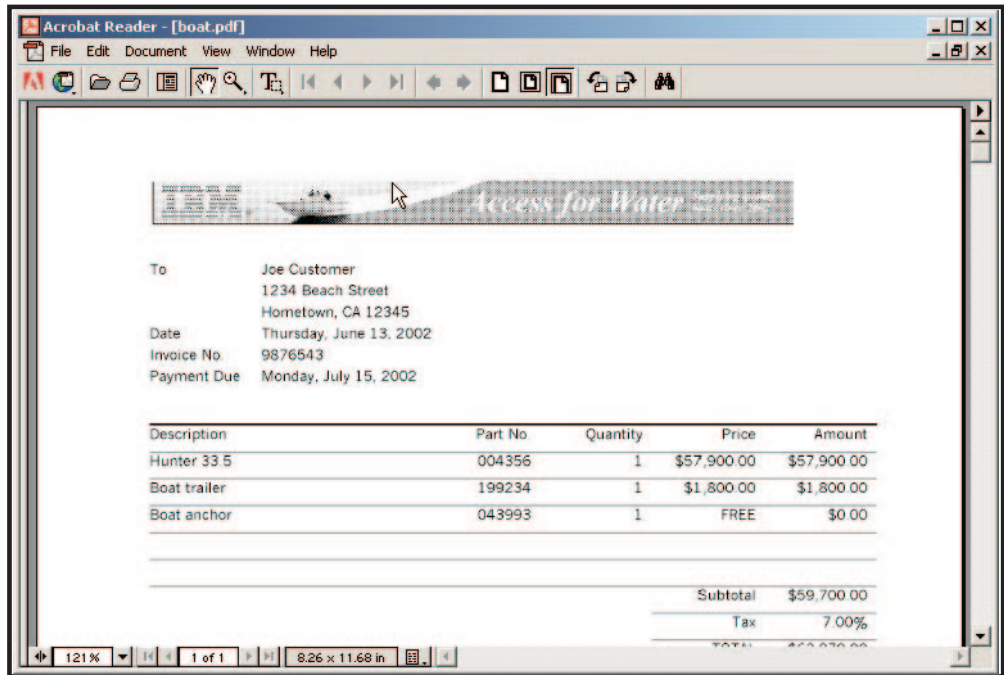


Figure 175. File Displayed in Acrobat Reader

- **Personal Folder** lets you name the PDF file and place it in your iSeries Access for Web folder and place it in other iSeries Access for Web users' folders. It also displays the PDF file with the Acrobat Reader plug-in. A personal folder with one PDF folder item in it is shown below: The My Folder link is in the upper left corner of the home page.

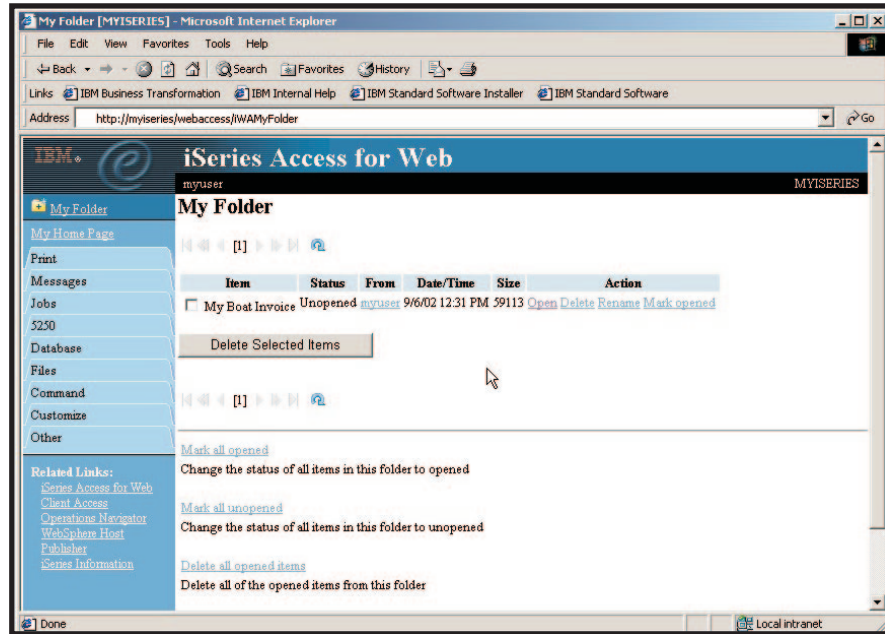


Figure 176. Personal Folder Page

- **Mail as Attachment**^{4 5} lets you send the PDF file as an e-mail attachment. You can specify the subject and message content, as shown below:

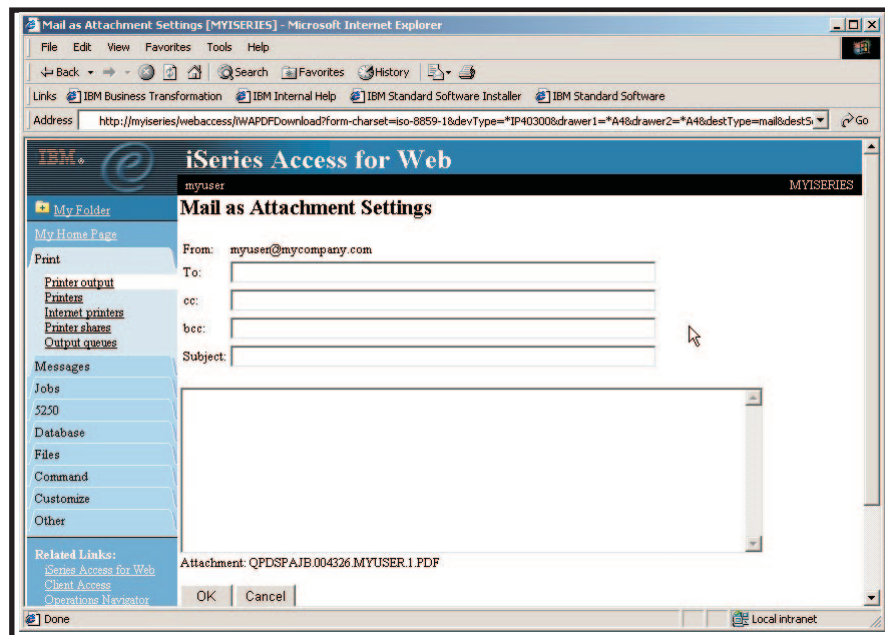


Figure 177. Settings for Mail an Attachment Option

5. Specify whether you want the original spooled file to stay on the output queue after it is transformed to PDF.

4. This option does not use Infoprint Server e-mail support.

5. This option is only available if you have used the iSeries Access for Web SMTP mail configuration support.

Using iSeries Access

These sections give you step-by-step directions for using iSeries Access to complete OS/400 print-related tasks.

Installing AFP Manager

Many of the tasks described in this section require AFP Manager. If you specified Full Install at install time, AFP Manager is automatically installed. Otherwise, follow these steps to install AFP Manager:

1. Double-click the iSeries Access icon.
2. Double-click the iSeries Navigator icon.
3. Open a connection to an iSeries system.
4. Click **Install additional components** in the task window, this option is circled in Figure 178.

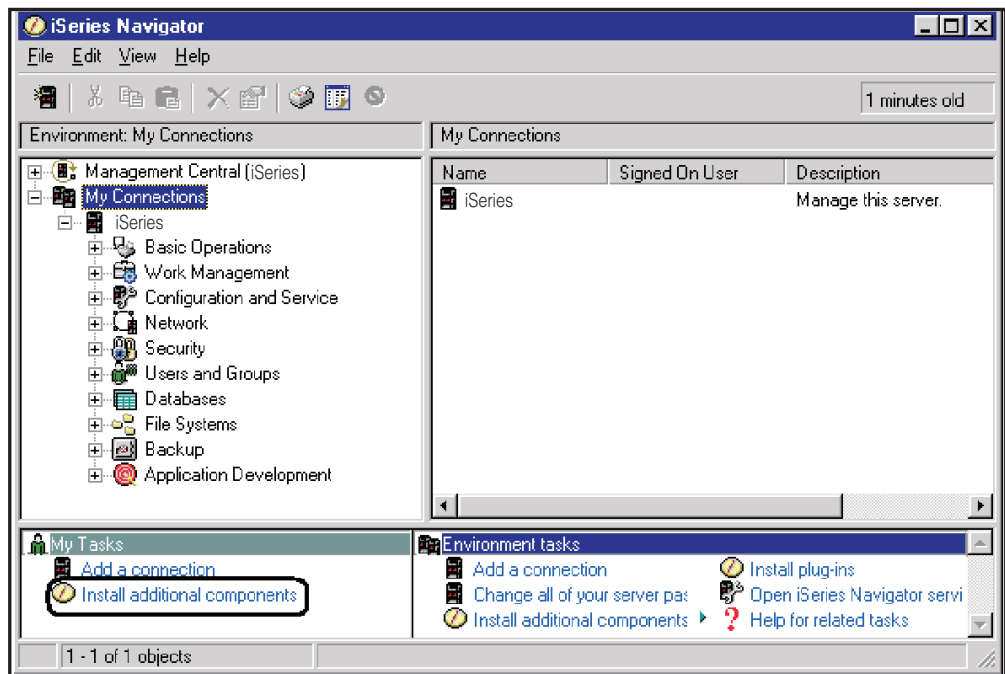


Figure 178. iSeries Navigator Window – Install Additional Components

5. The Selective Setup window opens. Click the arrow to specify the iSeries from which you want to install AFP Manager and click **OK**.

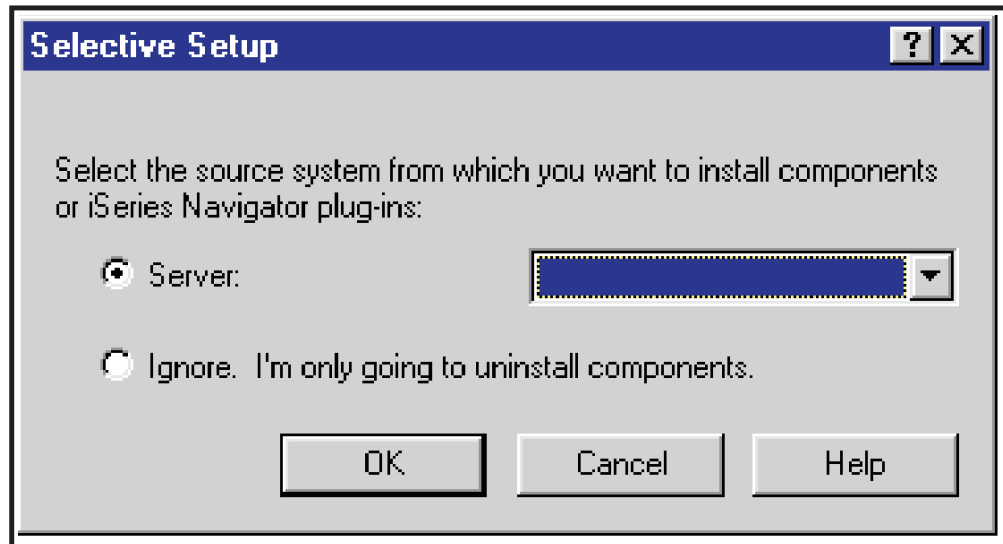


Figure 179. Selective Setup Window

6. The Selective Setup Wizard starts. Follow the directions on the screen. When the Component Selection window opens, all of the components installed on the specified iSeries server are selected. Scroll down and select **AFP Manager**. Click **Next** and follow the remaining directions.

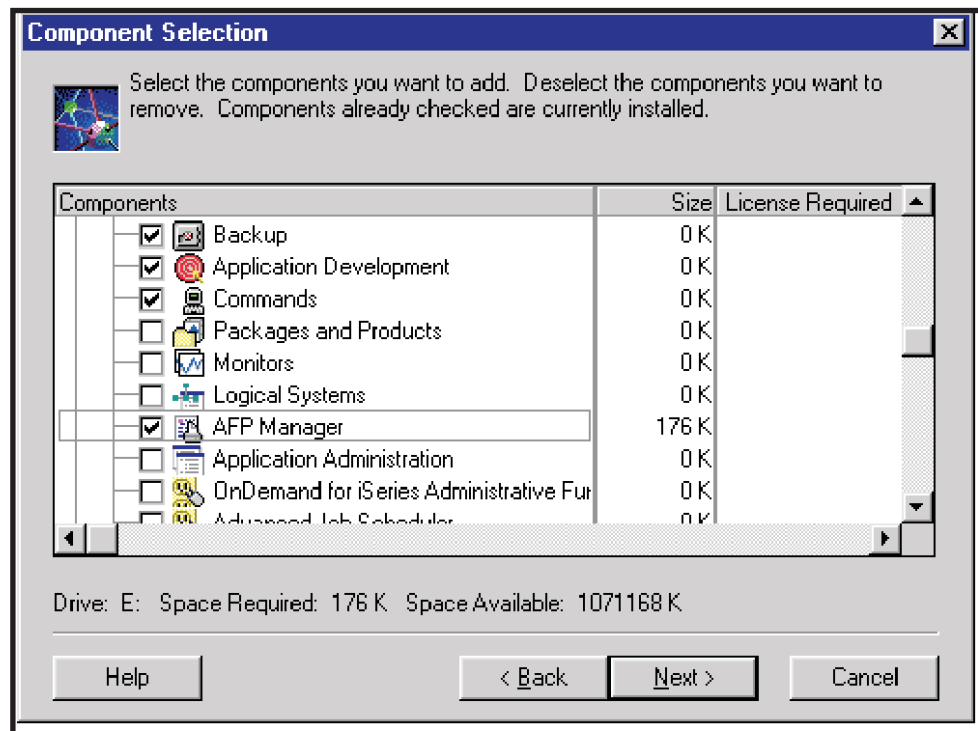


Figure 180. Component Selection Window

Converting a Spooled File to PDF for E-mailing, Spooling, or Storing

You can use AFP Manager to convert a spooled file to PDF then send it as e-mail, store it in the Integrated File System, or put it on an output queue.

Before you can invoke the PDF subsystem, you need these:

- A PSF configuration object that specifies that data is to be transformed to PDF. The PSF configuration object also specifies what is done with the PDF output. If your spooled file contains groups, use the PSF configuration object to specify whether the transform generates a separate PDF file for each group or one PDF file containing all of the data and index tags at group boundaries.
- A properly configured device that specifies the PSF configuration object on the “User defined object” parameter.

You might already have a PSF configuration object and properly configured device. Step 7 describes how you can tell whether you have them.

To convert a spooled file to PDF, follow these steps:

1. Double-click the iSeries Access icon.
2. Double-click the iSeries Navigator icon.
3. Open a connection to an iSeries system.
4. Expand **Basic Operations**.
5. Open **Printer Output**.
6. From the list of printer output (spooled files), right-click the file you want to convert and select **Convert to PDF** as shown below.

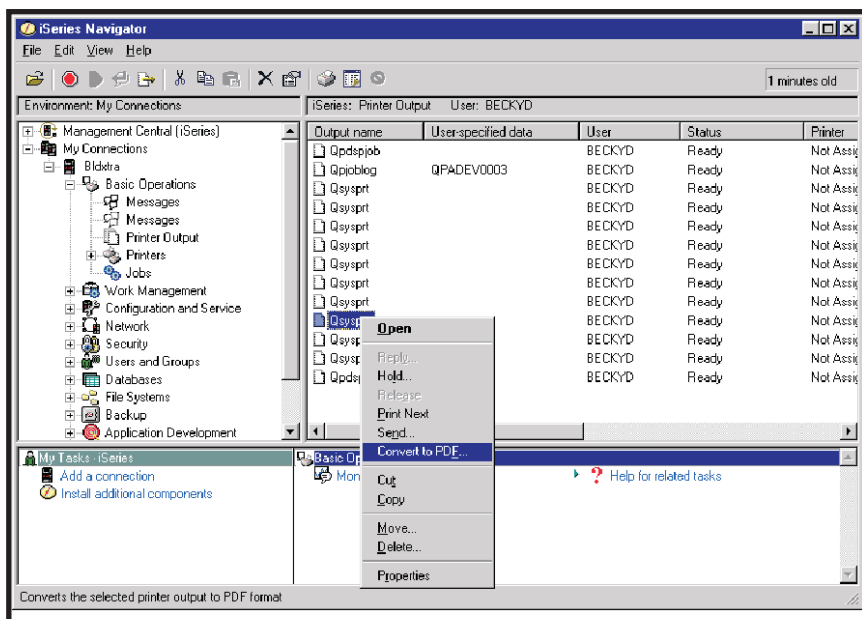


Figure 181. Selecting a Spooled File to Convert to PDF

7. The “Convert printer output to PDF” window opens, as shown in Figure 182 on page 271. Use these fields to specify what is done with the PDF, then specify the e-mail address to which the e-mail is sent, the directory to place the output in, or the output queue to place the output on, as appropriate. In the “Device for conversion” section, choose the printer to use to convert your output. If there are no devices listed, there are no appropriately configured devices. For help, press F1 or click **Help**.

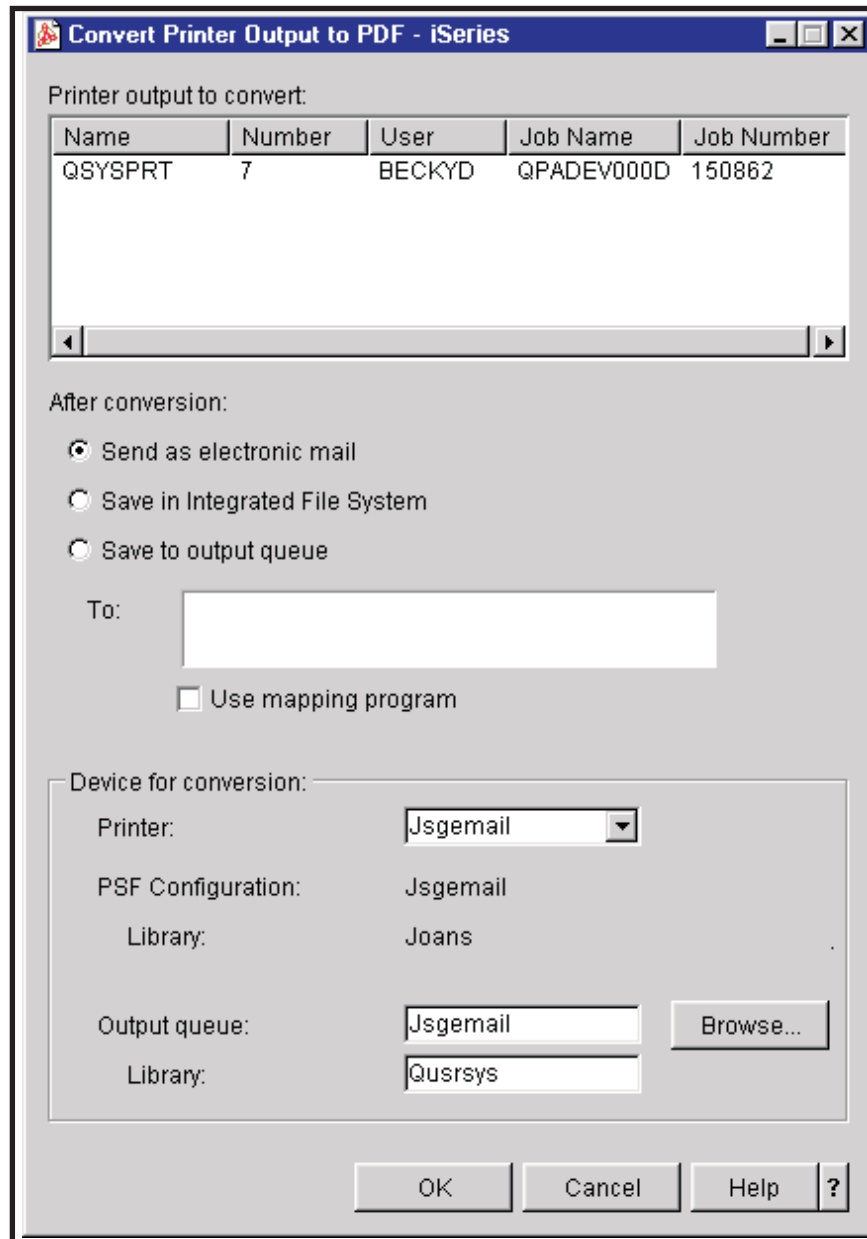


Figure 182. Convert Printer Output to PDF Window

8. Click **OK**.

Creating a PSF Configuration Object

This task requires AFP Manager. To create a PSF configuration object, follow these steps:

1. Double-click the iSeries Access icon.
2. Double-click the iSeries Navigator icon.
3. Open a connection to an iSeries system.
4. Expand **AFP Manager**.
5. Right-click **PSF Configurations** and select **New....**

6. The window shown in Figure 183 opens. Fill in the fields as prompted. For help press F1 or click **Help**.

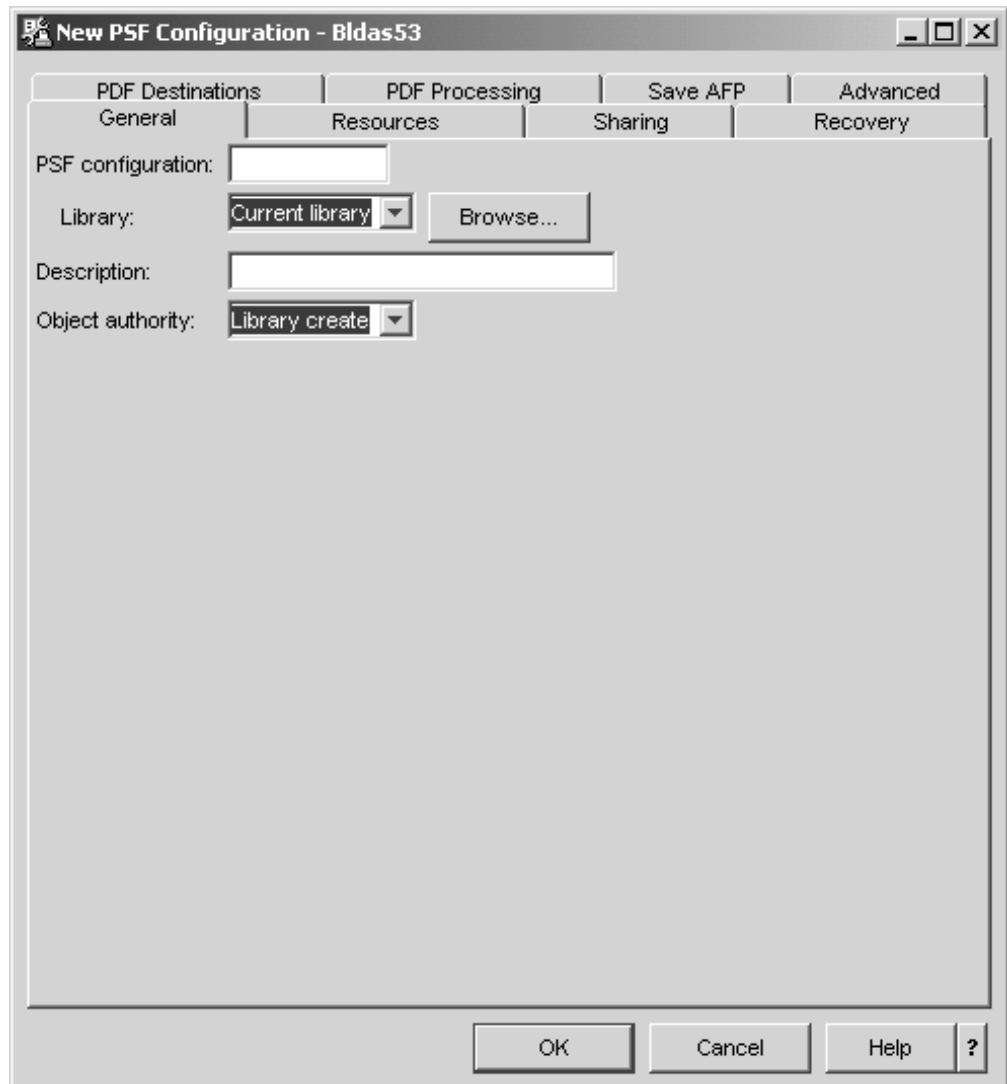


Figure 183. New PSF Configuration Window

7. Click **OK**.

Changing a PSF Configuration Object

This task requires AFP Manager. To change a PSF configuration object, follow these steps:

1. Double-click the iSeries Access icon.
2. Double-click the iSeries Navigator icon.
3. Open a connection to an iSeries system.
4. Expand **AFP Manager**.
5. Expand **PSF Configurations**.
6. Right-click the PSF configuration object you want to change and select **Configure**. The window shown in Figure 184 on page 273 opens, with the name of the PSF configuration object in the title.

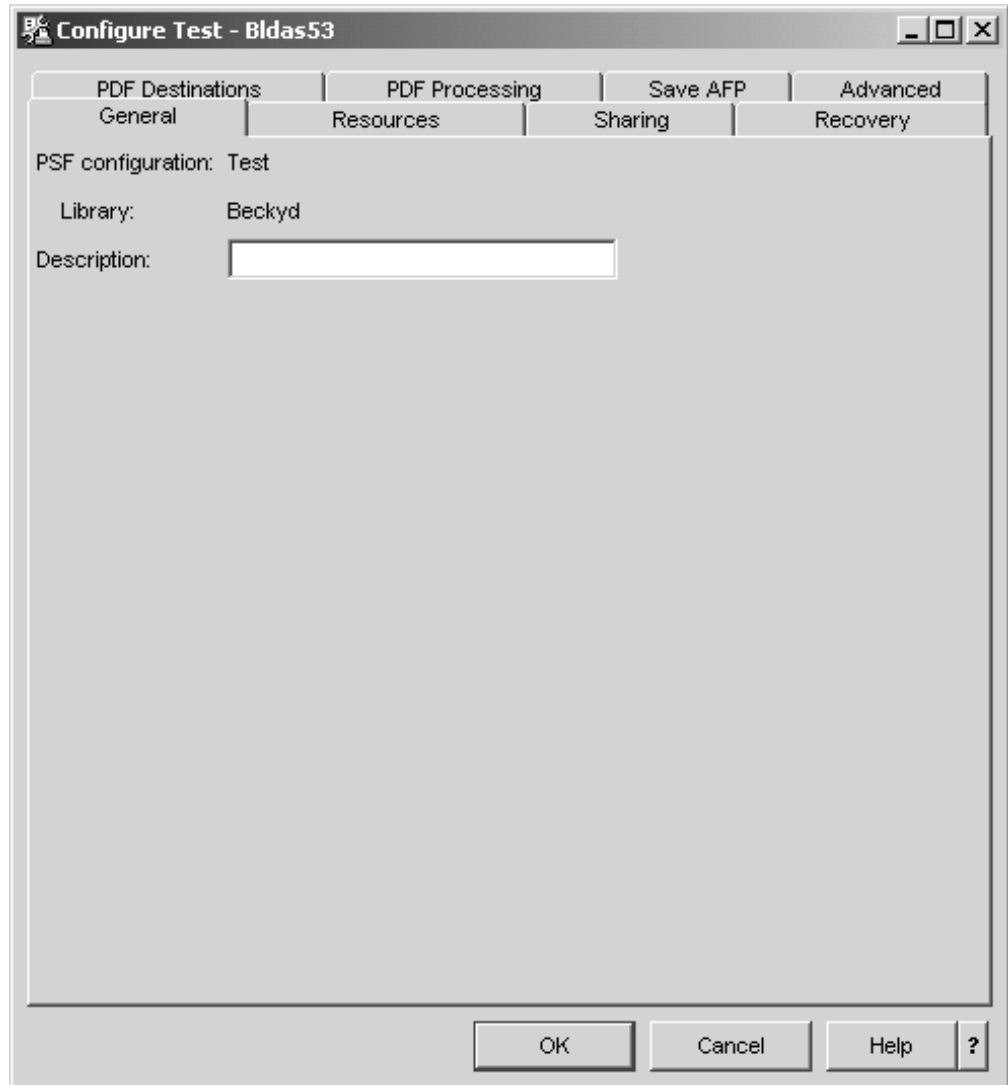


Figure 184. Configure PSF Configuration Window

7. Change any fields you want. For help press F1 or click **Help**.
8. Click **OK**.

Importing Resources to the OS/400

You can use iSeries Access to import an AFP resources from your PC to the OS/400. To do this, you need AFP Manager.

To use AFP Manager to import AFP resources to the OS/400, follow these steps:

1. Double-click the iSeries Access icon.
2. Double-click the iSeries Navigator icon.
3. Open a connection to an iSeries system.
4. Expand **AFP Manager** then **Resources**.
5. Right-click the type of resource you want to store on the OS/400, and select **Import**. A window like the one shown in Figure 185 on page 274 opens, depending on the type of resource you selected.

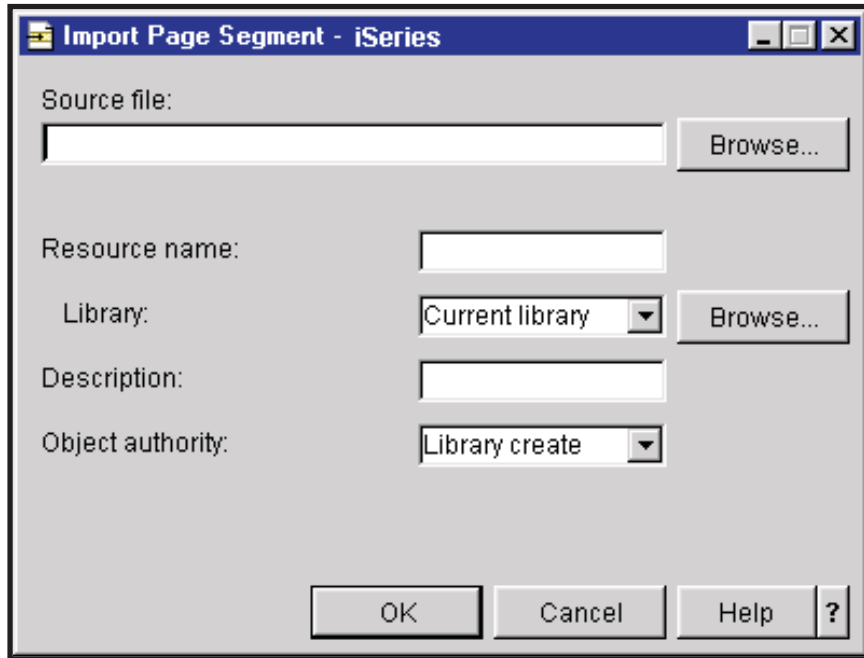


Figure 185. Import Page Segment Window

6. Fill in the fields as prompted. For help press F1 or click **Help**.

Creating a Printer Share

You can use iSeries Access to create a printer share on the OS/400. If you have Infoprint Server installed and you specify a PostScript, PDF, or PCL printer driver, the printer share can be used to print PostScript, PDF, or PCL data that are generated by your PC application on AFP printers. For information about transforming PostScript, PDF, or PCL data to AFP data using Infoprint Server, refer to *Infoprint Server for iSeries: User's Guide*. To create a printer share, follow these steps:

1. Double-click the iSeries Access icon.
2. Double-click the iSeries Navigator icon.
3. Open a connection to an iSeries system.
4. Expand **Basic Operations** then **Printers**.
5. Right-click the printer you want to set up a share to and select **Sharing** then **New Share**, as shown below. If you want to use this printer share to print PCL, PDF, or PostScript data on an IPDS printer, be sure to select an IPDS printer.

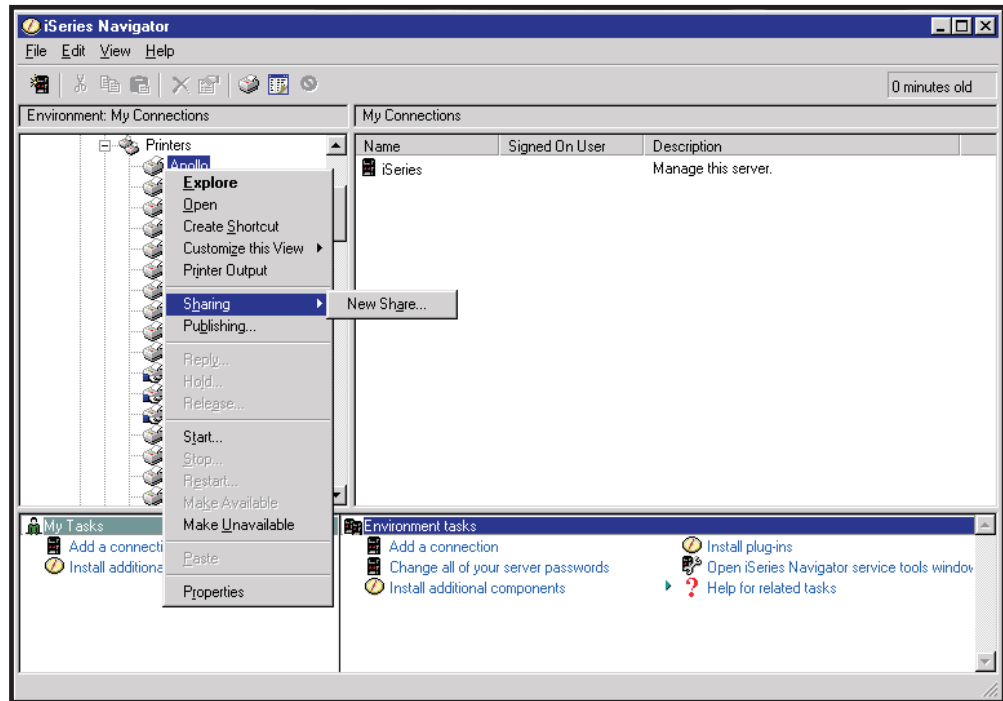


Figure 186. iSeries Access Components

6. The **iSeries NetServer Print Share** dialog opens, as shown in Figure 187 on page 276.
 - Specify a share name. This name should describe the type of job it will print (based on the printer file), such as legal or letter.
 - Give a detailed description of the share. This is optional.
 - Specify a driver. This is optional. If you specify a driver here, the driver must be available to you, and everyone who uses this share has to use this driver. If you do not specify a driver in this step, you have to specify one when you install the printer on your workstation.
 - Specify the library-qualified printer file this share uses.

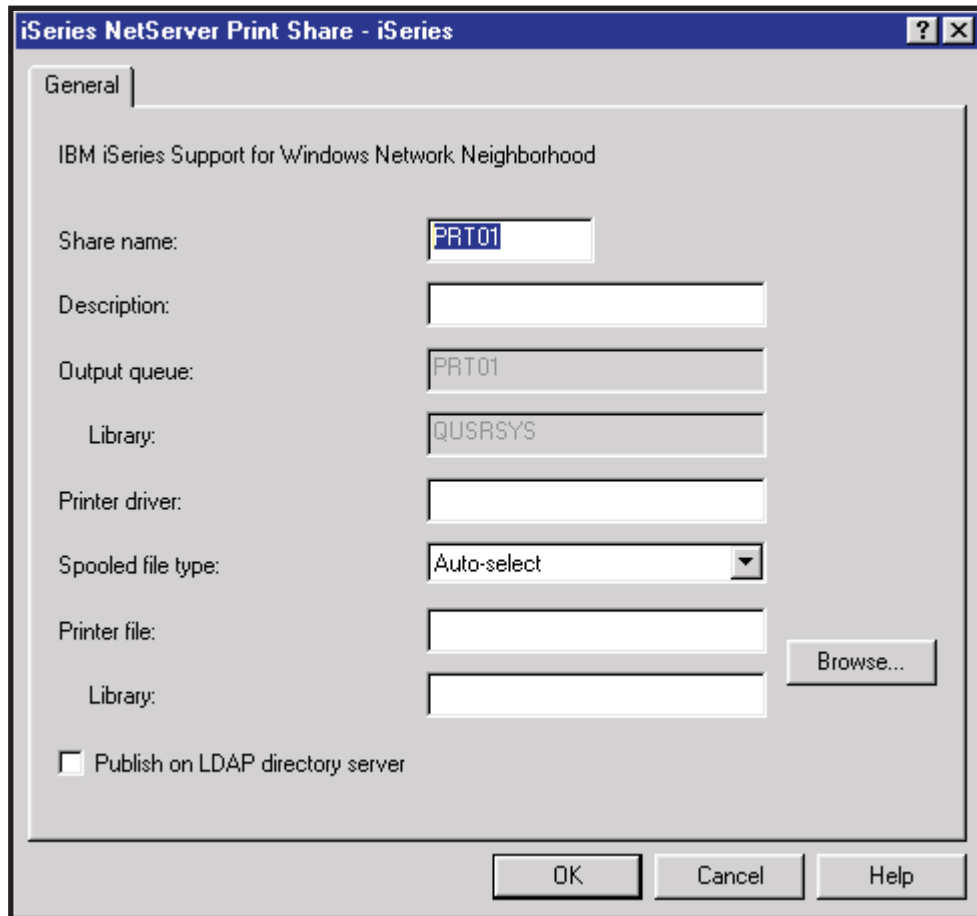


Figure 187. iSeries NetServer Print Share Dialog

7. Click **OK**.

Notes:

1. You can create multiple shares for the same printer.
2. You must create a different share for each printer file you want to use.
3. To transform jobs with this printer share, the printer device description must be configured appropriately.

You need to install the shared printer on your PC before you can use it.

Installing a Shared Printer

To install the printer on your PC, follow these steps.

Note: These steps are for a Windows NT operating system. The steps might be slightly different on different Windows operating systems. See the help for your operating system for steps to install a shared printer on a different operating system.

1. Display the printer shares on your iSeries by following these steps:
 - a. On your Windows system, click **Start**, select **Find**, then select **Computer...**
 - b. Enter the name of your iSeries and click **Find now**.
 - c. When your iSeries is found, double-click the icon by its name to open it.
 - d. All of the printers in the list are shared.
2. Right-click the printer share you want to install on your PC and select **Install**.

3. If you get the **Connect to Printer** message, click **OK** to install the driver on your PC.
4. Follow the steps in the **Add Printer** wizard. If you want to use this printer to print PCL, PDF, or PostScript data generated by your PC application, make sure that you specify a PCL, PDF, or PostScript driver.
5. The printer share shows up on your PC like any other printer. You can now send appropriate-type (depending on the driver you installed) PC jobs to this printer.

Specifying which Libraries to Display

When you display the contents of a container in iSeries Access, it displays the objects in the container's default library. To change the library used for display, follow these steps:

1. Double-click the iSeries Access icon.
2. Double-click the iSeries Navigator icon.
3. Open a connection to an iSeries system.
4. Expand the container whose objects you want to display.
5. From the **View** menu, select **Customize this view**, then **Include...**
6. Select a library from the list or specify a name and click **OK**. The library you specify becomes the default library for this container.

Chapter 24. Indexing, Archiving, and Retrieving Output with Content Manager OnDemand for iSeries

Content Manager OnDemand for iSeries (hereafter referred to as CM OnDemand) is an application solution to store large volumes of data and retrieve selective data, whether on disk, optical, or tape storage media. It provides computer output to laser disk (COLD) and extended archiving functions for the iSeries. With Version 5, there are two archive server environments for iSeries; CM OnDemand Spool File Archive and CM OnDemand Common Server. Over time, CM OnDemand Common Server will become the standard iSeries archive middleware.

CM OnDemand:

- Provides a functionally rich, cost effective application solution to store and retrieve large volumes of data
- Gives users the ability to find a specific document that may have originated as just one small part of a large spooled file
- Extracts index values, and compresses and stores data on disk, optical, or tape storage media
- Manages the life cycle requirements of data and automatically moves data to the appropriate media
- Helps reduce cost by eliminating the need to print to paper or microfiche, improving the usability of report information, increasing productivity, and enhancing customer service capabilities
- Offers a powerful administration system for data definition, management, and security

For more information about CM OnDemand, refer to *IBM Content Manager OnDemand for iSeries Common Server Administration Guide*.

Super Sun Seeds Case Study

CM OnDemand complements the Super Sun Seeds invoicing application well, providing a comprehensive system to manage documents after they have been created. For example, a user can define the spooled file of Super Sun Seeds invoices to CM OnDemand then CM OnDemand can automatically index, compress, and archive them to disk, optical, or tape media. A user can then easily retrieve the information, down to the individual invoice.

A user can also reprint or fax an invoice (using Facsimile Support for iSeries, for example). This saves the Super Sun Seeds corporation money since they will not need to print multiple copies of invoices for their records, send the data to microfiche, or keep large spooled files on their iSeries. More significantly, CM OnDemand improves productivity and customer responsiveness by providing the capability for customer service representatives, for example, to view, print, and fax invoices right from their desks while speaking with customers. At the same time, management can analyze sales summary reports across multiple months, quarters, or years, viewing only certain product lines or geographic areas, as needed.

Defining the Invoices to CM OnDemand

This example shows how the invoice spooled file is defined to CM OnDemand using a graphical administration system. The graphical administration system allows for point-and-click definition of index values and more, as well as setup of security authorizations, archive media, and “life of data” requirements.

The Super Sun Seeds invoices are defined to CM OnDemand one time, and then each time the spooled file of invoices is created (weekly or monthly, for example), it can be automatically captured and stored in CM OnDemand. There are a number of ways to automate the process so that no operator intervention is required.

Using the CM OnDemand Report Wizard is the most common way to begin the definition of the Super Sun Seeds invoices. The Report Wizard lets you define invoices to CM OnDemand in three steps:

1. Identify the type of spooled file data you are defining. In our Super Sun Seeds example, we choose AFP, as shown in Figure 188. CM OnDemand can store other data types as well, such as SCS, SCS-extended, Line, PDF, and “user-defined” data.

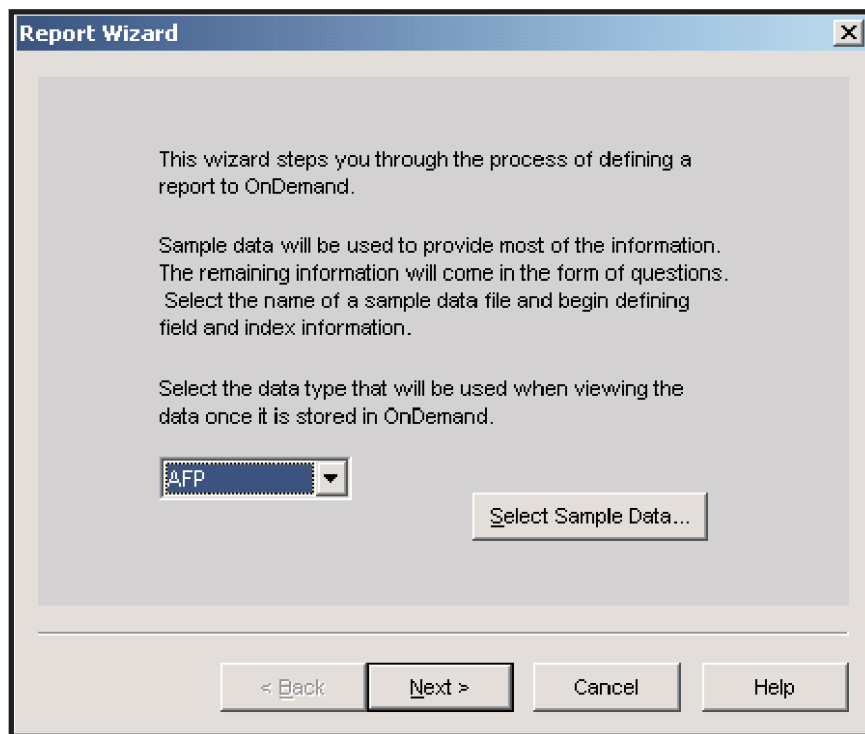


Figure 188. Starting the CM OnDemand Report Wizard

2. Invoke the graphical tool that lets you copy an iSeries spooled file down to your workstation so you can point-and-click to mark your actual iSeries print data.
3. Mark the index fields you require. The marking technique also lets you control how CM OnDemand separates the spooled file pages into individual invoices. The following section takes you through this process.

In the Super Sun Seeds example, you first use the Wizard to copy a spooled file of invoices to your workstation. Then you define the invoice number as a “trigger”, which can be used to tell CM OnDemand how to break up the spooled file into separate, multi-page invoices. CM OnDemand does not care how many pages belong to each invoice - that number can vary - as long as you tell CM OnDemand how to determine when one invoice ends and the next one begins within the spooled file. CM OnDemand does this “segmenting” so that users can retrieve a specific invoice rather than scrolling through the entire spooled file to find a particular one.

To define the invoice number as a trigger, highlight the invoice number, right-click, and select **Trigger**. This opens the “Add a Trigger” window. Figure 189 shows the trigger definition.

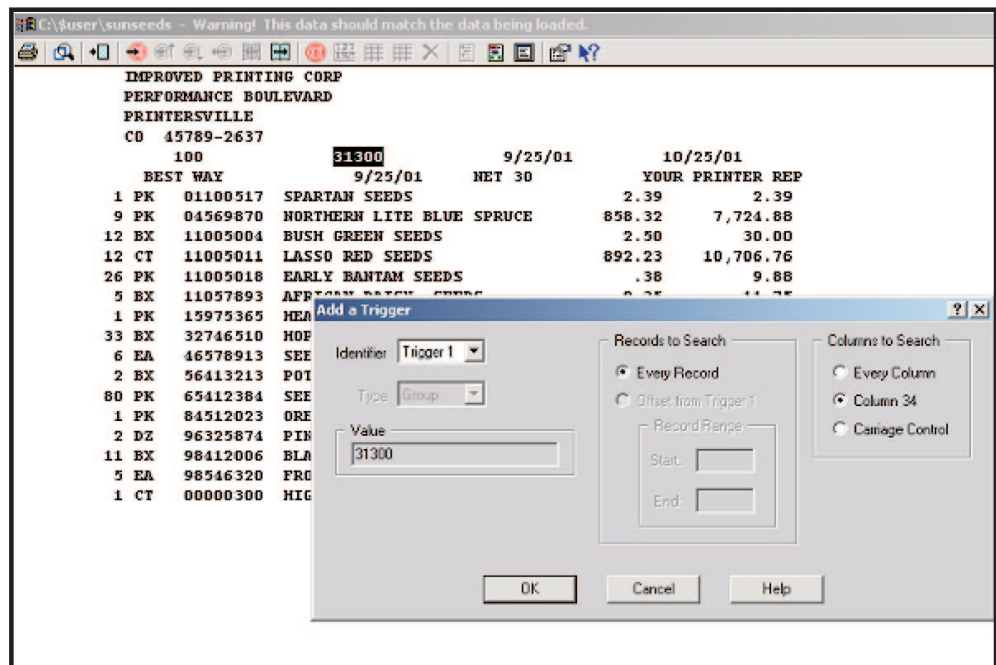


Figure 189. Adding a Trigger

Next, define the index fields, by marking the data that you want to be able to use for searching. Even if the index value is not always on the same line or in the same column within the invoice, CM OnDemand can find it.

You can define up to 32 index fields. In the example, you would mark invoice date, invoice number, customer number, and customer name. You can also define fields that would not typically be used for searching, but might be valuable for an end-user to see along with the index values on the hit list that results after entering invoice search criteria. In this example, we use invoice amount in this way.

To define the customer name as an index field, highlight the customer name, right-click, and select **Field**. The “Add a Field” window opens, as shown in Figure 190 on page 282. The customer name in this example is IMPROVED PRINTING CORP. Notice that “Index” was selected as the Database Field Type. The Folder Field Name is used with the Search Criteria and Document List, presented later.

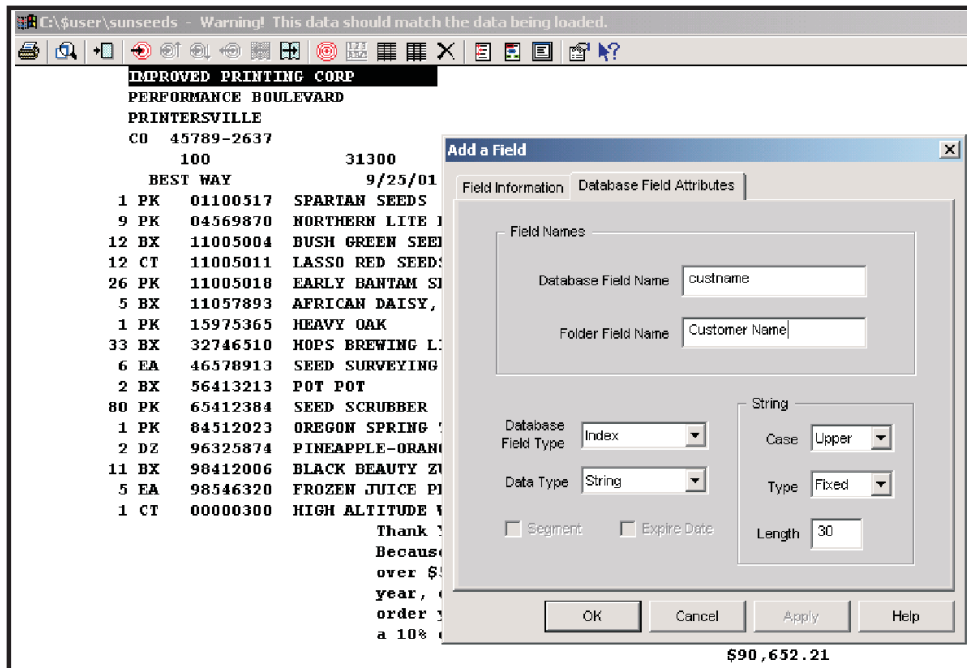


Figure 190. Defining an Index Field

After you have told CM OnDemand how to identify individual invoices and defined all your index fields, you use the Wizard to tell CM OnDemand what type of media and archive time frames you require. Then you name the definition; this is the name that your end users will use when they search the CM OnDemand archive for the invoices. In our example, we will use the name SUNSEEDS.

Retrieving the Archived Invoices from CM OnDemand

Once the invoices are stored, the real value of CM OnDemand becomes obvious, providing the capability to quickly find any invoice ever stored. Once found, an invoice can easily be viewed, printed, or faxed using the CM OnDemand Client.

After logging on to the server using the dialog box that appears when you click the CM OnDemand icon on your workstation desktop, a list appears with folders (reports) to which you are authorized. Figure 191 on page 283 shows a folder list.

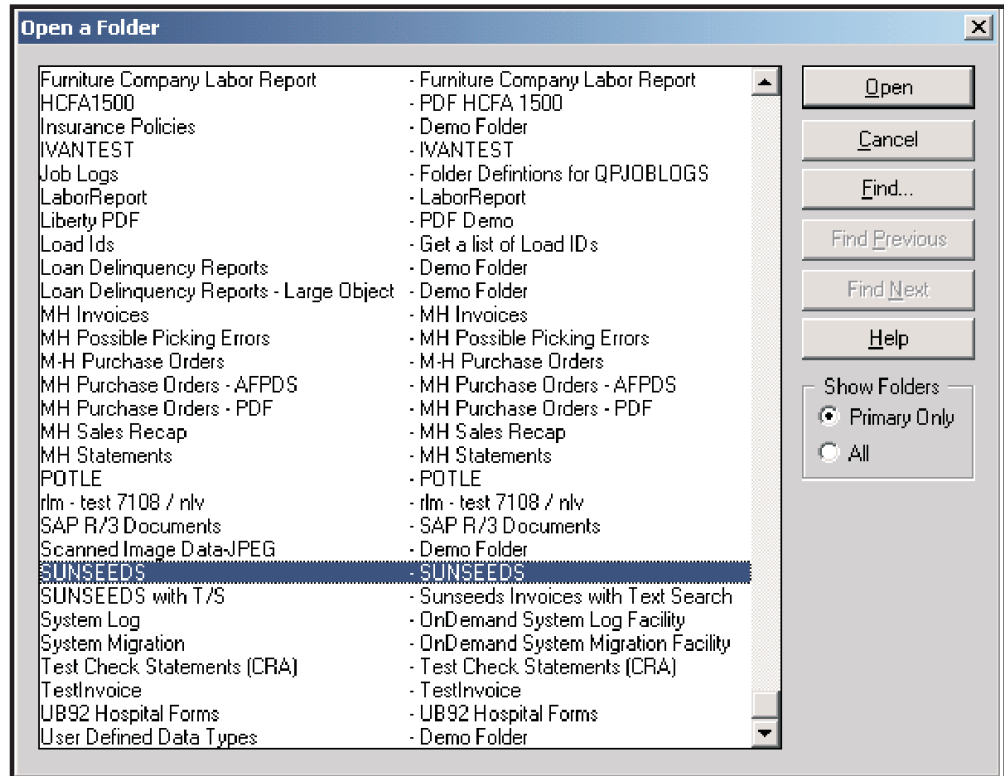


Figure 191. List of Folders

As a user searching for an invoice, the first step is to tell CM OnDemand what you are looking for. In the case study, you are looking for SUNSEEDS invoices, so double-click the SUNSEEDS folder, or select it and click **Open**.

On the resulting window, shown in Figure 192 on page 284, enter the search criteria needed to locate the archived invoices you want to retrieve. You might know the actual invoice number or you might need to review all the invoices for a particular customer number or customer name for the last six months (by expanding the date range).

To save keystrokes, you can use the Like operator with a “wildcard” (%) value. For example, you can enter IMP% for the customer name with the Like operator. This tells CM OnDemand to find any invoices for customers with names beginning with IMP (such as IMPROVED PRINTING CORP). Or, if you need to find all invoices for a particular date that were greater than a certain dollar amount, you can click on the operator button by Invoice Amount to see all available comparison operators (such as Equal To, Not Equal To, Like, Between, Greater Than, and Less Than) and choose Greater Than.

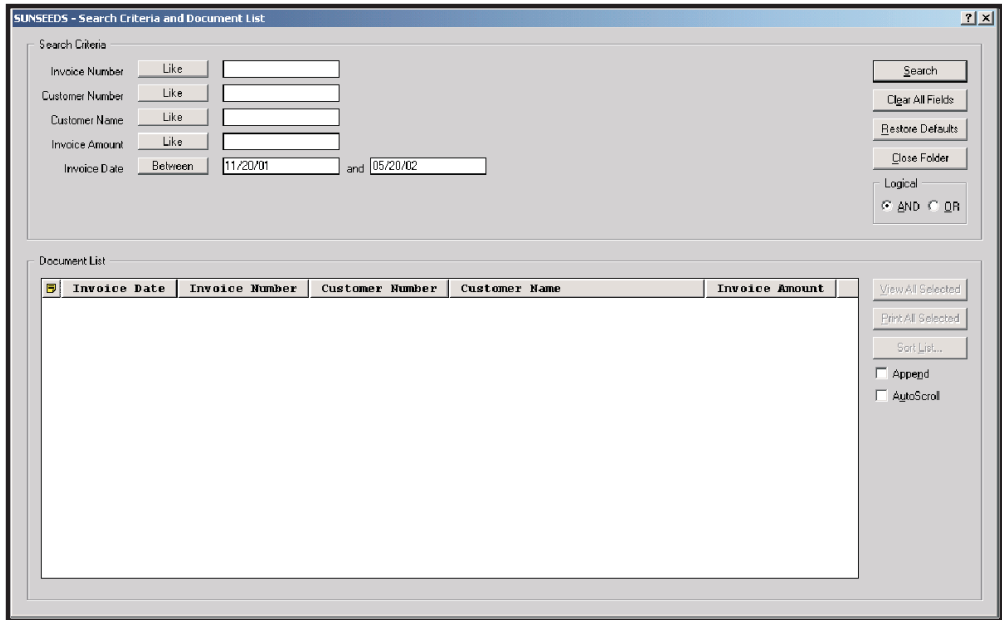


Figure 192. The Search Window

When you have entered your search criteria and chosen your search operator, click **Search**. CM OnDemand fills the documents list with the documents that satisfy your search criteria, as shown in Figure 193. Notice that this display shows the index values (invoice date, invoice number, customer number, and customer name) and the informational field (invoice amount).

It is possible that you can answer a significant number of customer inquiries about invoices simply by using this panel to tell a customer the total invoice amount, and then faxing or reprinting a copy for him.

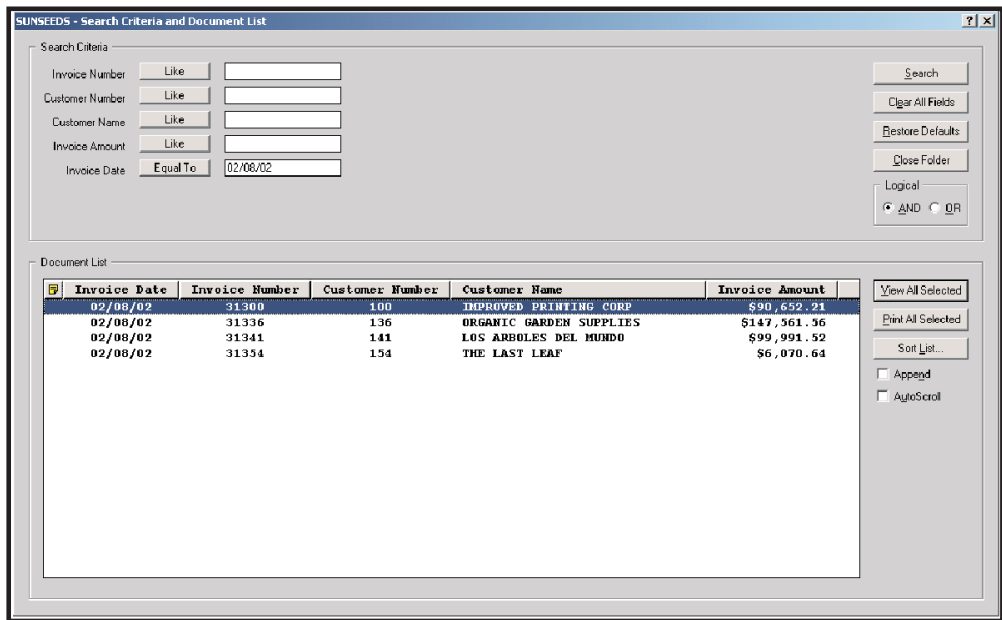


Figure 193. List of Invoices that Meet Search Criteria

From this list, you can either choose a single document to view or select multiple documents to view simultaneously. Figure 194 shows the selected invoice for Improved Printing Corporation in the viewer.

489 CPT Parkway
Fountain, NJ 08008

Super Sun Seeds

Office: 609-469-2307
Fax: 609-469-3904

IMPROVED PRINTING CORP
PERFORMANCE BOULEVARD
PRINCETONVILLE
CO 45789-2637

-- Sold To -- -- Ship To --

Customer: 100 Invoice #: 31300 Invoice Date: 1/29/02 Payment Date: 2/29/02

Ship Via: BEST WAY Shipped Date: 1/29/02 Terms: NET 30 REP: YOUR PRINTER REP

Qty	UOM	Item #	Item Description	Price	Extension
1	PK	01100517	SPARTAN SEEDS	2.39	2.39
9	PK	04569870	NORTHERN LITE BLUE SPRUCE	859.32	7,724.88
12	BX	11005004	BUSH GREEN SEEDS	2.50	30.00
12	CT	11005011	LASSO RED SEEDS	892.23	10,706.76
26	PK	11005018	EARLY BANTAM SEEDS	.36	9.88
5	BX	11057893	AMERICAN DAISY SEEDS	2.35	11.75
1	PK	15975365	HEAVY OAK	129.09	129.09
33	BX	32746510	HOPS BREWING LIGHT	1.20	39.60
6	EA	46578913	SEED SURVEYING SITE	50.00	300.00
2	BX	56413213	POT POT	7.65	15.30
80	PK	65412384	SEED SCRUBBER	888.79	71,103.20
1	PK	84512023	OREGON SPRING TOMATO SEED	.97	.97
2	DZ	94325374	PINEAPPLE-ORANGE SEEDS	1.29	2.58
11	BX	98412006	BLACK BEAUTY ZUCCHINI	2.30	25.30
5	EA	98546320	FROZEN JUICE PROCESSOR	109.90	549.50
1	CT	00000300	HIGH ALTITUDE WATERMELON	1.01	1.01

Thank You
Because you have ordered
over \$500 of seeds this
year, on your next seed
order you will receive
a 10% discount.

Ready Page 1 of 6 100% 12

Figure 194. Viewing an Invoice

The invoice, in its fully-resolved form, can be reprinted or faxed, if available, from the workstation or the iSeries server.

Chapter 25. Deciding on an Output Strategy and Set of Tools

We've covered a lot of territory, using our SuperSunSeeds.com case study to explore various tools for creating, printing, and delivering documents and reports. While you may have already mapped out your approach, here are several summary looks at the tools that might assist your decision.

First, let's take a look from a task perspective. Table 15 lists tasks you might want to accomplish and then gives a recommendation of what is the best tool to address that task. These Web pages are referenced in the table below:

IBM Printing Systems Professional Services Web page:

<http://www.printers.ibm.com/r5psc.nsf/web/ec>

iSeries Access for Web home page:

<http://www-1.ibm.com/servers/eserver/series/access/web/>

Table 15. Recommended Approaches — by Task

Task	Recommended Approach	Where Can I Find Information?
Work with images		
Deploy one logo	<ul style="list-style-type: none"> IBM AFP Printer Driver IBM or similar printing services 	<ul style="list-style-type: none"> "Creating an Image Resource with the IBM AFP Printer Driver for Windows" on page 91 IBM Printing Systems Professional Services Web page
Create efficient, high-performing iSeries images or perform ongoing image content work in iSeries applications	Infoprint Designer (Image Editor)	<ul style="list-style-type: none"> Chapter 17, "Using Infoprint Designer," on page 197 <i>Infoprint Designer for iSeries: Getting Started</i>
Complex embedding of images within documents such as bank statements	<ul style="list-style-type: none"> AFP Toolbox Other design tools such as DOC1 or StreamServe 	<ul style="list-style-type: none"> <i>AFP Toolbox for Multiple Operating Systems: User's Guide</i> Appropriate product documentation
Transform images to AFP	Infoprint Server	<i>Infoprint Server for iSeries: User's Guide</i>
Work with overlays		
Deploy one overlay	<ul style="list-style-type: none"> IBM AFP Printer Driver IBM or similar printing services 	<ul style="list-style-type: none"> "Creating an Image Resource with the IBM AFP Printer Driver for Windows" on page 91 IBM Printing Systems Professional Services Web page
Add an overlay on top of existing application output	Printer file	Chapter 13, "Using Printer Files," on page 121
Create and work with PDF		
Produce PDF	Infoprint Server	<ul style="list-style-type: none"> <i>Infoprint Server for iSeries: User's Guide</i> <i>IBM @server iSeries Printing VI: Delivering the Output of e-business</i>

Table 15. Recommended Approaches — by Task (continued)

Task	Recommended Approach	Where Can I Find Information?
Produce PDF and e-mail, including electronic segmentation	Infoprint Server	<ul style="list-style-type: none"> • <i>Infoprint Server for iSeries: User's Guide</i> • <i>IBM @server iSeries Printing VI: Delivering the Output of e-business</i>
Produce and work with PDF using iSeries Access	iSeries Access (requires Infoprint Server)	<ul style="list-style-type: none"> • "Converting a Spooled File to PDF for E-mailing, Spooling, or Storing" on page 269 • <i>Infoprint Server for iSeries: User's Guide</i>
Produce and work with PDF using iSeries Access for Web	iSeries Access for Web (uses Infoprint Server if installed)	<ul style="list-style-type: none"> • "Using iSeries Access for Web to Convert and View a Spooled File in PDF" on page 264 • iSeries Access for Web home page
Other		
Reengineer existing application output	Infoprint Designer (all editors)	<ul style="list-style-type: none"> • Chapter 17, "Using Infoprint Designer," on page 197 • <i>Infoprint Designer for iSeries: Getting Started</i>
Manage spooled files to printer queues	APU	<ul style="list-style-type: none"> • Chapter 15, "Using Advanced Print Utility," on page 153 • <i>Advanced Print Utility for iSeries: User's Guide</i>
Add 2D bar codes	<ul style="list-style-type: none"> • DDS • PPFA • IBM or similar services (for font implementation) 	<ul style="list-style-type: none"> • "Specifying Bar Codes with DDS" on page 104 • Chapter 17, "Using Infoprint Designer," on page 197 • <i>Page Printer Formatting Aid: User's Guide</i> • IBM Printing Systems Professional Services Web page
Format variable dynamic paragraphs	<ul style="list-style-type: none"> • AFP Toolbox • Other design tools such as DOC1 or StreamServe 	<ul style="list-style-type: none"> • <i>AFP Toolbox for Multiple Operating Systems: User's Guide</i> • Appropriate product documentation
Use record format line data	PPFA	<ul style="list-style-type: none"> • Chapter 16, "Using Page Printer Formatting Aid," on page 183 • <i>Page Printer Formatting Aid: User's Guide</i> <p>Chapter 16, "Using Page Printer Formatting Aid," on page 183</p>
Produce portable AFP for electronic distribution	Infoprint Server (requires AFP Viewer for viewing)	<i>Infoprint Server for iSeries: User's Guide</i>
Convert PCL, PostScript, and PDF files to AFP	Infoprint Server	<i>Infoprint Server for iSeries: User's Guide</i>

Table 15. Recommended Approaches — by Task (continued)

Task	Recommended Approach	Where Can I Find Information?
Print to IPDS printers	PSF	Printer Device Programming

Next, let's look at the formatting tools and break them down by key characteristics. With this perspective, you see that, for example, there may be certain applications for which DDS is the best fit, but DDS requires access to the application source code and programming resources. In contrast, Infoprint Designer can be used by a non-technical person and does not require access to source code or programming.

Table 16. Comparing iSeries Formatting Approaches

	Printer File	DDS	Infoprint Designer	APU	PPFA	AFP Toolbox	SAP Print
Designed for Non-IT User: Someone without programming skills can use it.	X		X	X			
Application Independent: Does not require recompile or change to existing application.	X		X	X	X		X
Dynamic Data Positioning: Position of data can vary from page to page		X	X	X	X	X	X ¹
Complex Documents: Able to support the full range of electronic documents, including overlays, fonts, bar codes, and images.		X	X	X	X	X	X ¹
Standard Bar Codes: Lets you map variable data as a standard bar code.		X	X	X	X	X	X
2D Bar Codes: Supports 2D barcodes such as PDF417, UPS Maxicode, and Data Matrix.		X			X		
Integrated, One-Pass Formatting: Does not require monitor and separate print engine.	X	X	X		X		X
Note: ¹ Depending on type of SAP documents (ABAP or Sapscript)							

Finally, let's go over each iSeries output tool and summarize the application or applications that it is the best fit for.

Table 17. Best Application of Each Tool

Tool	Application
AFP Toolbox	Complex image or paragraph formatting
APU	Monitoring and managing spooled files
DDS	Formatting complex, dynamic pages
Infoprint Designer	General purpose output design, without application changes
Infoprint Server	<ul style="list-style-type: none"> • Deliver output electronically • Format PC images for use in iSeries documents • Create Internet-ready AFP • Convert PostScript, PCL, and PDF to AFP
iSeries Access	Accessing PDF and e-mail functions interactively
PPFA	Using record format
Printer File	Adding an overlay on top of a printed page
PSF	Required for printing to IPDS printers
SAP R/3 Print	Formatting SAP output to AFP pages

Part 4. Appendixes

Appendix A. iSeries Font Libraries

These are the system font libraries:

QFNTCPL

This library is shipped with the operating system and contains the 240-pel compatibility set of fonts. (Look for **OS/400 - AFP Compatibility Fonts** in your licensed program list.) All other font libraries are part of IBM AFP Font Collection.

QFNT01-QFNT19

QFNT01-QFNT15 are used if you purchase and install certain iSeries single-byte character set font licensed program products. These libraries contain only 240-pel fonts.

QFNT61-QFNT69

QFNT61-QFNT65 are used if you purchase and install certain iSeries double-byte character set font licensed program products. These libraries contain only 240-pel fonts.

240 and 300 pel fonts can be in the same library if they have unique names. If you have the same font with the same name at both 240 and 300 pel, they need to be in separate libraries.

The IBM AFP Font Collection contains libraries that can be restored into system font libraries. The libraries are listed below:

QFNTCDEPAG	Expanded code pages
QFNTCPL	240-pel Compatibility fonts
QFNTCF_ARA	Arabic coded fonts
QFNTCF4ARA	4-character Arabic coded fonts
QFNT240ARA	240-pel Arabic character sets
QFNT300ARA	300-pel Arabic character sets
QFNT300CPL	300-pel Compatibility fonts
QFNTCF_CYR	Cyrillic Greek coded fonts
QFNTCF4CYR	4-character Cyrillic Greek coded fonts
QFNT240CYR	240-pel Cyrillic Greek character sets
QFNT300CYR	300-pel Cyrillic Greek character sets
QFNTCF_HEB	Hebrew coded fonts
QFNTCF4HEB	4-character Hebrew coded fonts
QFNT240HEB	240-pel Hebrew character sets
QFNT300HEB	300-pel Hebrew character sets
QFNTCF_OCR	OCR coded fonts
QFNTCF4OCR	4-character OCR coded fonts
QFNT240OCR	240-pel OCR character sets
QFNT300OCR	300-pel OCR character sets
QFNTCF_APL	APL coded fonts
QFNTCF4APL	APL 4-character coded fonts
QFNT240APL	240-pel APL character sets
QFNT300APL	300-pel APL character sets
QFNTCF_KAT	Katakana coded fonts
QFNTCF4KAT	Katakana 4-character coded fonts
QFNT240KAT	240-pel Katakana character sets
QFNT300KAT	300-pel Katakana character sets
QFNTCF_LA1	Latin1 coded fonts
QFNTCF4LA1	4-character Latin1 coded fonts

QFNT240LA1	240-pel character sets for typographic fonts, such as Helvetica and Times New Roman
QFNT300LA1	300-pel character sets for typographic fonts, such as Helvetica and Times New Roman
QFNTCF_LA2	Latin2/3/4/5 coded fonts
QFNTCF4LA2	4-character Latin2/3/4/5 coded fonts
QFNT240LA2	240-pel Latin2/3/4/5 character sets
QFNT300LA2	300-pel Latin2/3/4/5 character sets
QFNTCF_SYM	Symbols coded fonts
QFNTCF4SYM	4-character Symbols coded fonts
QFNT240SYM	240-pel Symbols character sets
QFNT300SYM	300-pel Symbols character sets
QFNT240LAO	240-pel Lao character sets
QFNT300LAO	300-pel Lao character sets
QFNTCF_LAO	Lao coded fonts
QFNTCF4LAO	4-character Lao coded fonts
QFNTOLNLAO	Lao AFP outlines
QFNTCFOLAO	Coded fonts for Lao AFP outlines
QFNTCO4LAO	4-character coded fonts for Lao outlines
QFNTCF_TAI	Thai coded fonts
QFNTCF4TAI	4-character Thai coded fonts
QFNT240TAI	240-pel Thai character sets
QFNT300TAI	300-pel Thai character sets
QFNTOLNTAI	Thai AFP outlines
QFNTCFOTAI	Coded fonts for Thai AFP outlines
QFNTCO4TAI	4-character coded fonts for Thai outlines
QFNT240BM	240-pel BookMaster character sets
QFNT300BM	300-pel BookMaster character sets
QFNTOLNAPL	APL AFP outlines
QFNTCFOAPL	Coded fonts for APL AFP outlines
QFNTCO4APL	4-character coded fonts for APL AFP outlines
QFNTOLNARA	Arabic AFP outlines
QFNTCFOARA	Coded fonts for Arabic AFP outlines
QFNTCO4ARA	4-character coded fonts for Arabic AFP outlines
QFNTOLNBM	BookMaster AFP outlines
QFNTOLNCYR	Cyrillic Greek AFP outlines
QFNTCFOCYR	Coded fonts for Cyrillic Greek AFP outlines
QFNTCO4CYR	4-character coded fonts for Cyrillic Greek AFP outlines
QFNTOLNHEB	Hebrew AFP outlines
QFNTCFOHEB	Coded fonts for Hebrew AFP outlines
QFNTCO4APL	4-character coded fonts for Hebrew AFP outlines
QFNTOLNKAT	Gothic Katakana AFP outlines
QFNTCFOKAT	Coded fonts for Gothic Katakana AFP outlines
QFNTCO4KAT	4-character coded fonts for Gothic Katakana AFP outlines
QFNTOLNLA1	Latin1 AFP outlines
QFNTCFOLA1	Coded fonts for Latin AFP outlines
QFNTCO4LA1	4-character coded fonts for Latin1 AFP outlines
QFNTOLNLA2	Latin2/3/5 AFP outlines
QFNTCFOLA2	Coded fonts for Latin2/3/5 AFP outlines
QFNTCO4LA2	4-character coded fonts for Latin2/3/5 AFP outlines
QFNTOLNOCR	OCR AFP outlines
QFNTCFOOCR	Coded fonts for OCR AFP outlines
QFNTCO4OCR	4-character coded fonts for OCR AFP outlines
QFNTOLNSYM	Symbols AFP outlines

QFNTCFOSYM
QFNTCO4SYM

Coded fonts for Symbols AFP outlines
4-character coded fonts for Symbols AFP outlines

An example of a working set of font libraries for a standard English installation might include these libraries:

- QFNTCPL
- QFNTCDEPAG
- QFNTCPL300
- QFNT240LA1
- QFNT300LA1
- QFNT240SYM
- QFNT300SYM
- QFNTCF
- QFNTOLNLA1
- QFNTOLNSYM

If you were using only outline fonts, then the font libraries would be:

- QFNTCDEPAG
- QFNTCF
- QFNTOLNLA1
- QFNTOLNSYM

Note that you can name the font libraries however you choose, these are just recommendations. What is important is that the user and resource library lists used by your applications refer to the needed libraries.

of being printed. This data structure stores up to 48 detail lines. A page is completed either through overflow (38 detail lines for page one, 48 details line for a continuation page) or through end of the customer invoice.

The fields PAGCNT and PAGTYP keep track of the page number and page type, and in combination, determine what kind of page is being printed. INVNEW2 executes the subroutine PAGSR for each page to be printed (customer copy, packing list, and file copy). In turn, the subroutine PRTDET is called to print the invoice line items out of the DETDS data structure. Indicator 51, set for the packing list, controls the suppression of the price and extension fields. A terms and conditions overlay (INVBAC) is printed only on the back of the customer copy. After all the copy pages have been printed, subroutine BLKDET is called to blank the DETDS data structure.

INVNEW2 RPG Source

The RPG source for INVNEW2 is as follows:

```

5722WDS V5R2M0          SEU SOURCE LISTING          11/11/03 17:47:37          1
SOURCE FILE . . . . . SAMPLER/QRPGSRC
MEMBER . . . . . INVNEW2
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
100      * INVNEW2 - Super Sun Seeds Electronic Invoice
200      *           Example 2
300      *
400      * This is an enhancement to INVNEW1 that adds the following
500      * output functionality:
600      *
700      *     1. Multiple copies of invoice with routing designation -
800      *           "Customer Copy", "Packing List", "File Copy"
900      *     2. Overlay specifying terms and conditions to print on back of
1000     *           customer copy.
1100     *     3. Packing List copy suppresses price information.
1200     *
1300     * This program reads a transaction file (SEEDDETL), retrieves info
1400     * from item master (SEEDITEM) and customer master (SEEDCUST), then
1500     * prints invoices.
1600     *
1700     * Different electronic overlays are used depending on whether this
1800     * is a one page or multi-page invoice, as well as whether there is
1900     * room remaining on the invoice to print a customer offer. The following
2000     * overlays are used:
2100     *
2200     *   INVALL      1-page invoice (full address at top, payment at bottom)
2300     *   INVST      Page 1 of multi-page invoice (no payment at bottom)
2400     *   INVMID     Continuation page (No address, payment at bottom)
2500     *   INVLST     Last page of multi-page invoice
2600     *   INVBAC     Back side Terms and Conditions overlay
2700     *
2800     * Invoice detail lines are processed until either end of customer order
2900     * or end of the page (38 lines for page 1 and 48 lines for continuation
3000     * pages), at which time the invoice page is printed. At end of
3100     * order, a check is made to see if there are more than 18 detail lines
3200     * to print. If there are, that would not leave room in the invoice body
3300     * to print the customer offer. In this case, the page is printed, and a
3400     * new final page is built with the offer and the payment coupon.
3500     *
3600     * The offer is a message (and corresponding image, or page segment) based
3700     * on item sales in the customer master. For example, a flower image
3800     * would print if seed sales exceeded $500.
3900     *
4000     * ENHANCEMENTS:
4100     * =====
4200     * Electronic printing offers wide opportunities for increasing
4300     * the value of your documents. Several additional enhancements

```



```

4400      *      that could be made to Example 2 include:
4500      *
4600      *      1. Use of "floating" forms. With DDS dynamic positioning of
4700      *      document elements (ie. line, box, segment, overlay, etc.),
4800      *      the invoice information could be floated. This means that
4900      *      the grid for detail lines is drawn as transactions are
5000      *      processed, and the grid is closed at the end of a specific
5100      *      customer's transactions. All other document elements
5200      *      could be similarly floated.
5300      *      2. Use of multiple input and output bins of the printer to
5400      *      utilize different stock for different pages. For example,
5500      *      the packing list might use stock that embeds a barcoded
5600      *      packing label on it.
5700      *
5800      FSEEDDETLIP E          K          DISK
5900      FSEEDITEMIF E          K          DISK
6000      FSEEDCUSTIF E          K          DISK
6100      FINVNEW2 0 E          PRINTER
6200      *
6300      E          WDS      1  40 24          Text for Offer
6400      E          CPY      1   3 25          Text for Copy
6500      *
6600      IINVCDETL  01
6700      I          STNAME          32
6800      I          CUST# L1
6900      ICUSTDATA  02
7000      IITEMDATA  03
7100      *
7200      * Data structure to store up to 48 detail lines
7300      *
7400      IDETDS      DS          48
7500      I          1  40QTY
7600      I          5  6 UOM
7700      I          7 140ITEM#
7800      I          15 39 ITMDES
7900      I          40 452SELP RC
8000      I          46 522EXTPRC
8100      *
8200      IBARTOT      DS
8300      I          1  60CUST#
8400      I          7 152TOTDUE
8500      *
8600      IPAYDAA      DS
8700      I          1  20PAYMO
8800      I          3  40PAYDA
8900      I          5  60PAYYR
9000      *
9100      * First pass only
9200      *
9300      C          *IN90      IFEQ '0'
9400      C          SETON          90
9500      C          EXSR DATESR
9600      C          EXSR BLKDET
9700      C          MOVEL 'Page'      PAGCON  4
9800      C          ENDIF
9900      *
10000     * Start of Customer
10100     *
10200     C          *INL1      IFEQ '1'
10300     C          CUST#      ADD 31200      INVC#  60      Invent Invoice#
10400     C          Z-ADD0      TOTDUE  92      Reset Totals/Ctrs
10500     C          Z-ADD0      ITMCNT  30
10600     C          Z-ADD0      PAGCNT  20
10700     C          MOVEL ' '      PAGTYP  3
10800     C          CUST#      CHAINSEEDCUST      20      Get Cust Master
10900     C          32      MOVEL ' Same'STNAME      Ship to Name Is Blank
11000     C          SETOF          30

```

11100	C		Z-ADDZIP	ZIPP	90		
11200	C		ENDIF				
11300	*						
11400	*		* Process invoice detail				
11500	*						
11600	C	*IN01	CABNE'1'	ENDD		Item Processing	
11700	C		Z-ADD0	SELPRC			
11800	C	ITM1	CHAINSEEDITEM		21		
11900	C	*IN21	CABEQ'1'	ENDD			
12000	C*						
12100	C		ADD 1	ITMCNT			
12200	C		Z-ADD0	EXTPRC			
12300	C	QTY1	MULT SELPRC	EXTPRC	72		
12400	C		Z-ADDQTY1	QTY	40		
12500	C	EXTPRC	ADD TOTDUE	TOTDUE			
12600	C*						
12700	C	ITMCNT	OCUR DETDS			Set up DS	
12800	C*						
12900	C	PAGCNT	IFEQ 0				
13000	C	ITMCNT	CABLT38	ENDD		38 lines on Page 1	
13100	C		ELSE				
13200	C	ITMCNT	CABLT48	ENDD		48 lines on Page n	
13300	C		ENDIF				
13400	C*						
13500	C		MOVE'OF '	PAGTYP	3		
13600	C		MOVEPCPY,1	PAGNAM	25		
13700	C		ADD 1	PAGCNT			
13800	C		EXSR PAGSR			Print Cust Copy	
13900	C*						
14000	C		SETON		51		
14100	C		MOVEPCPY,2	PAGNAM	25		
14200	C		EXSR PAGSR			Packing List	
14300	C		SETOF		51		
14400	C*						
14500	C		SETON		52		
14600	C		MOVEPCPY,3	PAGNAM	25		
14700	C		EXSR PAGSR			File Copy	
14800	C		SETOF		52		
14900	C*						
15000	C		Z-ADD0	ITMCNT			
15100	C		MOVE ' ' '	PAGTYP		Reg or OF Page Type	
15200	C		EXSR BLKDET				
15300	C	ENDD	TAG				
15400	C*						
15500	C*	* End of Customer Invoice					
15600	C*						
15700	CL1	ITMCNT	IFGT 18			No Room for PSEG	
15800	CL1		MOVE 'OF '	PAGTYP		OF page	
15900	CL1		MOVEPCPY,1	PAGNAM	25		
16000	CL1		ADD 1	PAGCNT			
16100	CL1		EXSR PAGSR			Customer Copy	
16200	C*						
16300	CL1		SETON		51		
16400	CL1		MOVEPCPY,2	PAGNAM	25		
16500	CL1		EXSR PAGSR			Packing List	
16600	CL1		SETOF		51		
16700	C*						
16800	CL1		SETON		52		
16900	CL1		MOVEPCPY,3	PAGNAM	25		
17000	CL1		EXSR PAGSR			File Copy	
17100	CL1		SETOF		52		
17200	CL1		EXSR BLKDET			Blank DS	
17300	CL1		ENDIF				
17400	C*						
17500	CL1		MOVE BARTOT	BARPRC	150	Load Totals	
17600	CL1		Z-ADDTOTDUE	TOTD02	92		
17700	CL1		MOVE NAME	NAME02	25		

17800	CL1		MOVE STREET	STRE@2	25	
17900	CL1		MOVE CITY	CITY@2	25	
18000	CL1		MOVE STATE	STAT@2	2	
18100	CL1		MOVE ZIP	ZIP@2	9	
18200	C*					
18300	CL1		EXSR OFFSR			
18400	CL1		MOVE 'END'	PAGTYP		Last Page
18500	CL1		MOVE LCPY,1	PAGNAM		
18600	CL1		ADD 1	PAGCNT		
18700	CL1		EXSR PAGSR			Customer Copy
18800	C*					
18900	CL1		SETON		51	
19000	CL1		MOVE LCPY,2	PAGNAM		
19100	CL1		EXSR PAGSR			Packing List
19200	CL1		SETOF		51	
19300	C*					
19400	CL1		SETON		52	
19500	CL1		MOVE LCPY,3	PAGNAM		
19600	CL1		EXSR PAGSR			File Copy
19700	CL1		SETOF		52	
19800	C*					
19900	C*		Based on page number and page type, determines what overlay to			
20000	C*		use			
20100	C*					
20200	CSR		PAGSR	BEGSR		
20300	C		PAGTYP	IFEQ 'OF '		
20400	C		PAGCNT	IFEQ 1		
20500	C		MOVE 'INVST	'OVLAY	8	
20600	C		WRITEINVTOP			
20700	C		ELSE			
20800	C		WRITEINVTP2			
20900	C		MOVE 'INVMID	'OVLAY		
21000	C		ENDIF			
21100	C		ENDIF			
21200	C*					
21300	C		PAGTYP	IFEQ 'END'		
21400	C		PAGCNT	IFEQ 1		
21500	C		MOVE 'INVALL	'OVLAY	8	
21600	C		WRITEINVTOP			
21700	C		ELSE			
21800	C		WRITEINVTP2			
21900	C		MOVE 'INVLST	'OVLAY		
22000	C		ENDIF			
22100	C		ENDIF			
22200	C*					
22300	C		EXSR PRDDET			Print details
22400	C		WRITEPAGEOF			Pg 1 of n Msg
22500	C	30	WRITEOFFER			Write Offer
22600	C	30	WRITEPAGESEG			Write Segment
22700	C		PAGTYP	IFEQ 'END'		
22800	C		WRITEINVBOT			Invoice Totals
22900	C		ENDIF			
23000	C		WRITEPRTOVL			Print Overlay
23100	C		WRITEENDPAG			End Page
23200	C*					
23300	C		PAGNAM	IFEQ CPY,1		
23400	C		MOVE 'INVBAC	'OVLAY	8	Back Overlay
23500	C		WRITEPRTOVL			
23600	C		ENDIF			
23700	C		WRITEENDPAG			
23800	C*					
23900	CSR		ENDP	ENDSR		
24000	C*					
24100	C*		Print invoice detail lines from data structure			
24200	C*					
24300	CSR		PRTDET	BEGSR		
24400	C		Z-ADD1	X	20	

```

24500 C          1      DO 48      X
24600 C          X      OCUR DETDS
24700 C          ITEM#  IFGT 0
24800 C          WRITEDETLIN
24900 C          ENDIF
25000 C          END
25100 CSR         ENDSR
25200 C*
25300 C* Blank out data structure
25400 C*
25500 CSR         BLKDET  BEGSR
25600 C          1      DO 48      X
25700 C          X      OCUR DETDS
25800 C          MOVEL*BLANKS UOM
25900 C          MOVEL*BLANKS ITMDES
26000 C          Z-ADD0    QTY
26100 C          Z-ADD0    ITEM#
26200 C          Z-ADD0    SELPRC
26300 C          Z-ADD0    EXTPRC
26400 C          END
26500 CSR         ENDSR
26600 C* Set up Date
26700 CSR         DATESR  BEGSR
26800 C          Z-ADDUDAY  PAYDA
26900 C          UMONTH  ADD 1    PAYMO
27000 C          PAYMO   IFGT 12
27100 C          Z-ADD1    PAYMO
27200 C          UYEAR   ADD 1    PAYYR
27300 C          ELSE
27400 C          Z-ADDUYEAR PAYYR
27500 C          ENDIF
27600 C          MOVE PAYDAA  PAYDAT 60
27700 C          Z-ADDPAYDAT PAYDA@ 60
27800 CSR         ENDSR
27900 *
28000 * Set up Offer at end of invoice
28100 *
28200 CSR         OFFSR   BEGSR
28300 C          SLSSSED  IFGE 500
28400 C          Z-ADD1    IX      20
28500 C          MOVE 'FLWRNB 'PSEG 8      30      Seed
28600 C          SETON
28700 C          GOTO WRTOFR
28800 C          ENDIF
28900 C          SLSFRT   IFGE 500
29000 C          Z-ADD7    IX
29100 C          MOVE 'STRWNB 'PSEG
29200 C          SETON
29300 C          GOTO WRTOFR
29400 C          ENDIF
29500 C          SLSSUP   IFGE 500
29600 C          Z-ADD13   IX
29700 C          MOVE 'TREENB 'PSEG
29800 C          SETON
29900 C          GOTO WRTOFR
30000 C          ENDIF
30100 C          SLSCHM  IFGE 500
30200 C          Z-ADD19   IX
30300 C          MOVE 'BETLNB 'PSEG
30400 C          SETON
30500 C          GOTO WRTOFR
30600 C          ENDIF
30700 C*
30800 C          GOTO ENDOFR
30900 C*
31000 C          WRTOFR  TAG
31100 C          MOVE WDS,IX OFFR@1 24      Build Offer Text

```

```

31200      C              ADD 1      IX
31300      C              MOVE WDS,IX OFFR@2 24
31400      C              ADD 1      IX
31500      C              MOVE WDS,IX OFFR@3 24
31600      C              ADD 1      IX
31700      C              MOVE WDS,IX OFFR@4 24
31800      C              ADD 1      IX
31900      C              MOVE WDS,IX OFFR@5 24
32000      C              ADD 1      IX
32100      C              MOVE WDS,IX OFFR@6 24
32200      CSR           ENDOFR      ENDSR
32300      C*
32400      *
32500 ** WDS      WORDS FOR OFFER
32600 Thank You .....
32700 Because you have ordered
32800 over $500 of seeds this
32900 year, on your next seed
33000 order you will receive
33100 a 10% discount.
33200 Thank You .....
33300 Because you have ordered
33400 over $500 of fruit this
33500 year, on your next fruit
33600 order you will receive
33700 a 10% discount.
33800 Thank You .....
33900 Because you have ordered
34000 over $500 of trees this
34100 year, on your next tree
34200 order you will receive
34300 a 10% discount.
34400 Thank You .....
34500 Because you have ordered
34600 over $500 of stuff this
34700 year, on your next stuff
34800 order you will receive
34900 a 10% discount.
35000
35100 ** CPY      Copy Name
35200 Customer Copy
35300 Packing List
35400 File Copy

```

* * * * E N D O F S O U R C E * * * *

INVNEW2 DDS Source

The DDS source for INVNEW2 is as follows:

```

5722WDS V5R2M0              SEU SOURCE LISTING              11/11/03 17:47:37      1
SOURCE FILE . . . . . SAMPLER/QDDSSRC
MEMBER . . . . . INVNEW2
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
 100      A* INVNEW2 - Printer File DDS for Super Sun Seeds Invoice
 200      A*              Example 2 (copies and duplex function added)
 300      A*
 400      A* Printer File Create or Change Parameters:
 500      A* - Printer Device Type (DEVTYPE)          *AFPDS
 600      A* - Overflow Line (OVRFLW)                 64
 700      A* - Print on Both Sides (DUPLEX)          *YES
 800      A*       if running on duplex printer
 900      A*
1000      A* Page 1 Header
1100      A*- includes Postnet Zip+4
1200      A*
1300      A          R INVTOP              SKIPB(10)
1400      A          ZIPPN              9S      12 BARCODE(POSTNET)
1500      A              SPACEA(2)

```

1600	A	NAME	25A	12	
1700	A	STNAME	25A	48	
1800	A				SPACEA(1)
1900	A	STREET	25A	12	
2000	A	STSTRT	25A	48	
2100	A				SPACEA(1)
2200	A	CITY	25A	12	
2300	A	STCITY	25A	48	
2400	A				SPACEA(1)
2500	A	STATE	2A	12	
2600	A	ZIP	9S	16	EDTWRD(' -')
2700	A	STSTE	2A	48	
2800	A	STZIP	9S	52	EDTWRD(' -')
2900	A				SPACEA(3)
3000	A	CUST#	6S 0	14	EDTCDE(Z)
3100	A	INVC#	6S 0	32	EDTCDE(Z)
3200	A				49DATE EDTCDE(Y)
3300	A	PAYDAT	6S 0	66	EDTCDE(Y)
3400	A				SPACEA(2)
3500	A	SHPVIA	10A	14	
3600	A				34DATE EDTCDE(Y)
3700	A	TERMS	10A	47	
3800	A	SLSMAN	16A	64	
3900	A				SPACEA(4)
4000	A*				
4100	A*	Page 2 Header			
4200	A*				
4300	A	R INVTP2			SKIPB(10)
4400	A	NAME	25A	12	
4500	A				SPACEA(2)
4600	A	CUST#	6S 0	14	EDTCDE(Z)
4700	A	INVC#	6S 0	32	EDTCDE(Z)
4800	A				49DATE EDTCDE(Y)
4900	A	PAYDAT	6S 0	66	EDTCDE(Y)
5000	A				SPACEA(4)
5100	A*				
5200	A*	Detail Lines			
5300	A*				
5400	A	R DETLIN			SPACEA(1)
5500	A	QTY	4S 0	8	EDTCDE(Z)
5600	A	UOM	2A	13	
5700	A	ITEM#	8S 0	18	
5800	A	ITMDES	25A	28	
5900	A N51	SELPRC	6S 2	58	EDTCDE(J)
6000	A N51	EXTPRC	7S 2	70	EDTCDE(J)
6100	A*				
6200	A*	Multipage Message			
6300	A*				
6400	A	R PAGEOF			
6500	A	PAGCON	4A		POSITION(10.7 7.3)
6600	A				FNTCHRSET(C0H200A0 T1V10037)
6700	A	PAGCNT	2S 0		POSITION(10.7 7.8)
6800	A				FNTCHRSET(C0H200A0 T1V10037)
6900	A				EDTCDE(Z)
7000	A	PAGNAM	25A		POSITION(10.7 3.8)
7100	A				FNTCHRSET(C0H400B0 T1V10037)
7200	A*				
7300	A*	Invoice Totals			
7400	A*	- includes Interleaf 2 of 5 barcode			
7500	A*				
7600	A	R INVBOT			SKIPB(51)
7700	A N51	TOTDUE	9S 2	67	EDTWRD(' , , \$0. -')
7800	A				SPACEA(4)
7900	A 51			67	' '
8000	A				SPACEA(4)
8100	A	PAYDA0	6S 0	25	EDTCDE(Y)
8200	A N51	TOTD02	9S 2	67	EDTWRD(' , , \$0. -')

```

8300      A                               SPACEA(2)
8400      A 51                            67 ' '
8500      A                               SPACEA(2)
8600      A          NAME@2          25A    12
8700      A                               SPACEA(1)
8800      A          STRE@2          25A    12
8900      A          BARPRC          15S 0   52BARCODE(INTERL20F5 3)
9000      A                               SPACEA(1)
9100      A          CITY@2          25A    12
9200      A                               SPACEA(1)
9300      A          STAT@2           2A     12
9400      A          ZIP@2            9A     16
9500      A*
9600      A* Offer Print
9700      A* - Font 92 is Courier Italic 12-pitch
9800      A*
9900      A          R OFFER                               SKIPB(43)
10000     A                               FONT(92)
10100     A          OFFR@1          24A    36
10200     A                               SPACEA(1)
10300     A          OFFR@2          24A    36
10400     A                               SPACEA(1)
10500     A          OFFR@3          24A    36
10600     A                               SPACEA(1)
10700     A          OFFR@4          24A    36
10800     A                               SPACEA(1)
10900     A          OFFR@5          24A    36
11000     A                               SPACEA(1)
11100     A          OFFR@6          24A    36
11200     A                               SPACEA(1)
11300     A*
11400     A* Images/Page Segments
11500     A* - variable page segment name from program
11600     A*
11700     A          R PAGSEG                               PAGSEG(&PSEG 7.0 2.6)
11800     A          PSEG            8A P
11900     A*
12000     A*
12100     A* Images/Page Segments
12200     A* - variable overlay name from program
12300     A*
12400     A          R PRTOVL                               OVERLAY(&OVLAY 0 0)
12500     A          OVLAY            8A P
12600     A*
12700     A* Forces page advance
12800     A*
12900     A          R ENDPAG                               ENDPAGE
                                     * * * * E N D   O F   S O U R C E * * * *

```

INVNEW3 Floating Super Sun Seeds Invoicing

This version of the Super Sun Seeds invoice demonstrates a dynamic or “floating” document. It uses the capabilities of variable or dynamic positioning within DDS. The invoice is not limited by the structure of the electronic overlay, but is tailored by the customer data.

Figure 196 on page 306 shows a sample of the output for INVNEW3.

The INVNEW3 program is very similar to INVNEW2, except that it uses variations of the overlays - INVHEAD, INVHEAD2, and INVPAY. It also includes printer file writes as follows:

DTGRID

Extends the vertical lines .167 inches (6 lines per inch) with each detail line

CLOSE

Draws the horizontal line to close the transaction grid

TOTAL

Draws the total box, and prints the totals

INVNEW3 RPG Source

The INVNEW3 RPG source is shown below:

```

5722WDS V5R2M0                SEU SOURCE LISTING                11/11/03 17:47:37      1
SOURCE FILE . . . . . SAMPLER/QRPGSRC
MEMBER . . . . . INVNEW3
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
100      * INVNEW3 - Super Sun Seeds Electronic Invoice
200      *           Example 3 - Floating Invoice
300      *
400      * This is an enhancement to INVNEW1 and INVNEW2 that adds the
500      * following function:
600      *
700      *     1. Multiple copies of invoice with routing designation -
800      *         "Customer Copy", "Packing List", "File Copy"
900      *     2. Overlay specifying terms and conditions to print on back of
1000     *         customer copy.
1100     *     3. Packing List copy suppresses price information.
1200     *     4. The invoice is a floating document. Only the invoice
1300     *         header and payment coupon are static overlays. The
1400     *         main part of the invoice is built dynamically using
1500     *         line commands. Output format DTGRID extends the vertail
1600     *         lines of the invoice detail area down. Output format
1700     *         CLOSE draws the horizontal line to close the grid. Format
1800     *         TOTAL draws the "Total Due" box.
1900     *
2000     * This program reads a transaction file (SEEDDETL), retrieves info
2100     * from item master (SEEDITEM) and customer master (SEEDCUST), then
2200     * prints invoices.
2300     *
2400     * Different electronic overlays are used depending on whether this
2500     * is a one page or multi-page invoice, as well as whether there is
2600     * room remaining on the invoice to print a customer offer. The following
2700     * overlays are used:
2800     *
2900     *     INVHEAD    1-page invoice header
3000     *     INVHEAD2  Continuation page invoice header
3100     *     INVPAY    Payment coupon at bottom
3200     *     INVBAC    Back side Terms and Conditions overlay
3300     *
3400     * Invoice detail lines are processed until either end of customer order
3500     * or end of the page (38 detail lines for page 1, 48 detail lines for
3600     * the continuation pages) is reached. At the end of an order,
3700     * a check is made to see if there are more than 18 detail lines left
3800     * to print. If there are, that would not leave room in the invoice body
3900     * to print the customer offer. In this case, the page is printed, and a
4000     * new final page is built with the offer and the payment coupon.
4100     *
4200     * The offer is a message (and corresponding image, or page segment) based
4300     * on item sales in the customer master. For example, a flower image
4400     * would print if seed sales exceeded $500.
4500     *
4600     *
4700     FSEEDDETLIP E           K           DISK
4800     FSEEDITEMIF E          K           DISK

```

```

4900 FSEEDCUSTIF E          K          DISK
5000 FINVNEW3 0 E          PRINTER
5100 *
5200 E          WDS      1  40 24          Text for Offer
5300 E          CPY      1   3 25          Text for Copy
5400 *
5500 IINVCDETL  01
5600 I          STNAME          32
5700 I          CUST# L1
5800 ICUSTDATA  02
5900 IITEMDATA  03
6000 *
6100 * Data structure to store up to 48 detail lines
6200 *
6300 IDETDS      DS          48
6400 I          1  40QTY
6500 I          5   6 UOM
6600 I          7 140ITEM#
6700 I          15 39 ITMDES
6800 I          40 452SELPRC
6900 I          46 522EXTPRC
7000 *
7100 * Cust# and Total Amount to print in bar code
7200 *
7300 IBARTOT      DS
7400 I          1  60CUST#
7500 I          7 152TOTDUE
7600 *
7700 IPAYDAA      DS
7800 I          1  20PAYMO
7900 I          3  40PAYDA
8000 I          5  60PAYYR
8100 *
8200 * First pass only
8300 *
8400 C          *IN90  IFEQ '0'
8500 C          SETON          90
8600 C          EXSR DATESR
8700 C          EXSR BLKDET
8800 C          MOVEL 'Page'  PAGCON  4
8900 C          MOVEL 'Total' TOTCON  9
9000 C          MOVE 'Due'   TOTCON  9
9100 C          ENDIF
9200 *
9300 * Start of Customer
9400 *
9500 C          *INL1  IFEQ '1'
9600 C          CUST#  ADD 31200  INVC#  60  Invent Invoice#
9700 C          Z-ADD0  TOTDUE  92  Reset Totals/Ctrs
9800 C          Z-ADD0  ITMCNT  30
9900 C          Z-ADD0  PAGCNT  20
10000 C          MOVEL ' '  PAGTYP  3
10100 C          CUST#  CHAINSEEDCUST  20  Get Cust Master
10200 C          32  MOVEL ' '  SAME 'STNAME
10300 C          Z-ADDZIP  ZIPPN  90
10400 C          SETOF          30
10500 C          ENDIF
10600 *
10700 * Process invoice detail
10800 *
10900 C          *IN01  CABNE '1'  ENDDET
11000 C          Z-ADD0  SELPRC
11100 C          ITM1  CHAINSEEDITEM  21
11200 C          *IN21  CABEQ '1'  ENDDET
11300 C*
11400 C          ADD 1  ITMCNT
11500 C          Z-ADD0  EXTPRC

```

11600	C	QTY1	MULT SELPRC	EXTPRC	72		
11700	C		Z-ADDQTY1	QTY	40		
11800	C	EXTPRC	ADD TOTDUE	TOTDUE			
11900	C*						
12000	C*	Load item detail fields into data structure					
12100	C*						
12200	C	ITMCNT	OCUR DETDS			Set up DS	
12300	C*						
12400	C*	Check end of page					
12500	C*						
12600	C	PAGCNT	IFEQ 0				
12700	C	ITMCNT	CABLT39	ENDDT		35 lines on P1	
12800	C		ELSE				
12900	C	ITMCNT	CABLT48	ENDDT		48 lines on Pn	
13000	C		ENDIF				
13100	C*						
13200	C*	If end of page, print					
13300	C*						
13400	C		MOVE 'OF '	PAGTYP	3		
13500	C		MOVE LCPY,1	PAGNAM	25		
13600	C		ADD 1	PAGCNT			
13700	C		EXSR PAGSR			Print Cust Copy	
13800	C*						
13900	C		SETON		51		
14000	C		MOVE LCPY,2	PAGNAM	25		
14100	C		EXSR PAGSR			Packing List	
14200	C		SETOF		51		
14300	C*						
14400	C		SETON		52		
14500	C		MOVE LCPY,3	PAGNAM	25		
14600	C		EXSR PAGSR			File Copy	
14700	C		SETOF		52		
14800	C*						
14900	C		Z-ADD0	ITMCNT			
15000	C		MOVE ' ' '	PAGTYP		Reg or OF Page	
15100	C		EXSR BLKDET				
15200	C	ENDDT	TAG				
15300	C*						
15400	C*	End of Customer Invoice					
15500	C*						
15600	CL1	ITMCNT	IFGT 18			No Room for PSEG	
15700	CL1		MOVE 'OF '	PAGTYP		OF page	
15800	CL1		MOVE LCPY,1	PAGNAM	25		
15900	CL1		ADD 1	PAGCNT			
16000	CL1		EXSR PAGSR			Customer Copy	
16100	C*						
16200	CL1		SETON		51		
16300	CL1		MOVE LCPY,2	PAGNAM	25		
16400	CL1		EXSR PAGSR			Packing List	
16500	CL1		SETOF		51		
16600	C*						
16700	CL1		SETON		52		
16800	CL1		MOVE LCPY,3	PAGNAM	25		
16900	CL1		EXSR PAGSR			File Copy	
17000	CL1		SETOF		52		
17100	C*						
17200	CL1		EXSR BLKDET			Blank DS	
17300	CL1		ENDIF				
17400	C*						
17500	C*	Setup payment coupon fields					
17600	C*						
17700	CL1		MOVE BARTOT	BARPRC	150		
17800	CL1		Z-ADDTOTDUE	TOTD02	92		
17900	CL1		MOVE NAME	NAME@2	25		
18000	CL1		MOVE STREET	STRE@2	25		
18100	CL1		MOVE CITY	CITY@2	25		
18200	CL1		MOVE STATE	STAT@2	2		

18300	CL1		MOVE ZIP	ZIP@2	9	
18400	C*					
18500	CL1		EXSR OFFSR			
18600	CL1		MOVE 'END'	PAGTYP		Last Page
18700	CL1		MOVE LCPY,1	PAGNAM		
18800	CL1		ADD 1	PAGCNT		
18900	CL1		EXSR PAGSR			Customer Copy
19000	C*					
19100	CL1		SETON		51	
19200	CL1		MOVE LCPY,2	PAGNAM		
19300	CL1		EXSR PAGSR			Packing List
19400	CL1		SETOF		51	
19500	C*					
19600	CL1		SETON		52	
19700	CL1		MOVE LCPY,3	PAGNAM		
19800	CL1		EXSR PAGSR			File Copy
19900	CL1		SETOF		52	
20000	C*					
20100	C*		Based on page number and page type, determines what overlay to			
20200	C*		use			
20300	C*					
20400	CSR	PAGSR	BEGSR			
20500	C	PAGTYP	IFEQ 'OF '			
20600	C	PAGCNT	IFEQ 1			
20700	C		MOVE 'INVHEAD 'OVRLAY	8		Full Header
20800	C		WRITEINVTOP			
20900	C		ELSE			
21000	C		WRITEINVTP2			
21100	C		MOVE 'INVHEAD2'OVRLAY			Abbrev. Header
21200	C		ENDIF			
21300	C		ENDIF			
21400	C*					
21500	C	PAGTYP	IFEQ 'END'			
21600	C	PAGCNT	IFEQ 1			
21700	C		MOVE 'INVHEAD 'OVRLAY	8		Full Header
21800	C		WRITEINVTOP			
21900	C		ELSE			
22000	C		WRITEINVTP2			
22100	C		MOVE 'INVHEAD2'OVRLAY			Abbrev. Header
22200	C		ENDIF			
22300	C		ENDIF			
22400	C*					
22500	C		EXSR PRTDET			Print details
22600	C		WRITEDTGRID			Extend grid
22700	C		ADD .167	DWN		.167 inch down
22800	C		WRITECLOSE			Close grid
22900	C		WRITEPAGEOF			Pg 1 of n Msg
23000	C	30	WRITEOFFER			Write Offer
23100	C	30	WRITEPAGSEG			Write Segment
23200	C		WRITEPRTOVL			Print Overlay
23300	C*					
23400	C	PAGTYP	IFEQ 'END'			
23500	C		WRITEINVBOT			Invoice Totals
23600	C		MOVE 'INVPAY 'OVRLAY	8		
23700	C		WRITEPRTOVL			Print Overlay
23800	C	DWN	ADD .5	DWN2		Offset - Total
23900	C	DWN	ADD .35	DWN3		Box
24000	C		WRITETOTAL			
24100	C		ENDIF			
24200	C*					
24300	C		WRITEENDPAG			End Page
24400	C*					
24500	C	PAGNAM	IFEQ CPY,1			
24600	C		MOVE 'INVBAC 'OVRLAY	8		Back Overlay
24700	C		WRITEPRTOVL			
24800	C		ENDIF			
24900	C		WRITEENDPAG			

```

25000 C*
25100 CSR          ENDP          ENDSR
25200 C*
25300 C* Print invoice detail lines from DS
25400 C*
25500 CSR          PRDDET        BEGSR
25600 C            PAGCNT        IFEQ 1
25700 C            Z-ADD3.75      DWN    53      Offset for grid
25800 C            ELSE
25900 C            Z-ADD2.42      DWN          Offset for grid
26000 C            ENDIF
26100 C*
26200 C            Z-ADD1           X      20
26300 C            1             DO 48      X
26400 C            X             OCUR DETDS
26500 C            ITEM#         IFGT 0
26600 C            WRITEDETLIN
26700 C            WRITEDTGRID
26800 C            ADD .167       DWN
26900 C            ENDIF
27000 C            END
27100 CSR          ENDSR
27200 C*
27300 C* Blank out DS
27400 C*
27500 CSR          BLKDET        BEGSR
27600 C            1             DO 48      X
27700 C            X             OCUR DETDS
27800 C            MOVEL*BLANKS     UOM
27900 C            MOVEL*BLANKS     ITMDES
28000 C            Z-ADD0          QTY
28100 C            Z-ADD0          ITEM#
28200 C            Z-ADD0          SELPRC
28300 C            Z-ADD0          EXTPRC
28400 C            END
28500 CSR          ENDSR
28600 C* Set up Date
28700 CSR          DATESR        BEGSR
28800 C            Z-ADDUDAY        PAYDA
28900 C            UMONTH          ADD 1      PAYMO
29000 C            PAYMO          IFGT 12
29100 C            Z-ADD1          PAYMO
29200 C            UYEAR          ADD 1      PAYYR
29300 C            ELSE
29400 C            Z-ADDUYEAR        PAYYR
29500 C            ENDIF
29600 C            MOVE PAYDAA        PAYDAT 60
29700 C            Z-ADDPAYDAT      PAYDA0 60
29800 CSR          ENDSR
29900 * Set up Offer at end of invoice
30000 CSR          OFFSR        BEGSR
30100 C            SLSSSED         IFGE 500
30200 C            Z-ADD1          IX      20
30300 C            MOVE 'FLWRNB     'PSEG  8      Seed
30400 C            SETON          30
30500 C            GOTO WRTOFR
30600 C            ENDIF
30700 C            SLSFRT         IFGE 500
30800 C            Z-ADD7          IX          Strawberry
30900 C            MOVE 'STRWNB     'PSEG
31000 C            SETON          30
31100 C            GOTO WRTOFR
31200 C            ENDIF
31300 C            SLSSSUP         IFGE 500
31400 C            Z-ADD13         IX
31500 C            MOVE 'TREENB     'PSEG          Tree
31600 C            SETON          30

```

```

31700      C                GOTO WRTOFR
31800      C                ENDIF
31900      C          SLSCHM  IFGE 500
32000      C                Z-ADD19      IX
32100      C                MOVE 'BETLNB 'PSEG      Beetle
32200      C                SETON              30
32300      C                GOTO WRTOFR
32400      C                ENDIF
32500      C*
32600      C                GOTO ENDOFR
32700      C*
32800      C          WRTOFR  TAG
32900      C                MOVE WDS,IX      OFFR@1 24      Build Offer Text
33000      C                ADD 1          IX
33100      C                MOVE WDS,IX      OFFR@2 24
33200      C                ADD 1          IX
33300      C                MOVE WDS,IX      OFFR@3 24
33400      C                ADD 1          IX
33500      C                MOVE WDS,IX      OFFR@4 24
33600      C                ADD 1          IX
33700      C                MOVE WDS,IX      OFFR@5 24
33800      C                ADD 1          IX
33900      C                MOVE WDS,IX      OFFR@6 24
34000      CSR          ENDOFR  ENDSR
34100      C*
34200      *
34300      ** WDS      WORDS FOR OFFER
34400      Thank You .....
34500      Because you have ordered
34600      over $500 of seeds this
34700      year, on your next seed
34800      order you will receive
34900      a 10% discount.
35000      Thank You .....
35100      Because you have ordered
35200      over $500 of fruit this
35300      year, on your next fruit
35400      order you will receive
35500      a 10% discount.
35600      Thank You .....
35700      Because you have ordered
35800      over $500 of trees this
35900      year, on your next tree
36000      order you will receive
36100      a 10% discount.
36200      Thank You .....
36300      Because you have ordered
36400      over $500 of stuff this
36500      year, on your next stuff
36600      order you will receive
36700      a 10% discount.
36800
36900      ** CPY      Copy Name
37000      Customer Copy
37100      Packing List
37200      File Copy

```

* * * * * E N D O F S O U R C E * * * * *

INVNEW3 DDS Source

In the INVNEW3 DDS source, several program variables (&DWN, &DWN2, and &DWN3) are used to control spacing down the page for the DTGRID, CLOSE, and TOTAL record formats.

The INVNEW3 DDS printer file source is shown below:

```

5722WDS V5R2M0                      SEU SOURCE LISTING                      11/11/03 17:47:37          1
SOURCE FILE . . . . . SAMPLER/QDDSSRC
MEMBER . . . . . INVNEW3
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
100   A* INVNEW3 - Printer File DDS for Super Sun Seeds Invoice
200   A*           Example 3 (floating grid)
300   A*
400   A* Printer File Create or Change Parameters:
500   A* - Printer Device Type (DEVTYPE)      *AFPDS
600   A* - Overflow Line (OVRFLW)            64
700   A* - Print on Both Sides (DUPLEX)      *YES
800   A*           if running on duplex printer
900   A*
1000  A* Page 1 Header
1100  A*- includes Postnet Zip+4
1200  A*
1300  A           R INVTOP                      SKIPB(10)
1400  A           ZIPPN                      9S      12 BARCODE(POSTNET)
1500  A                                     SPACEA(2)
1600  A           NAME                      25A     12
1700  A           STNAME                    25A     48
1800  A                                     SPACEA(1)
1900  A           STREET                    25A     12
2000  A           STSTRT                    25A     48
2100  A                                     SPACEA(1)
2200  A           CITY                      25A     12
2300  A           STCITY                    25A     48
2400  A                                     SPACEA(1)
2500  A           STATE                     2A      12
2600  A           ZIP                       9S      16 EDTWRD(' - ')
2700  A           STSTE                     2A      48
2800  A           STZIP                     9S      52 EDTWRD(' - ')
2900  A                                     SPACEA(3)
3000  A           CUST#                     6S 0    14 EDTCDE(Z)
3100  A           INVC#                     6S 0    32 EDTCDE(Z)
3200  A                                     49DATE EDTCDE(Y)
3300  A           PAYDAT                    6S 0    66EDTCDE(Y)
3400  A                                     SPACEA(2)
3500  A           SHPVIA                    10A     14
3600  A                                     34DATE EDTCDE(Y)
3700  A           TERMS                     10A     47
3800  A           SLSMAN                    16A     64
3900  A                                     SPACEA(4)
4000  A*
4100  A* Page 2 Header
4200  A*
4300  A           R INVTP2                      SKIPB(10)
4400  A           NAME                      25A     12
4500  A                                     SPACEA(2)
4600  A           CUST#                     6S 0    14 EDTCDE(Z)
4700  A           INVC#                     6S 0    32 EDTCDE(Z)
4800  A                                     49DATE EDTCDE(Y)
4900  A           PAYDAT                    6S 0    66EDTCDE(Y)
5000  A                                     SPACEA(4)
5100  A*
5200  A* Detail Lines
5300  A*
5400  A           R DETLIN                      SPACEA(1)
5500  A           QTY                       4S 0    8 EDTCDE(Z)
5600  A           UOM                       2A      13
5700  A           ITEM#                     8S 0    18
5800  A           ITMDES                    25A     28
5900  A N51     SELPRC                     6S 2    58 EDTCDE(J)
6000  A N51     EXTPRC                     7S 2    70 EDTCDE(J)
6100  A*
6200  A* Floating Grid
6300  A*

```

```

6400      A          R DTGRID
6500      A          LINE(&DWN .55 0.167 *VRT *NARROW)
6600      A          LINE(&DWN 1.15 0.167 *VRT *NARROW)
6700      A          LINE(&DWN 1.55 0.167 *VRT *NARROW)
6800      A          LINE(&DWN 2.55 0.167 *VRT *NARROW)
6900      A          LINE(&DWN 5.65 0.167 *VRT *NARROW)
7000      A          LINE(&DWN 6.55 0.167 *VRT *NARROW)
7100      A          LINE(&DWN 7.94 0.167 *VRT *NARROW)
7200      A          DWN          5S 3P
7300      A*
7400      A* Close Grid
7500      A*
7600      A          R CLOSE
7700      A          LINE(&DWN .55 7.40 *HRZ *NARROW)
7800      A          DWN          5S 3P
7900      A*
8000      A* Total Box
8100      A*
8200      A          R TOTAL
8300      A          LINE(&DWN 6.55 0.5 *VRT *NARROW )
8400      A          LINE(&DWN 7.93 0.5 *VRT *NARROW )
8500      A          LINE(&DWN2 6.55 1.4 *HRZ *NARROW )
8600      A          TOTCON          9A          POSITION(&DWN3 5.1)
8700      A          FNTCHRSET(C0H200A0 T1V10037)
8800      A N51          TOTDUE          9S 2          EDTWRD(' , , $0. -')
8900      A          POSITION(&DWN3 6.5)
9000      A          DWN          5S 3P
9100      A          DWN2          5S 3P
9200      A          DWN3          5S 3P
9300      A*
9400      A* Multipage Message
9500      A*
9600      A          R PAGEOF
9700      A          PAGCON          4A          POSITION(10.7 7.3)
9800      A          FNTCHRSET(C0H200A0 T1V10037)
9900      A          PAGCNT          2S 0          POSITION(10.7 7.8)
10000     A          FNTCHRSET(C0H200A0 T1V10037)
10100     A          EDTCDE(Z)
10200     A          PAGNAM          25A          POSITION(10.7 3.8)
10300     A          FNTCHRSET(C0H400B0 T1V10037)
10400     A*
10500     A* Invoice Totals
10600     A* - includes Interleaf 2 of 5 barcode
10700     A*
10800     A          R INVBOT          SKIPB(55)
10900     A          PAYDA@          6S 0          25 EDTCDE(Y)
11000     A N51          TOTD@2          9S 2          67 EDTWRD(' , , $0. -')
11100     A          SPACEA(2)
11200     A 51          67 ' '
11300     A          SPACEA(2)
11400     A          NAME@2          25A          12
11500     A          SPACEA(1)
11600     A          STRE@2          25A          12
11700     A          BARPRC          15S 0          52BARCODE(INTERL20F5 3)
11800     A          SPACEA(1)
11900     A          CITY@2          25A          12
12000     A          SPACEA(1)
12100     A          STAT@2          2A          12
12200     A          ZIP@2          9A          16
12300     A*
12400     A* Offer Print
12500     A* - Font 92 is Courier Italic 12-pitch
12600     A*
12700     A          R OFFER          SKIPB(43)
12800     A          FONT(92)
12900     A          OFFR@1          24A          36
13000     A          SPACEA(1)

```



```

13100      A          OFFR@2      24A      36
13200      A          OFFR@3      24A      36      SPACEA(1)
13300      A          OFFR@3      24A      36
13400      A          OFFR@4      24A      36      SPACEA(1)
13500      A          OFFR@4      24A      36
13600      A          OFFR@5      24A      36      SPACEA(1)
13700      A          OFFR@5      24A      36
13800      A          OFFR@6      24A      36      SPACEA(1)
13900      A          OFFR@6      24A      36
14000      A          OFFR@6      24A      36      SPACEA(1)
14100      A*
14200      A* Images/Page Segments
14300      A* - variable page segment name from program
14400      A*
14500      A          R PAGSEG          PAGSEG(&PSEG 7.0 2.6)
14600      A          PSEG              8A P
14700      A*
14800      A*
14900      A* Images/Page Segments
15000      A* - variable overlay name from program
15100      A*
15200      A          R PRTOVL          OVERLAY(&OVLAY 0 0)
15300      A          OVLAY              8A P
15400      A*
15500      A          R ENDPAG          ENDPAGE
                                     * * * * E N D   O F   S O U R C E   * * * *

```


Appendix C. Source Code and Line Data for Super Sun Seeds

This appendix contains this information for the PPFA example described in Chapter 16, "Using Page Printer Formatting Aid," on page 183.

- "Page Definition Source Code"
- "Form Definition Source Code" on page 323
- "Sample Line Data" on page 328

Page Definition Source Code

This is the PPFA source code needed to generate the page definition for the line data shown in "Sample Line Data" on page 328. The output from using this page definition with the below form definition is shown in Chapter 16, "Using Page Printer Formatting Aid," on page 183.

```
5722WDS V5R2M0          SEU SOURCE LISTING          11/11/03 17:47:37          1
SOURCE FILE . . . . . AFPLIB/PPFASRC
MEMBER . . . . . SEEDS
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ..
100 /* * * * * * W H A T   T H I S   I S   * * * * * */
200 /* The following statements, written as PPFA source statements, */
300 /* describe how to process output from a specific application. The */
400 /* application-generated output was used along with a specification */
500 /* of the desired print format to create the following pagedef and */
600 /* formdef. PPFA is required to process these source statements */
700 /* and from them make a pagedef and a formdef object which PSF */
800 /* uses when printing. */
900 /* * * * * * */
1000
1100 /* * * * * * */
1200 /* pagedef Name: SEEDS */
1300 /* Conditional Processing inside this pagedef: */
1400 /* This is a somewhat complicated pagedef since there are actually */
1500 /* three Pageformats within the pagedef. The following describes */
1600 /* how the page formats work together to allow dynamical format */
1700 /* changing based upon the content of the application data: */
1800 /* 10F1 - First and only page of a single-page billing. */
1900 /* This Pageformat is given control initially under the */
2000 /* assumption that the customer's billing is 1 page long */
2100 /* (logic within this Pageformat will switch to the following */
2200 /* 10FMULTI Pageformat if the customer's billing data turns out */
2300 /* to span multiple pages). */
2400 /* 10FMULTI - First & interior pages of a multi-page billing. */
2500 /* This Pageformat is given control from the 10F1 Pageformat */
2600 /* and will continue processing pages until the last/summary page*/
2700 /* is reached (at which point logic within this Pageformat will */
2800 /* switch to the following NOFMULTI Pageformat). */
2900 /* NOFMULTI - Last/summary page of a multi-page billing. */
3000 /* One page will be processed, and control will be returned */
3100 /* unconditionally to Pageformat 10F1 for the next billing. */
3200 /* */
3300 /* Flow      Start-> 10F1 ->-+--> 10FMULTI ->-+--> NOFMULTI ->-+ */
3400 /* Summary:      A      |      A      V      |      */
3500 /*                |      V      +---<---+      V      */
3600 /*                +-----<-+-----<-----+      */
3700 /* * * * * * */
3800
3900 /* * * *   G E N E R A L   S E C T I O N   * * * * * */
```

```

4000 /* pagedef Name: SEEDS */
4100 /* Here we specify characteristics of the pagedef */
4200 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
4300 PAGEDEF SEEDS /* PPFA will produce P1SEEDS as output */
4400 REPLACE YES; /* PPFA should replace existing copy */
4500 /* Define font nicknames as follows: */
4600 FONT BOLD CB10; /* Courier Roman Bold 10 Char/Inch */
4700 FONT ITAL CI12; /* Courier Italic Normal 12 Char/inch */
4800 FONT NORM CR10; /* Courier Roman Normal 10 Char/inch */
4900 SETUNITS 1 IN 1 IN /* This indicates that the default unit */
5000 /* for all measurements is the inch */
5100 LINESP 6 LPI; /* Lines spacing at 6 lines/inch */
5200
5300 /* * * * P A G E F O R M A T 1 0 F 1 * * * * * * * * * * */
5400 /* 10F1 - This Pageformat lays out billing data to correspond */
5500 /* around the 0110F1 Overlay. This overlay has a customer */
5600 /* account section at the top of the page, followed by room */
5700 /* for up to 26 lines of billing details and 6 lines of a */
5800 /* "Thank You" message. At the bottom of the page is a pay- */
5900 /* ment slip to contain the customer address and "Amount Due". */
6000 /* Processing Logic Note: */
6100 /* Pageformat 10F1 gets control at the beginning since it is */
6200 /* the first Pageformat listed in the pagedef. It also */
6300 /* receives control from Pageformat NOFMULTI if the previous */
6400 /* billing spanned multiple pages. It is always true, then, */
6500 /* that the current data is on a first billing page. During */
6600 /* processing, it will be determined whether this is a single */
6700 /* page bill (in which case control stays in this Pageformat) */
6800 /* or a multiple-page billing (in which case control is passed */
6900 /* to 10FMULTI to reprocess). */
7000 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
7100 PAGEFORMAT 10F1; /* Name */
7200
7300 /* Customer Address Block - 4 input lines, 4 output lines */
7400 PRINTLINE
7500 CHANNEL 1 /* When an input line has a first col. */
7600 /* (CC) '1' value, this PRINTLINE gets */
7700 /* control - meaning: top of a new page */
7800 POSITION MARGIN 1.8 /* Horiz Pos:Margin; Vert Pos:Down 1.8" */
7900 FONT BOLD /* Customer name in bold print */
8000 REPEAT 4; /* Process 4 input lines this way */
8100
8200 /* Customer account data - 1 input line, 1 output line */
8300 PRINTLINE
8400 POSITION MARGIN 2.9 /* Horiz Pos:Margin; Vert Pos:Down 2.9" */
8500 FONT NORM; /* Normal font to be used here */
8600
8700 /* Shipping information - 1 input line, 1 output line */
8800 PRINTLINE
8900 POSITION MARGIN 3.2 /* Horiz Pos:Margin; Vert Pos:Down:3.2" */
9000 FONT NORM; /* Normal font to be used here */
9100
9200 /* Billing Detail Section - 20 lines of input, 20 lines of output*/
9300 /* Process the first 20 lines of billing details */
9400 PRINTLINE
9500 POSITION MARGIN 3.8 /* Horiz Pos:Margin; Vert Pos: Down 3.8"*/
9600 FONT NORM /* Normal font to be used here */
9700 REPEAT 20; /* Same processing for 20 lines */
9800 FIELD START 1 LENGTH 57; /* This field always printed */
9900 FIELD START 58 LENGTH 22 /* Note: Positioning is assumed by */
10000 /* default to follow previous FIELD */

```

```

10100                                     /* - which in this case is desirable. */
10200             SUPPRESSION PRICES ; /* Copygroup will decide if to print. */
10300
10400     /* These next 6 lines are either 6 more billing lines or 6 lines */
10500     /* of "Thank You" text. - 6 lines of input, 6 lines of output */
10600     /* Process the first 2 lines as if they are Thank You (Italic) */
10700     PRINTLINE
10800             POSITION MARGIN NEXT /* Horiz Pos:Margin; Vert Pos:continue */
10900             FONT ITAL          /* Italic font to be used here */
11000             REPEAT 2;          /* Same processing for 2 lines */
11100
11200     /* We must check line 3 to ensure it really is in the Thank You */
11300     /* area and, therefore, a first page of a single-page billing. */
11400     /* Furthermore, the field we check will indicate which "icon" */
11500     /* (the little picture of a seed, tree, or fruit) this page */
11600     /* includes. The inclusion is controlled here by specifying the */
11700     /* correct Copygroup. Note: The default icon is the seed icon; */
11800     /* if it is determined to be correct, no switch is necessary. */
11900     /* If the field has a non-icon value, we are in the middle of */
12000     /* a valid first page of a multiple-page billing and we must */
12100     /* immediately switch to the 10FMULTI Pageformat, which will */
12200     /* reprocess the page correctly. */
12300     PRINTLINE
12400             POSITION MARGIN NEXT /* Horiz Pos:Margin; Vert Pos: continue */
12500             FONT ITAL;          /* Italic font to be used here */
12600
12700             CONDITION GOT010FM /* Check if really first of single-page */
12800                                     /* Note: comparisons are case sensitive*/
12900             START 49 LENGTH 5 /* look for... */
13000             WHEN EQ 'seeds' /* Yes, 1 of 1 - although seeds is the */
13100                                     /* default icon, we need to check for */
13200                                     /* it or OTHERWISE would get control! */
13300             WHEN EQ 'trees' /* Yes, 1 of 1 - bring in trees picture */
13400             BEFORE SUBPAGE COPYGROUP 10F1TREE /* same Pageformat */
13500             WHEN EQ 'fruit' /* Yes, 1 of 1 - bring in fruit picture */
13600             BEFORE SUBPAGE COPYGROUP 10F1FRUI /* same PageFormat */
13700             OTHERWISE /* No, 1 of Multi - switch to 10FMULTI */
13800             BEFORE SUBPAGE COPYGROUP 10FMULTI PAGEFORMAT 10FMULTI;
13900
14000     /* If above conditions did not switch processing to NOFMULTI */
14100     /* (Switching Copygroup does not affect flow of control), then */
14200     /* this page really is the first of a single-page billing, so */
14300     /* continue processing through the end of the page. */
14400     PRINTLINE
14500             POSITION MARGIN NEXT /* Horiz Pos:Margin; Vert Pos: continue */
14600             FONT ITAL          /* Normal font to be used here */
14700             REPEAT 3;          /* 3 remaining "Thank you" lines */
14800
14900     /* "Total Due" line - 1 line of input, 1 line of output */
15000     PRINTLINE
15100             POSITION MARGIN 8.4 /* Horiz Pos:Margin; Vert Pos:Down 8.5" */
15200             FONT NORM;        /* Normal font to be used here */
15300             FIELD START 1 LENGTH 79
15400             SUPPRESSION PRICES ; /* Copygroup will decide if to print. */
15500
15600     /* Date/Total Line - 1 line of input, 1 line of output */
15700     PRINTLINE
15800             POSITION MARGIN 9.17 /* Horiz Pos:Margin; Vert Pos:Down 8.5" */
15900             FONT NORM;          /* Normal font to be used here */
16000             FIELD START 1 LENGTH 60; /* Always print the date */
16100             FIELD START 61 LENGTH 19 /* Total may be suppressed */

```

```

16200          SUPPRESSION PRICES ; /* Copygroup will decide if to print.  */
16300
16400          /* Footer Billing info - 5 lines of input, 5 lines of output  */
16500          PRINTLINE
16600          POSITION MARGIN 9.50 /* Horiz Pos:Margin; Vert Pos:Down 9.33"*/
16700          FONT NORM          /* Normal font to be used here      */
16800          REPEAT 5;          /* five lines input file lines output */
16900
17000          /* * * *   P A G E F O R M A T   1 0 F M U L T Y * * * * * */
17100          /* 10FMULTI - This Pageformat lays out billing data to      */
17200          /* correspond around the 0110FM Overlay. This overlay has a  */
17300          /* customer account section at the top of the page, followed */
17400          /* by room for up to 26 lines of billing details.           */
17500          /* Processing Logic Note:                                     */
17600          /* Pageformat 10F1 has already determined the current data to */
17700          /* be the first of a multiple-page billing. Now, this        */
17800          /* Pageformat will be used to process data until it determines */
17900          /* the last/summary page has been reached, at which time it  */
18000          /* gives control to Pageformat NOFMULTI.                      */
18100          /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
18200          PAGEFORMAT 10FMULTI; /* Name                               */
18300
18400          /* Customer Address Block - 4 input lines, 4 output lines    */
18500          PRINTLINE
18600          CHANNEL 1          /* When an input line has a first col. */
18700                          /* (CC) '1' value, this PRINTLINE gets */
18800                          /* control - meaning: top of a new page */
18900          POSITION MARGIN 1.8 /* Horiz Pos:Margin; Vert Pos:Down 1.8" */
19000          FONT BOLD         /* Customer name in bold print          */
19100          REPEAT 4;         /* Process 4 input lines this way      */
19200
19300          /* Customer account data - 1 input line, 1 output line      */
19400          PRINTLINE
19500          POSITION MARGIN 2.9 /* Horiz Pos:Margin; Vert Pos:Down 2.9" */
19600          FONT NORM;        /* Normal font to be used here          */
19700
19800          /* Shipping information - 1 input line, 1 output line        */
19900          PRINTLINE
20000          POSITION MARGIN 3.2 /* Horiz Pos:Margin; Vert Pos:Down:3.2" */
20100          FONT NORM;        /* Normal font to be used here          */
20200
20300          /* For the details section, change to 4 LPI to "fill out" area. */
20400          SETUNITS LINESP 4 LPI; /* 4 lines per inch                    */
20500
20600          /* Billing Detail Section - 26 lines of input, 26 lines of output*/
20700          /* Process the first 22 lines of billing details                */
20800          PRINTLINE
20900          POSITION MARGIN 3.8 /* Horiz Pos:Margin; Vert Pos: Down 3.8"*/
21000          FONT NORM          /* Normal font to be used here          */
21100          REPEAT 22;         /* Same processing for 22 lines         */
21200          FIELD START 1 LENGTH 57; /* This field always printed           */
21300          FIELD START 58 LENGTH 22 /* Note: Positioning is assumed by     */
21400                          /* default to follow previous FIELD    */
21500                          /* - which in this case is desirable.  */
21600          SUPPRESSION PRICES ; /* Copygroup will decide if to print.  */
21700
21800          /* We must check billing detail line 23 for the possibility that */
21900          /* this page is last/summary page. If it is, we switch to the  */
22000          /* NOFMULTI Pageformat, which will reprocess the page correctly. */
22100          PRINTLINE
22200          POSITION MARGIN NEXT /* Horiz Pos:Margin; Vert Pos: continue */

```

```

22300      FONT NORM;           /* Normal font to be used here      */
22400      FIELD START 1 LENGTH 57; /* This field always printed      */
22500      FIELD START 58 LENGTH 22 /* Note: Positioning is assumed by */
22600                                     /* default to follow previous FIELD */
22700                                     /* - which in this case is desirable. */
22800      SUPPRESSION PRICES ; /* Copygroup will decide if to print. */
22900
23000      CONDITION GOTONOF      /* Check if need to process last/summary*/
23100                                     /* Note: comparisons are case sensitive*/
23200      START 49 LENGTH 5      /* look for...                      */
23300      WHEN EQ 'seeds'        /* Yes, N of Multi - get seed picture */
23400      BEFORE SUBPAGE COPYGROUP NOFMSEED PAGEFORMAT NOFMULTI
23500      WHEN EQ 'trees'        /* Yes, N of Multi - get tree picture */
23600      BEFORE SUBPAGE COPYGROUP NOFMTREE PAGEFORMAT NOFMULTI
23700      WHEN EQ 'fruit'        /* Yes, N of Multi - get fruit picture*/
23800      BEFORE SUBPAGE COPYGROUP NOFMFRUI PAGEFORMAT NOFMULTI;
23900
24000      /* If above conditions did not switch processing to NOFMULTI, */
24100      /* then this page is still an internal page, so continue on.   */
24200      PRINTLINE
24300      POSITION MARGIN NEXT     /* Horiz Pos:Margin; Vert Pos: continue */
24400      FONT NORM               /* Normal font to be used here          */
24500      REPEAT 3;               /* Same processing for lines 24-26      */
24600      FIELD START 1 LENGTH 57; /* This field always printed           */
24700      FIELD START 58 LENGTH 22 /* Note: Positioning is assumed by     */
24800                                     /* default to follow previous FIELD     */
24900                                     /* - which in this case is desirable.   */
25000      SUPPRESSION PRICES ; /* Copygroup will decide if to print.  */
25100
25200      /* "CONTINUED" line      - 1 input line, 1 output line          */
25300      PRINTLINE
25400      POSITION MARGIN 10.5     /* Horiz Pos:Margin; Vert Pos:Down 10.5"*/
25500      FONT ITAL;             /* Italic font to be used here          */
25600
25700      /* "Page x" line         - 1 input line, 1 output line          */
25800      PRINTLINE
25900      POSITION MARGIN NEXT     /* Horiz Pos:Margin; Vert Pos: continue */
26000      FONT NORM;             /* Normal font to be used here          */
26100
26200      /* * * * P A G E F O R M A T   N O F M U L T I Y * * * * * * * */
26300      /* NOFMULTI - This Pageformat lays out billing data to          */
26400      /* correspond around the 01NOFM Overlay. This overlay has      */
26500      /* one line for a customer name at the top, and a large         */
26600      /* payment slip at the bottom to contain customer address and   */
26700      /* "Amount Due". In the middle of the page there is room for    */
26800      /* up to 26 lines of billing details and 6 lines for a "Thank  */
26900      /* You" message.                                                 */
27000      /* Processing Logic Note:                                       */
27100      /* Pageformat 10FMULTI has already determined the current data  */
27200      /* to be the last of a multiple-page billing. Now, this        */
27300      /* Pageformat will process this last page and return control   */
27400      /* to Pageformat 10F1 for the next customer's billing.         */
27500      /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
27600      PAGEFORMAT NOFMULTI;    /* Name                                 */
27700
27800      SETUNITS LINESP 6 LPI; /* 6 lines per inch                    */
27900
28000      /* Customer Address Block - 4 input lines, 1/2 output line     */
28100      /* We want to output part of the 1st line, and suppress the    */
28200      /* remaining 3 input lines (historically the application needed  */
28300      /* to output the information, but it no longer should be printed */

```

```

28400 /* (Note: this feature shows how we are protected from changing */
28500 /* (the application) */
28600 PRINTLINE
28700     CHANNEL 1 /* When an input line has a first col. */
28800 /* (CC) '1' value, this PRINTLINE gets */
28900 /* control - meaning: top of a new page */
29000     POSITION MARGIN 1.6; /* Horiz Pos:Margin; Vert Pos: Down 1.6"*/
29100     FIELD START 1 LENGTH 44/* Just put out "Sold to" Customer Name */
29200     FONT BOLD; /* Customer name in bold print */
29300 /* Before we forget, make a note that after finishing this page */
29400 /* we switch control back to Pageformat 10F1. This condition */
29500 /* has no connection with the above PRINTLINE, it could be */
29600 /* attached to any PRINTLINE in the Pageformat. */
29700     CONDITION GOTO10F1
29800     START 1 LENGTH 1
29900     WHEN GE X'00' /* Unconditional switch when finished */
30000 /* Note: seeds icon is made default for */
30100 /* the next customer billing here. */
30200     AFTER SUBPAGE COPYGROUP 10F1SEED PAGEFORMAT 10F1;
30300
30400
30500 /* Extra address lines - 3 input lines, 0 output lines (suppress)*/
30600 PRINTLINE
30700     POSITION MARGIN SAME /* Positioning doesn't matter */
30800     REPEAT 3; /* For three lines */
30900     FIELD TEXT ' '; /* Just put out blanks instead */
31000
31100 /* Customer account data - 1 input line, 1 output line */
31200 PRINTLINE
31300     POSITION MARGIN 1.9 /* Horiz Pos:Margin; Vert Pos:Down 1.9" */
31400     FONT NORM; /* Normal font to be used here */
31500
31600 /* Shipping information - 1 input line, 0 output lines (suppress)*/
31700 PRINTLINE
31800     POSITION MARGIN SAME; /* Positioning doesn't matter */
31900     FIELD TEXT ' '; /* Just put out blanks instead */
32000
32100 /* Billing Detail Section - 20 lines of input, 20 lines of output*/
32200 /* for the packing list copy. */
32300 PRINTLINE
32400     POSITION MARGIN 2.67 /* Horiz Pos:Margin; Vert Pos: Down 2.67*/
32500     FONT NORM /* Normal font to be used here */
32600     REPEAT 20; /* Same processing for 20 lines */
32700     FIELD START 1 LENGTH 57; /* This field always printed */
32800     FIELD START 58 LENGTH 22 /* Note: Positioning is assumed by */
32900 /* default to follow previous FIELD */
33000 /* - which in this case is desirable. */
33100     SUPPRESSION PRICES ; /* Copygroup will decide if to print. */
33200
33300 /* Thank You Section - 6 lines of input, 6 lines of output*/
33400 PRINTLINE
33500     POSITION MARGIN NEXT /* Horiz Pos:Margin; Vert Pos: continue */
33600     FONT ITAL /* Italic font to be used here */
33700     REPEAT 6; /* Same processing for 6 lines */
33800
33900 /* "Total Due" line - 1 line of input, 1 line of output */
34000 PRINTLINE
34100     POSITION MARGIN 8.4 /* Horiz Pos:Margin; Vert Pos:Down 8.4" */
34200     FONT NORM; /* Normal font to be used here */
34300     FIELD START 1 LENGTH 79
34400     SUPPRESSION PRICES ; /* Copygroup will decide if to print. */

```



```

34500
34600      /* Date/Total Line - 1 line of input, 1 line of output      */
34700      PRINTLINE
34800      POSITION MARGIN 9.17      /* Horiz Pos:Margin; Vert Pos:Down 8.5" */
34900      FONT NORM;              /* Normal font to be used here      */
35000      FIELD START 1 LENGTH 60; /* Always print the date            */
35100      FIELD START 61 LENGTH 19 /* Total may be suppressed          */
35200      SUPPRESSION PRICES ; /* Copygroup will decide if to print. */
35300
35400      /* Footer Billing info - 5 lines of input, 5 lines of output  */
35500      PRINTLINE
35600      POSITION MARGIN 9.50      /* Horiz Pos:Margin; Vert Pos:Down 9.33"*/
35700      FONT NORM              /* Normal font to be used here      */
35800      REPEAT 5;              /* five lines input five lines output */

```

Form Definition Source Code

This is the PPFA source code needed to generate the form definition for the line data shown in "Sample Line Data" on page 328. The output from using this form definition with the above page definition is shown in Chapter 16, "Using Page Printer Formatting Aid," on page 183.

```

35900 /* * * *   G E N E R A L   S E C T I O N   * * * * * * * * * * */
36000 /*formdef Name: SEEDS                                     */
36100 /* Here we specify characteristics of the formdef       */
36200 /*                                                       */
36300 /* Conditional Logic within the pagedef will switch us between the */
36400 /* different Copygroups. It is necessary to switch Copygroups      */
36500 /* to allow the application data to control if we use a first page  */
36600 /* overlays or a last/summary page overlay. Also, application data */
36700 /* determines which "icon" (seed, tree, or fruit) is to be included */
36800 /* on the summary sheet.                                           */
36900 /* Note: formdefs play no part in the switching logic - they are   */
37000 /* subject to the logical decisions made within the pagedefs.     */
37100 /*                                                       */
37200 /* Multiple output copies and Suppressible field design notes:    */
37300 /* Another requirement of the output is to produce three pages of  */
37400 /* output for each input page, one page for a packing list, one    */
37500 /* page for the customer, and one page for a "File" copy. Moreover, */
37600 /* the prices should be suppressed from the packing list printout,  */
37700 /* and each copy should be uniquely designated "Packing", "Customer" */
37800 /* or "File". Producing multiple copies is the responsibility of   */
37900 /* the formdef alone. For the suppression to work correctly, both  */
38000 /* the pagedef and formdef play a part; the pagedef identifies which */
38100 /* fields are suppressible, and the formdef turns on Suppression as */
38200 /* is necessary.                                                 */
38300 /*                                                       */
38400 /* There need to be seven Copygroups in this formdef (so many are  */
38500 /* needed since there are multiple combinations of icons to be    */
38600 /* merged onto summary-type sheets). There are 3 Copygroups for the */
38700 /* 3 10F1 scenarios (10F1SEED, 10F1TREE, and 10F1FRUI), 1 Copygroup */
38800 /* for the 10FMULTI, and 3 Copygroups for the 3 10FMULTI scenarios  */
38900 /* (10FMSEED, 10FMTREE, and 10FMFRUI).                          */
39000 /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
39100 FORMDEF SEEDS              /* PPFA will produce F1SEEDS as output */
39200 REPLACE YES              /* PPFA should replace existing copy  */
39300 DUPLEX NORMAL;          /* Print on both sides of the paper   */
39400
39500
39600 SUPPRESSION PRICES;      /* Prices will be suppressed on packing */
39700                          /* lists below.                        */

```

```

39800
39900 /* * * * C O P Y G R O U P 1 0 F 1 S E E D * * * * * * * * */
40000 /* 10F1SEED - This Copygroup provides the overlay for the */
40100 /* first page of a single-page billing. Also, it merges in */
40200 /* the seed icon at a fixed location so that the picture */
40300 /* appears next to the "Thank You" message. */
40400 /* * * * * * * * * * * * * * * * * * * * * * * * * * * */
40500 COPYGROUP 10F1SEED /* Name */
40600 CONSTANT BACK;
40700 /* No variable data on back page */
40800 /* You need to "register" overlays to be used in the copygroup */
40900 OVERLAY 10F1; /* 0110F1 - static design; lines, text */
41000 OVERLAY PACK; /* 01PACK - overlay for "PACK" copy */
41100 OVERLAY FILE; /* 01FILE - overlay for "FILE" copy */
41200 OVERLAY CUST; /* 01CUST - overlay for "Customer" copy*/
41300 OVERLAY TERMS; /* 01TERMS - Terms and Conditions ovly */
41400 OVERLAY SEED; /* 01SEED - Seed icon/picture */
41500
41600 /* For each page of application data, 3 duplex pages must be */
41700 /* printed: Front side Back side */
41800 /* Page 1 Packing list w/out prices Terms & Conditions */
41900 /* Page 2 Customer Copy with all data Terms & Conditions */
42000 /* Page 3 File Copy with all data Terms & Conditions */
42100 /* ----- Packing List Copy ----- */
42200 SUBGROUP FRONT COPIES 1/* Page 1 FRONT */
42300 SUPPRESSION PRICES /* no prices */
42400 OVERLAY 10F1 PACK /* packlist */
42500 SEED; /* seed icon */
42600 SUBGROUP BACK COPIES 1/* BACK */
42700 OVERLAY TERMS; /* Terms and Conditions */
42800 /* ----- Customer Copy ----- */
42900 SUBGROUP FRONT COPIES 1/* Page 2 FRONT */
43000 OVERLAY 10F1 CUST /* w/prices customer */
43100 SEED; /* seed icon */
43200 SUBGROUP BACK COPIES 1/* BACK */
43300 OVERLAY TERMS; /* Terms and Conditions */
43400 /* ----- "File" Copy ----- */
43500 SUBGROUP FRONT COPIES 1/* Page 3 FRONT */
43600 OVERLAY 10F1 FILE /* w/prices file copy */
43700 SEED; /* seed icon */
43800 SUBGROUP BACK COPIES 1/* BACK */
43900 OVERLAY TERMS; /* Terms and Conditions */
44000
44100 /* * * * C O P Y G R O U P 1 0 F 1 T R E E * * * * * * * * */
44200 /* 10F1TREE - This Copygroup provides the overlay for the */
44300 /* first page of a single-page billing. Also, it merges in */
44400 /* the tree icon at a fixed location so that the picture */
44500 /* appears next to the "Thank You" message. */
44600 /* * * * * * * * * * * * * * * * * * * * * * * * * * * */
44700 COPYGROUP 10F1TREE /* Name */
44800 CONSTANT BACK;
44900
45000 /* You need to "register" overlays to be used in the copygroup */
45100 OVERLAY 10F1; /* 0110F1 - static design; lines, text */
45200 OVERLAY PACK; /* 01PACK - overlay for "PACK" copy */
45300 OVERLAY FILE; /* 01FILE - overlay for "FILE" copy */
45400 OVERLAY CUST; /* 01CUST - overlay for "Customer" copy*/
45500 OVERLAY TERMS; /* 01TERMS - Terms and Conditions ovly */
45600 OVERLAY TREE; /* 01TREE - Tree icon/picture */
45700
45800 /* For each page of application data, 3 duplex pages must be */

```

```

45900      /* printed:      Front side          Back side          */
46000      /* Page 1      Packing list w/out prices  Terms & Conditions */
46100      /* Page 2      Customer Copy with all data  Terms & Conditions */
46200      /* Page 3      File Copy with all data      Terms & Conditions */
46300              /* ----- Packing List Copy ----- */
46400      SUBGROUP FRONT COPIES 1/* Page 1 FRONT          */
46500          SUPPRESSION PRICES /*          no prices          */
46600          OVERLAY 10F1 PACK /*          packlist          */
46700              TREE; /*          tree icon          */
46800      SUBGROUP BACK COPIES 1/*          BACK          */
46900          OVERLAY TERMS; /*          Terms and Conditions */
47000              /* ----- Customer Copy ----- */
47100      SUBGROUP FRONT COPIES 1/* Page 2 FRONT          */
47200          OVERLAY 10F1 CUST /*          w/prices customer */
47300              TREE; /*          tree icon          */
47400      SUBGROUP BACK COPIES 1/*          BACK          */
47500          OVERLAY TERMS; /*          Terms and Conditions */
47600              /* ----- "File" Copy ----- */
47700      SUBGROUP FRONT COPIES 1/* Page 3 FRONT          */
47800          OVERLAY 10F1 FILE /*          w/prices file copy */
47900              TREE; /*          tree icon          */
48000      SUBGROUP BACK COPIES 1/*          BACK          */
48100          OVERLAY TERMS; /*          Terms and Conditions */
48200
48300      /* * * *   C O P Y   G R O U P   1 0  F 1  F R U I   * * * * * */
48400      /* 10F1FRUI - This Copygroup provides the overlay for the */
48500      /* first page of a single-page billing. Also, it merges in */
48600      /* the fruit icon at a fixed location so that the picture */
48700      /* appears next to the "Thank You" message. */
48800      /* * * * * * * * * * * * * * * * * * * * * * * * * * * * */
48900      COPYGROUP 10F1FRUI /* Name          */
49000          CONSTANT BACK;
49100
49200      /* You need to "register" overlays to be used in the copygroup */
49300      OVERLAY 10F1; /* 0110F1 - static design; lines, text */
49400      OVERLAY PACK; /* 01PACK - overlay for "PACK" copy */
49500      OVERLAY FILE; /* 01FILE - overlay for "FILE" copy */
49600      OVERLAY CUST; /* 01CUST - overlay for "Customer" copy*/
49700      OVERLAY TERMS; /* 01TERMS - Terms and Conditions ovly */
49800      OVERLAY FRUI; /* 01FRUI - Fruit icon/picture */
49900
50000      /* For each page of application data, 3 duplex pages must be */
50100      /* printed:      Front side          Back side          */
50200      /* Page 1      Packing list w/out prices  Terms & Conditions */
50300      /* Page 2      Customer Copy with all data  Terms & Conditions */
50400      /* Page 3      File Copy with all data      Terms & Conditions */
50500              /* ----- Packing List Copy ----- */
50600      SUBGROUP FRONT COPIES 1/* Page 1 FRONT          */
50700          SUPPRESSION PRICES /*          no prices          */
50800          OVERLAY 10F1 PACK /*          packlist          */
50900              FRUI; /*          fruit icon          */
51000      SUBGROUP BACK COPIES 1/*          BACK          */
51100          OVERLAY TERMS; /*          Terms and Conditions */
51200              /* ----- Customer Copy ----- */
51300      SUBGROUP FRONT COPIES 1/* Page 2 FRONT          */
51400          OVERLAY 10F1 CUST /*          w/prices customer */
51500              FRUI; /*          fruit icon          */
51600      SUBGROUP BACK COPIES 1/*          BACK          */
51700          OVERLAY TERMS; /*          Terms and Conditions */
51800              /* ----- "File" Copy ----- */
51900      SUBGROUP FRONT COPIES 1/* Page 3 FRONT          */

```

```

52000          OVERLAY 10F1 FILE /*          w/prices file copy */
52100                    FRUI; /*          fruit icon */
52200          SUBGROUP BACK COPIES 1/*          BACK */
52300          OVERLAY TERMS; /*          Terms and Conditions */
52400
52500          /* * * * C O P Y G R O U P 1 0 F M U L T I * * * * * */
52600          /* 10FMULTI - This Copygroup provides the overlay for the */
52700          /* first & internal pages in a multiple-page billing. */
52800          /* * * * * * * * * * * * * * * * * * * * * * * * * */
52900          COPYGROUP 10FMULTI /* Name */
53000          CONSTANT BACK;
53100
53200          OVERLAY 10FM; /* 0110FM - static design; lines, text */
53300          OVERLAY PACK; /* 01PACK - overlay for Packing List */
53400          OVERLAY FILE; /* 01FILE - overlay for "FILE" copy */
53500          OVERLAY CUST; /* 01CUST - overlay for "Customer" copy*/
53600          OVERLAY TERMS; /* 01TERMS - Terms and Conditions ovly */
53700
53800          /* For each page of application data, 3 duplex pages must be */
53900          /* printed: Front side Back side */
54000          /* Page 1 Packing list w/out prices Terms & Conditions */
54100          /* Page 2 Customer Copy with all data Terms & Conditions */
54200          /* Page 3 File Copy with all data Terms and Conditions */
54300          /* ----- Packing List Copy ----- */
54400          SUBGROUP FRONT COPIES 1/* Page 1 FRONT */
54500          SUPPRESSION PRICES /* no prices */
54600          OVERLAY 10FM PACK; /* packlist */
54700          SUBGROUP BACK COPIES 1/* BACK */
54800          OVERLAY TERMS; /* Terms and Conditions */
54900          /* ----- Customer Copy ----- */
55000          SUBGROUP FRONT COPIES 1/* Page 2 FRONT */
55100          OVERLAY 10FM CUST; /* w/prices customer */
55200          SUBGROUP BACK COPIES 1/* BACK */
55300          OVERLAY TERMS; /* Terms and Conditions */
55400          /* ----- "File" Copy ----- */
55500          SUBGROUP FRONT COPIES 1/* Page 3 FRONT */
55600          OVERLAY 10FM FILE; /* w/prices file copy */
55700          SUBGROUP BACK COPIES 1/* BACK */
55800          OVERLAY TERMS; /* Terms and Conditions */
55900
56000          /* * * * C O P Y G R O U P N O F M S E E D * * * * * */
56100          /* NOFMSEED - This Copygroup provides the overlay for the */
56200          /* last page of a multi-page billing. Also, it merges in */
56300          /* the seed icon at a fixed location so that the picture */
56400          /* appears next to the "Thank You" message. */
56500          /* * * * * * * * * * * * * * * * * * * * * * * * * */
56600          COPYGROUP NOFMSEED /* Name */
56700          CONSTANT BACK;
56800
56900          /* You need to "register" overlays to be used in the Copygroup */
57000          OVERLAY NOFM; /* 01NOFM - static design; lines, text */
57100          OVERLAY PACK; /* 01PACK - overlay for "PACK" copy */
57200          OVERLAY FILE; /* 01FILE - overlay for "FILE" copy */
57300          OVERLAY CUST; /* 01CUST - overlay for "Customer" copy*/
57400          OVERLAY TERMS; /* 01TERMS - Terms and Conditions ovly */
57500          OVERLAY SEED; /* 01SEED - Seed icon/picture */
57600
57700          /* For each page of application data, 3 duplex pages must be */
57800          /* printed: Front side Back side */
57900          /* Page 1 Packing list w/out prices Terms & Conditions */
58000          /* Page 2 Customer Copy with all data Terms & Conditions */

```

```

58100      /* Page 3      File Copy with all data      Terms & Conditions */
58200      /* ----- Packing List Copy ----- */
58300      SUBGROUP FRONT COPIES 1/* Page 1 FRONT */
58400      SUPPRESSION PRICES /* no prices */
58500      OVERLAY NOFM PACK /* packlist */
58600      SEED; /* seed icon */
58700      SUBGROUP BACK COPIES 1/* BACK */
58800      OVERLAY TERMS; /* Terms and Conditions */
58900      /* ----- Customer Copy ----- */
59000      SUBGROUP FRONT COPIES 1/* Page 2 FRONT */
59100      OVERLAY NOFM CUST /* w/prices customer */
59200      SEED; /* seed icon */
59300      SUBGROUP BACK COPIES 1/* BACK */
59400      OVERLAY TERMS; /* Terms and Conditions */
59500      /* ----- "File" Copy ----- */
59600      SUBGROUP FRONT COPIES 1/* Page 3 FRONT */
59700      OVERLAY NOFM FILE /* w/prices file copy */
59800      SEED; /* seed icon */
59900      SUBGROUP BACK COPIES 1/* BACK */
60000      OVERLAY TERMS; /* Terms and Conditions */
60100
60200      /* * * * C O P Y G R O U P   N O F M T R E E * * * * * * * */
60300      /* NOFMTREE - This Copygroup provides the overlay for the */
60400      /* last page of a multi-page billing. Also, it merges in */
60500      /* the tree icon at a fixed location so that the picture */
60600      /* appears next to the "Thank You" message. */
60700      /* * * * * * * * * * * * * * * * * * * * * * * * * * * * */
60800      COPYGROUP NOFMTREE /* Name */
60900      CONSTANT BACK;
61000
61100      /* You need to "register" overlays to be used in the Copygroup */
61200      OVERLAY NOFM; /* 01NOFM - static design; lines, text */
61300      OVERLAY PACK; /* 01PACK - overlay for "PACK" copy */
61400      OVERLAY FILE; /* 01FILE - overlay for "FILE" copy */
61500      OVERLAY CUST; /* 01CUST - overlay for "Customer" copy*/
61600      OVERLAY TERMS; /* 01TERMS - Terms and Conditions ovly */
61700      OVERLAY SEED; /* 01SEED - Seed icon/picture */
61800
61900      /* For each page of application data, 3 duplex pages must be */
62000      /* printed:      Front side      Back side */
62100      /* Page 1      Packing list w/out prices      Terms & Conditions */
62200      /* Page 2      Customer Copy with all data      Terms & Conditions */
62300      /* Page 3      File Copy with all data      Terms & Conditions */
62400      /* ----- Packing List Copy ----- */
62500      SUBGROUP FRONT COPIES 1/* Page 1 FRONT */
62600      SUPPRESSION PRICES /* no prices */
62700      OVERLAY NOFM PACK /* packlist */
62800      TREE; /* tree icon */
62900      SUBGROUP BACK COPIES 1/* BACK */
63000      OVERLAY TERMS; /* Terms and Conditions */
63100      /* ----- Customer Copy ----- */
63200      SUBGROUP FRONT COPIES 1/* Page 2 FRONT */
63300      OVERLAY NOFM CUST /* w/prices customer */
63400      TREE; /* tree icon */
63500      SUBGROUP BACK COPIES 1/* BACK */
63600      OVERLAY TERMS; /* Terms and Conditions */
63700      /* ----- "File" Copy ----- */
63800      SUBGROUP FRONT COPIES 1/* Page 3 FRONT */
63900      OVERLAY NOFM FILE /* w/prices file copy */
64000      TREE; /* tree icon */
64100      SUBGROUP BACK COPIES 1/* BACK */

```

```

64200          OVERLAY TERMS;      /*          Terms and Conditions  */
64300
64400          /* * * *   C O P Y G R O U P   N O F M F R U I * * * * * * * */
64500          /* NOFMFRUI - This Copygroup provides the overlay for the      */
64600          /* last page of a multi-page billing. Also, it merges in        */
64700          /* the fruit icon at a fixed location so that the picture      */
64800          /* appears next to the "Thank You" message.                      */
64900          /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
65000          COPYGROUP NOFMFRUI      /* Name          */
65100          CONSTANT BACK;
65200
65300          /* You need to "register" overlays to be used in the Copygroup */
65400          OVERLAY NOFM;            /* 01NOFM - static design; lines, text */
65500          OVERLAY PACK;           /* 01PACK - overlay for "PACK" copy   */
65600          OVERLAY FILE;          /* 01FILE - overlay for "FILE" copy   */
65700          OVERLAY CUST;          /* 01CUST - overlay for "Customer" copy*/
65800          OVERLAY TERMS;         /* 01TERMS - Terms and Conditions ovly */
65900          OVERLAY FRUI;          /* 01FRUI - Fruit icon/picture       */
66000
66100          /* For each page of application data, 3 duplex pages must be    */
66200          /* printed:      Front side          Back side          */
66300          /* Page 1      Packing list w/out prices  Terms & Conditions */
66400          /* Page 2      Customer Copy with all data  Terms & Conditions */
66500          /* Page 3      File Copy with all data     Terms & Conditions */
66600          /* ----- Packing List Copy ----- */
66700          SUBGROUP FRONT COPIES 1/* Page 1 FRONT          */
66800          SUPPRESSION PRICES /*          no prices          */
66900          OVERLAY NOFM PACK /*          packlist          */
67000          FRUI; /*          fruit icon          */
67100          SUBGROUP BACK COPIES 1/* BACK          */
67200          OVERLAY TERMS; /*          Terms and Conditions */
67300          /* ----- Customer Copy ----- */
67400          SUBGROUP FRONT COPIES 1/* Page 2 FRONT          */
67500          OVERLAY NOFM CUST /*          w/prices customer */
67600          FRUI; /*          fruit icon          */
67700          SUBGROUP BACK COPIES 1/* BACK          */
67800          OVERLAY TERMS; /*          Terms and Conditions */
67900          /* ----- "File" Copy ----- */
68000          SUBGROUP FRONT COPIES 1/* Page 3 FRONT          */
68100          OVERLAY NOFM FILE /*          w/prices file copy */
68200          FRUI; /*          fruit icon          */
68300          SUBGROUP BACK COPIES 1/* BACK          */
68400          OVERLAY TERMS; /*          Terms and Conditions */
          * * * *   E N D   O F   S O U R C E   * * * *

```

Sample Line Data

This is an example of line data produced by an application program.

```

1          TEST COMPANY          SAME
          TEST STREET
          TEST CITY
          TS 12345-6789
              100          31300          11/06/02          12/06/02
              BEST WAY          11/06/02          NET 30          TESTSALES
1 CT 00000300 HIGH ALTITUDE WATERMELON          1.01          1.01
1 PK 01100517 SPARTAN SEEDS          2.39          2.39
9 PK 04569870 NORTHERN LITE BLUE SPRUCE          858.32          7,724.88
12 BX 11005004 BUSH GREEN SEEDS          2.50          30.00
12 CT 11005011 LASSO RED SEEDS          892.23          10,706.76
26 PK 11005018 EARLY BANTAM SEEDS          .38          9.88
5 BX 11057893 AFRICAN DAISY, SEEDS          2.35          11.75
1 PK 15975365 HEAVY OAK          129.09          129.09

```

33 BX	32746510	HOPS BREWING LIGHT	1.20	39.60
33 BX	32746510	HOPS BREWING LIGHT	1.20	39.60
6 EA	46578913	SEED SURVEYING SITE	50.00	300.00
2 BX	56413213	POT POT	7.65	15.30
80 PK	65412384	SEED SCRUBBER	888.79	71,103.20
1 PK	84512023	OREGON SPRING TOMATO SEED	.97	.97
2 DZ	96325874	PINEAPPLE-ORANGE SEEDS	1.29	2.58
11 BX	98412006	BLACK BEAUTY ZUCCHINI	2.30	25.30
5 EA	98546320	FROZEN JUICE PROCESSOR	109.90	549.50

Thank You
Because you have ordered
over \$500 of seeds this
year, on your next seed
order you will receive
a 10% discount.

\$90,691.81
12/06/02 \$90,691.81

TEST COMPANY
TEST STREET
TEST CITY
TS 123456789

				Page	1
1	THE LAST LEAF		CAMBIUM LAYER LIMITED		
	340 DESPERADO COURT		2222 SAPLING CIRCLE		
	LONGVIEW		BARKERSVILLE		
	CA 12345-6789		BC 47365-7290		
	154	31354	11/06/02	12/06/02	
	TREE TRUCK	11/06/02	NET 15	MARY PINETREE	
9 EA	00231300	SEED ROASTER OVEN SET	199.99	1,799.91	
9 EA	00231300	SEED ROASTER OVEN SET	199.99	1,799.91	
4 PK	03698741	STRING GRAPEFRUIT	2.01	8.04	
4 PK	03698741	STRING GRAPEFRUIT	2.01	8.04	
300 EA	11000146	AZALIA, GIANT ROSE SEEDS	.55	165.00	
300 EA	11000146	AZALIA, GIANT ROSE SEEDS	.55	165.00	
6 CT	11005010	EARLY DWARF DANISH SEEDS	3.01	18.06	
24 DZ	11005013	MINCOR NANTES CARROT SEED	.87	20.88	
6 DZ	11005013	MINCOR NANTES CARROT SEED	.87	5.22	
24 PK	11005020	FRENCH PICKLING SEEDS	2.39	57.36	
12 CT	12382910	SUCCATASH SEEDS	.38	4.56	
12 CT	12382910	SUCCATASH SEEDS	.38	4.56	
55 CT	13145340	SOUR GRAPE SEEDS	.15	8.25	
55 CT	13145340	SOUR GRAPE SEEDS	.15	8.25	
14 BZ	32165478	BLACK EYED BANANA	3.01	42.14	
14 BZ	32165478	BLACK EYED BANANA	3.01	42.14	
600 DZ	44646510	PLUMP RED PLUMS	.49	294.00	
600 DZ	44646510	PLUMP RED PLUMS	.49	294.00	
40 DZ	45613712	CRANAPPLE BERRY SEEDS	1.28	51.20	
40 DZ	45613712	CRANAPPLE BERRY SEEDS	1.28	51.20	
4 PK	65412384	SEED SCRUBBER	888.79	3,555.16	
4 PK	65412384	SEED SCRUBBER	888.79	3,555.16	
1 PK	65412384	SEED SCRUBBER	888.79	888.79	
1 PK	65412384	SEED SCRUBBER	888.79	888.79	
12 DZ	84512130	SUB-ARTIC TOMATO SEEDS	3.84	46.08	
12 DZ	84512130	SUB-ARTIC TOMATO SEEDS	3.84	46.08	

CONTINUED

				Page	1
1	THE LAST LEAF		CAMBIUM LAYER LIMITED		
	340 DESPERADO COURT		2222 SAPLING CIRCLE		
	LONGVIEW		BARKERSVILLE		
	CA 12345-6789		BC 47365-7290		
	154	31354	11/06/02	12/06/02	
	TREE TRUCK	11/06/02	NET 15	MARY PINETREE	
951 PK	11005018	EARLY BANTAM SEEDS	.38	361.38	
46 PK	11005018	EARLY BANTAM SEEDS	.38	17.48	

45 PK	11005018	EARLY BANTAM SEEDS	.38	17.10
4 BX	11057893	AFRICAN DAISY, SEEDS	2.35	9.40
4 BX	11057893	AFRICAN DAISY, SEEDS	2.35	9.40
100 EA	31321655	SEMILLAS DEL SUS SOMBEROS	24.95	2,495.00
100 EA	31321655	SEMILLAS DEL SUS SOMBEROS	24.95	2,495.00
1000 BX	56413213	POT POT	7.65	7,650.00

Thank You
 Because you have ordered
 over \$500 of trees this
 year, on your next tree
 order you will receive
 a 10% discount.

12/06/02
 THE LAST LEAF
 340 DESPERADO COURT
 LONGVIEW
 CA 123456789

\$26,882.54
 \$26,882.54

Page 2

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.

All sample code is provided by IBM for illustrative purposes only. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

All programs contained herein are provided to you "AS IS" without any warranties of any kind. The implied warranties of non-infringement, merchantability and fitness for a particular purpose are expressly disclaimed.

Appendix D. Sample Toolbox Source Code

This appendix contains source code for a COBOL program and an RPG program, both using the AFP Toolbox.

Sample COBOL Program using AFP Toolbox

The sample COBOL program shown in this appendix creates the familiar “Super Sun Seeds” application (shown in Figure 198) as it could be implemented with AFP Toolbox function calls.

The image shows a sample invoice from Super Sun Seeds, a Growth Company. The invoice includes customer information, shipping details, a table of items, a personalized thank-you message, and a barcode. Callouts point to various AFP Toolbox features used in the layout:

- Overlay:** Points to the Super Sun Seeds logo and company name.
- Page Segment:** Points to the top header area.
- Various Alignments with Typographic Fonts:** Points to the customer address and shipping information.
- Variable Size Boxes:** Points to the item table.
- Personalized Paragraph:** Points to the thank-you message.
- Page Number Count:** Points to the 'Page 1 of 1' text.
- Bar Code:** Points to the barcode at the bottom right.

Qty	UOM	Item #	Item Description	Price	Extension
1000	BX	56413213	Pot Pot	7.65	7,650.00
45	BZ	11005015	Chantenay Seeds	2.19	98.55
900	EA	00001200	Arboles Del Sur	45.00	40,500.00
98	PK	84512023	Oregon Spring Tomato Seed	.37	95.06
4	BX	11057893	African Daisy, Seeds	2.35	9.40
951	CT	11005011	Lasso Red Seeds	892.23	48,510.73
46	DZ	11005014	Scarlet Nantes Seeds	5.90	271.40
100	EA	31321655	Semillas Del Sus Somberos	24.95	2,495.00

Figure 198. Super Sun Seeds Case Study Created with AFP Toolbox

Only portions of the sample program are shown here to illustrate some of the Toolbox functions. A description of the COBOL program listing follows the source code.

Refer to *IBM Advanced Function Presentation Toolbox for Multiple Operating Systems (S544-5292)* for instruction on how to execute the sample programs.

The entire source for this program can be found in library QAOCL/QAYTBSAMPS(CBLTEST) once the Toolbox product is installed. The copy files to declare the AFP variables and PERFORM statements are provided in QAOCL/QAYTBDECLS(AOCLCBLH) and QAOCL/QAYTBDECLS(QAOCLCLP), respectively.

The sample data is supplied in QAOCL/QAYTBSAMPS(SUNSEEDATA).

```

000100000000 IDENTIFICATION DIVISION.
000200000000 *****
000300000000 * COBOL PROGRAM -- AFP TOOLBOX SUPER SUN SEEDS *
000400000000 * *
000500000000 * This program invokes the AFP Toolbox to produce a sample *
000600000000 * customer billing statement. See the "AFP Toolbox *
000700000000 * User's Guide" for a picture of the printed output produced *
000800000000 * by this program, and for more information about the Toolbox *
000900000000 * function calls. This sample program is also provided in *
001000000000 * member CBLTEST of the QAOCL/QAYTBSAMPS file. *
001100000000 * *
001200000000 *****
001300000000 PROGRAM-ID. SUNSEEDS.
001400000000 ENVIRONMENT DIVISION.
001500000000
001600000000 CONFIGURATION SECTION.
001700000000 SOURCE-COMPUTER. IBM.
001800000000 OBJECT-COMPUTER. IBM.
001900000000
002000000000 INPUT-OUTPUT SECTION.
002100000000 FILE-CONTROL.
002200000000 SELECT INPUT-DATA
002300000000 ASSIGN TO DISK-SUNSEEDATA
002400000000 ORGANIZATION IS SEQUENTIAL
002500000000 FILE STATUS IS FILE-STATUS.
002600000000
002700000000 *****
002800000000 * *
002900000000 * DATA DIVISION *
003000000000 * *
003100000000 *****
003200000000 DATA DIVISION.
003300000000 FILE SECTION.
003400000000 FD INPUT-DATA
003500000000 RECORD CONTAINS 80 CHARACTERS.
003600000000 01 INPUT-RECORD.
003700000000 03 TRANS-QUANTITY PIC 9(4).
003800000000 03 TRANS-UOM PIC X(2).
003900000000 03 TRANS-ITEMNUM PIC X(8).
004000000000 03 TRANS-DESCRIPTION PIC X(25).
004100000000 03 TRANS-AMOUNT-IN PIC S9(7)V9(2).
004200000000 03 TRANS-CUST-NUM PIC 9(4).
004300000000 03 TRANS-INVOICE-NUM PIC 9(8).
004400000000 03 TRANS-INVOICE-DATE PIC 9(6).
004500000000 03 TRANS-PAYMENT-DATE PIC 9(6).
004600000000 03 TRANS-EXTRA PIC X(8).
004700000000
004800000000 WORKING-STORAGE SECTION.
004900000000
005000000000 01 FILE-STATUS PIC 99.
005100000000 01 CUST-IN.
005200000000 03 CUST-NAME PIC X(25) VALUE SPACES.
005300000000 03 CUST-ST-ADDR PIC X(30) VALUE SPACES.
005400000000 03 CUST-CITY-STATE PIC X(35) VALUE SPACES.
005500000000 03 ACCOUNT-NUM-IN PIC 9(4) VALUE ZERO.
005600000000 03 INVOICE-NUM-IN PIC 9(8) VALUE ZERO.
005700000000 03 INVOICE-DATE-IN PIC 9(6) VALUE ZERO.
005800000000 03 PAYMENT-DATE-IN PIC 9(6) VALUE ZERO.
005900000000 03 SHIP-DATE-IN PIC 9(6) VALUE ZERO.
006000000000 03 TERMS-IN PIC X(4) VALUE SPACES.
006100000000 03 SALESMAN-IN PIC X(25) VALUE SPACES.
006200000000
006300000000 01 TRANSACTION-PROCESSING-VARS.
006400000000 03 PAGE-HEADER-DEPTH PIC 9(5)V9(4) BINARY
006500000000 VALUE 0.0.
006600000000 03 PAGE-BODY PIC 9(5)V9(4) BINARY

```

```

006700000000 VALUE 0.0.
006800000000 03 BOTTOM-MARGIN PIC 9(5)V9(4) BINARY
006900000000 VALUE 20.0.
007000000000 03 NUM-CUSTOMER-PAGES PIC 99 BINARY VALUE 1.
007100000000
007200000000 01 CUST-OUT.
007300000000 03 ACCOUNT-NUM-OUT PIC 9999.
007400000000 03 INVOICE-NUM-OUT PIC 99999999.
007500000000 03 OLD-BALANCE-OUT PIC $$, $$, $9.99.
007600000000 03 MIN-AMOUNT-DUE-OUT PIC $$$, $9.99.
007700000000 03 SHIP-DATE-OUT PIC 99/99/99.
007800000000 03 INVOICE-DATE-OUT PIC 99/99/99.
007900000000 03 TRANSACTION-DATE-OUT PIC 99/99/99.
008000000000 03 TRANS-AMOUNT-OUT PIC $$, $9,999.99.
008100000000 03 EXTENSION-AMOUNT-OUT PIC $$, $99,999.99.
008200000000 03 NUM-CUSTOMER-PAGES-OUT PIC Z9.
008300000000
008400000000 01 PROCESSING-SWITCHES.
008500000000 03 DATA-REMAINS-SWITCH PIC X(3) VALUE "YES".
008600000000 03 NEW-CUSTOMER PIC X(3) VALUE "NO ".
008700000000 03 EXTENSION-AMOUNT-COMP PIC 9(7)V99.
008800000000 03 TOT-AMOUNT-DUE-COMP PIC 9(7)V99.
008900000000
009000000000 01 EXCLAMATION PIC X VALUE X"4f".
009100000000
009200000000 01 FONT-IDS.
009300000000 05 TIM10MED PIC 9(4) BINARY VALUE 0.
009400000000 05 TIM10BOLD PIC 9(4) BINARY VALUE 0.
009500000000 01 RULE-HANDLES.
009600000000 03 RULEHNDL1 USAGE IS POINTER.
009700000000 03 RULEHNDL2 USAGE IS POINTER.
009800000000 03 RULEHNDL3 USAGE IS POINTER.
009900000000 03 RULEHNDL4 USAGE IS POINTER.
010000000000 03 RULEHNDL5 USAGE IS POINTER.
010100000000 03 RULEHNDL6 USAGE IS POINTER.
010200000000 03 RULEHNDL7 USAGE IS POINTER.
010300000000 03 RULEHNDL8 USAGE IS POINTER.
010400000000 01 AFP-STRING-VARIABLES.
010500000000 03 AFP-STRING-1 PIC X(10)
010600000000 VALUE "Thank you ".
010700000000 03 AFP-STRING-2 PIC X(160)
010800000000 VALUE "Because almost half your order was Lasso Sun S
- "eeds, you will receive a 10% discount on your next ord
- "er.".
011000000000
011100000000 *****
011100000001 *
011200000000 * Include the provided copy file that contains *
011200000001 * the AFP Toolbox variables *
011200000002 * *
011200000003 *****
011300000000 COPY AOCLCBLH.
011400000000
011500000000 /-----*
011600000000 *****
011700000000 *
011800000000 * MAINLINE *
011900000000 *
012000000000 * This program produces a sample customer statement.*
012100000000 * The statement contains an overlay with the Super *
012200000000 * Sun Seeds logo and other fixed data. The *
012300000000 * customer transactions are formatted into columns *
012400000000 * of text that are right, left, center, and *
012500000000 * character-aligned to demonstrate the different *
012600000000 * Toolbox functions. Variable size boxes are also *
012700000000 * drawn to create the columns. *
012800000000 *

```

```

012900000000
013000000000
013100000000
013200000000
013300000000
013400000000
013500000000
013600000000
013700000000
013800000000
013900000000
014000000000
014100000000
014200000000
014300000000
014400000000
014500000000
014600000000
014700000000
014800000000
014900000000
015000000000
015100000000
015200000000
015300000000
015400000000
015500000000
015600000000
015700000000
015800000000
015900000000
016000000000
016100000000
016200000000
016300000000
016400000000
016500000000
016600000000
016700000000
016800000000
016900000000
017000000000
017100000000
017200000000
017400000000
017600000000
017700960322
017800000000
018000000000
018100000000
018200000000
018300000000
018400000000
018500000000
018600000000
018700000000
018900000000
019100000000
019200000000
019300000000
019400000000
019500000000
019600000000
019700000000
019800000000
019900000000
020000000000

*****
PROCEDURE DIVISION.

MAINLINE.

    OPEN INPUT INPUT-DATA.
    IF FILE-STATUS NOT = ZEROS
        DISPLAY "UNABLE TO OPEN INPUT FILE"
        DISPLAY "FILE STATUS" FILE-STATUS
        STOP RUN.

    PERFORM SETUP-AFPTBOX.
    PERFORM READ-DATA.
    PERFORM PROCESS-A-CUSTOMER UNTIL DATA-REMAINS-SWITCH
        = "NO ".
    PERFORM END-PROCESSING.

    CLOSE INPUT-DATA.
    DISPLAY "SUNSEEDS COMPLETED".
    STOP RUN.

/-----*
*-----*
*
*   SETUP-AFPTBOX.
*       Initialize the session.
*       Begin a document.
*       Define the fonts.
*
*-----*

SETUP-AFPTBOX.
    PERFORM AFPINIT.

*-----*
*
*   Begin the document and specify the output file.
*
*-----*

        MOVE "SUPER SUN SEEDS" TO AFP-DOC-NAME.
        MOVE "iSeries COBOL Program" TO AFP-DOC-COMMENT.
        MOVE FILED TO AFP-OUTPUT-TYPE.
        MOVE "/QSYS.LIB/QAOCL.LIB/QAYTBRESCS.FILE/SUNSEEDS.MBR"
            TO AFP-OUTPUT-FILENAME.
    PERFORM AFPBDOC.

*-----*
*
*   Define the fonts.
*
*-----*

        MOVE "T1V10500" TO AFP-CODE-PAGE.
        MOVE "TIMES NEW ROMAN LATIN1" TO AFP-DESCRIPTIVE-NAME.
        MOVE 100 TO AFP-POINT-SIZE.
        MOVE MEDIUM TO AFP-WEIGHT.
        MOVE NORMAL TO AFP-FONT-WIDTH.
        MOVE ROMAN TO AFP-STYLE.
    PERFORM AFPDFNTAT.
        MOVE AFP-FONT-ID TO TIM10MED.
        MOVE BOLD TO AFP-WEIGHT.
    PERFORM AFPDFNTAT.
        MOVE AFP-FONT-ID TO TIM10BOLD.

```

```

020100000000 *-----*
020200000000 *
020300000000 *   READ SAMPLE DATA   *
020400000000 *
020500000000 *-----*
020600000000 READ-DATA.
020700000000   READ INPUT-DATA AT END
020800000000     MOVE "NO" TO DATA-REMAINS-SWITCH
020900000000   END-READ.
021000000000   IF FILE-STATUS NOT = ZEROS AND FILE-STATUS NOT = 10
021100000000     DISPLAY "READ ERROR"
021200000000     DISPLAY "FILE STATUS" FILE-STATUS
021300000000     STOP RUN.
021400000000 *-----*
021500000000 * If this is a new customer, read the customer address and *
021600000000 * account balance. *
021700000000 *-----*
021800000000   IF TRANS-QUANTITY EQUAL ZEROS AND DATA-REMAINS-SWITCH
021900000000     NOT EQUAL "NO"
022000000000     MOVE "YES" TO NEW-CUSTOMER
022100000000     MOVE TRANS-CUST-NUM TO ACCOUNT-NUM-IN
022200000000     MOVE TRANS-INVOICE-NUM TO INVOICE-NUM-IN
022300000000     MOVE TRANS-INVOICE-DATE TO INVOICE-DATE-IN
022400000000     MOVE TRANS-PAYMENT-DATE TO PAYMENT-DATE-IN
022500000000
022600000000 *-----*
022700000000 * Read the customer name *
022800000000 *-----*
022900000000   READ INPUT-DATA AT END MOVE "NO" TO DATA-REMAINS-SWITCH
023000000000   END-READ
023100000000   MOVE TRANS-DESCRIPTION TO CUST-NAME
023200000000
023300000000 *-----*
023400000000 * Read the customer street address. *
023500000000 *-----*
023600000000   READ INPUT-DATA AT END MOVE "NO" TO DATA-REMAINS-SWITCH
023700000000   END-READ
023800000000   MOVE TRANS-DESCRIPTION TO CUST-ST-ADDR
023900000000
024000000000 *-----*
024100000000 * Read the customer city and state. *
024200000000 *-----*
024300000000   READ INPUT-DATA AT END MOVE "NO" TO DATA-REMAINS-SWITCH
024400000000   END-READ
024500000000   MOVE TRANS-DESCRIPTION TO CUST-CITY-STATE
024600000000
024700000000 *-----*
024800000000 * Read the first customer transaction *
024900000000 *-----*
025000000000   READ INPUT-DATA AT END MOVE "NO" TO DATA-REMAINS-SWITCH
025100000000   END-READ
025200000000   IF FILE-STATUS NOT = ZEROS
025300000000     DISPLAY "READ ERROR"
025400000000     DISPLAY "FILE STATUS" FILE-STATUS
025500000000     STOP RUN
025600000000   ELSE CONTINUE
025700000000   END-IF
025800000000   ELSE CONTINUE
025900000000   END-IF.
026000000000 /-----*
026100000000 *-----*
026200000000 *
026300000000 *   PROCESS THE CUSTOMER. *
026400000000 *
026500000000 *   Begin a page *
026600000000 *   Write the page header *
026700000000 *   Begin vertical rules *

```

```

026800000000 *          Process the customer transactions *
026900000000 *          End vertical rules and draw horizontal rule *
027000000000 *          Include 'thank you' notice if room *
027100000000 *          Write the page footer (tear-off strip) *
027200000000 *          End the page *
027300000000 *-----*
027400000000
027500000000 PROCESS-A-CUSTOMER.
027600000000 *-----*
027700000000 * Initialize the number of transactions for this customer to 0 *
027800000000 *-----*
027900000000     MOVE 1 TO NUM-CUSTOMER-PAGES.
028000000000     MOVE DEFAULT-WS TO AFP-EXTRA-VARS-SPACE.
028100000000     MOVE DEFAULT-ICS TO AFP-EXTRA-ICSPACE.
028200000000
028300000000     MOVE "Page 1" TO AFP-PAGE-NAME.
028500000000     PERFORM AFPBPAGE.
028600000000
028700000000     PERFORM CREATE-THE-HEADER.
028800000000
028900000000     PERFORM PROCESS-TRANSACTIONS.
029000000000
029100000000     PERFORM CREATE-THE-FOOTER.
029200000000
029300000000     PERFORM AFPEPAGE.
029400000000
029500000000 /-----*
029600000000 *-----*
029700000000 *
029800000000 *   CREATE THE HEADER. *
029900000000 * *
030000000000 *-----*
030100000000
030200000000     CREATE-THE-HEADER.
030300000000 *-----*
030400000000 * INCLUDE THE OVERLAY *
030500000000 *-----*
030600000000     MOVE 0 TO AFP-X-COORDINATE.
030700000000     MOVE ABSOLUTE-POS TO AFP-X-REF-COORD-SYS.
030800000000     MOVE 0 TO AFP-Y-COORDINATE.
030900000000     MOVE ABSOLUTE-POS TO AFP-Y-REF-COORD-SYS.
031000000000     PERFORM AFPSPPOS.
031100000000
031200000000     MOVE "INVHEAD1" TO AFP-OVLY-NAME.
031300000000     PERFORM AFPIOVL.
031400000000
031500000000     PERFORM PROCESS-THE-ADDRESS.
031600000000
031700000000 *-----*
031800000000 * Start vertical rules under the shaded box *
031900000000 *-----*
032000000000     MOVE 10 TO AFP-RULE-FAT-THICKNESS.
032100000000     MOVE BASELINE-DIR TO AFP-DIRECTION.
032200000000
032300000000     MOVE 793 TO AFP-X-COORDINATE.
032400000000     MOVE ABSOLUTE-POS TO AFP-X-REF-COORD-SYS.
032500000000     MOVE 5442 TO AFP-Y-COORDINATE.
032600000000     MOVE ABSOLUTE-POS TO AFP-Y-REF-COORD-SYS.
032700000000     PERFORM AFPSPPOS.
032800000000     PERFORM AFPBRULE.
032900000000     SET RULEHNDL1 TO AFP-RULE-HANDLE.
033000000000     MOVE 850 TO AFP-X-COORDINATE.
033100000000     PERFORM AFPHMOVE.
033200000000     PERFORM AFPBRULE.
033300000000     SET RULEHNDL2 TO AFP-RULE-HANDLE.
033400000000     MOVE 567 TO AFP-X-COORDINATE.
033500000000     PERFORM AFPHMOVE.

```

```

033600000000      PERFORM AFPBRULE.
033700000000      SET RULEHNDL3 TO AFP-RULE-HANDLE.
033800000000      MOVE 1474 TO AFP-X-COORDINATE.
033900000000      PERFORM AFPHMOVE.
034000000000      PERFORM AFPBRULE.
034100000000      SET RULEHNDL4 TO AFP-RULE-HANDLE.
034200000000      MOVE 4505 TO AFP-X-COORDINATE.
034300000000      PERFORM AFPHMOVE.
034400000000      PERFORM AFPBRULE.
034500000000      SET RULEHNDL5 TO AFP-RULE-HANDLE.
034600000000      MOVE 1304 TO AFP-X-COORDINATE.
034700000000      PERFORM AFPHMOVE.
034800000000      PERFORM AFPBRULE.
034900000000      SET RULEHNDL6 TO AFP-RULE-HANDLE.
035000000000      MOVE 1500 TO AFP-X-COORDINATE.
035100000000      PERFORM AFPHMOVE.
035200000000      PERFORM AFPBRULE.
035300000000      SET RULEHNDL7 TO AFP-RULE-HANDLE.
035400000000
035500000000      *-----*
035600000000      * Set font for transaction text                                *
035700000000      *-----*
035800000000      MOVE AFP-FONT-ID TO TIM10MED.
035900000000      PERFORM AFPSFONT.
036000000000
036100000000      *-----*
036200000000      * Move down for first transaction                                    *
036300000000      *-----*
036400000000      MOVE 0 TO AFP-X-COORDINATE.
036500000000      PERFORM AFPHMOVE.
036600000000      PERFORM AFPNLIN.
036700000000
036800000000      *-----*
036900000000      * Query the position and calculate the page header depth.          *
037000000000      *-----*
037100000000      PERFORM AFPQUERY.
037200000000      MOVE AFP-Y-COORDINATE TO PAGE-HEADER-DEPTH.
037300000000
037400000000      *-----*
037500000000      * Write the account number and invoice number                    *
037600000000      *-----*
037700000000      MOVE TIM10MED TO AFP-FONT-ID.
037800000000      PERFORM AFPSFONT.
037900000000
038000000000      MOVE 2254 TO AFP-X-COORDINATE.
038100000000      MOVE ABSOLUTE-POS TO AFP-X-REF-COORD-SYS.
038200000000      MOVE 4335 TO AFP-Y-COORDINATE.
038300000000      MOVE ABSOLUTE-POS TO AFP-Y-REF-COORD-SYS.
038400000000      PERFORM AFPSPOS.
038500000000
038600000000      MOVE ACCOUNT-NUM-IN TO ACCOUNT-NUM-OUT.
038700000000      MOVE 4 TO AFP-STRING-LENGTH.
038800000000      MOVE ACCOUNT-NUM-OUT TO AFP-CHARACTER-STRING.
039000000000      MOVE RIGHT-ALIGN TO AFP-ALIGNMENT-OPTION.
039100000000      MOVE FALS TO AFP-POSITION-OPTION.
039200000000      PERFORM AFPWRITE.
039300000000
039400000000      MOVE 3121 TO AFP-X-COORDINATE.
039500000000      PERFORM AFPHMOVE.
039600000000
039700000000      MOVE INVOICE-NUM-IN TO INVOICE-NUM-OUT.
039800000000      MOVE 8 TO AFP-STRING-LENGTH.
039900000000      MOVE INVOICE-NUM-OUT TO AFP-CHARACTER-STRING.
040100000000      MOVE RIGHT-ALIGN TO AFP-ALIGNMENT-OPTION.
040200000000      MOVE TRU TO AFP-POSITION-OPTION.
040300000000      PERFORM AFPWRITE.
040400000000

```

```

040500000000 *-----*
040600000000 * Write the invoice date and payment date *
040700000000 *-----*
040800000000
040900000000 MOVE 2551 TO AFP-X-COORDINATE.
041000000000 PERFORM AFPHMOVE.
041100000000
041200000000 MOVE INVOICE-DATE-IN TO INVOICE-DATE-OUT.
041300000000 MOVE INVOICE-DATE-OUT TO AFP-CHARACTER-STRING.
041400000000 MOVE 8 TO AFP-STRING-LENGTH.
041600000000 PERFORM AFPWRITE.
041700000000
041800000000 MOVE 2100 TO AFP-X-COORDINATE.
041900000000 PERFORM AFPHMOVE.
042000000000
042100000000 MOVE PAYMENT-DATE-IN TO SHIP-DATE-OUT.
042200000000 MOVE SHIP-DATE-OUT TO AFP-CHARACTER-STRING.
042300000000 MOVE 8 TO AFP-STRING-LENGTH.
042500000000 PERFORM AFPWRITE.
042600000000
042700000000 *-----*
042800000000 * Write the shipping information. *
042900000000 *-----*
043000000000
043100000000
043200000000 *-----*
043300000000 * Write the terms and sale info. *
043400000000 *-----*
043500000000
043600000000 PERFORM AFPNLINE.
043700000000 MOVE 2800 TO AFP-X-COORDINATE.
043800000000 PERFORM AFPHMOVE.
043900000000
044000000000 MOVE TERMS-IN TO AFP-CHARACTER-STRING.
044100000000 MOVE 3 TO AFP-STRING-LENGTH.
044300000000 MOVE LEFT-ALIGN TO AFP-ALIGNMENT-OPTION.
044400000000 PERFORM AFPWRITE.
044500000000
044600000000 MOVE 2800 TO AFP-X-COORDINATE.
044700000000 PERFORM AFPHMOVE.
044800000000
044900000000 CALL "TRIM" USING SALESMAN-IN,
045000000000 BY CONTENT LENGTH OF SALESMAN-IN,
045100000000 BY REFERENCE AFP-CHARACTER-STRING,
045200000000 AFP-STRING-LENGTH.
045300000000 ADD 1 TO AFP-STRING-LENGTH.
045600000000 MOVE LEFT-ALIGN TO AFP-ALIGNMENT-OPTION.
045700000000 PERFORM AFPWRITE.
045800000000
045900000000 /-----*
046000000000 *-----*
046100000000 * *
046200000000 * PROCESS THE ADDRESS. *
046300000000 * *
046400000000 *-----*
046500000000
046600000000 PROCESS-THE-ADDRESS.
046700000000 *-----*
046800000000 * Create the postal zip code. *
046900000000 *-----*
047000000000
047100000000 MOVE TIM10BOLD TO AFP-FONT-ID.
047200000000 PERFORM AFPSFONT.
047300000000
047400000000 MOVE 1450 TO AFP-X-COORDINATE.
047500000000 MOVE ABSOLUTE-POS TO AFP-X-REF-COORD-SYS.
047600000000 MOVE 2584 TO AFP-Y-COORDINATE.

```



```

047700000000          MOVE ABSOLUTE-POS TO AFP-Y-REF-COORD-SYS.
047800000000          PERFORM AFPSPDS.
047900000000
048000000000          *-----*
048100000000          * Write the customer name.                                *
048200000000          *-----*
048300000000
048400000000          PERFORM AFPNLIN.
048500000000          CALL "TRIM" USING CUST-NAME,
048600000000                      BY CONTENT LENGTH OF CUST-NAME,
048700000000                      BY REFERENCE AFP-CHARACTER-STRING,
048800000000                      AFP-STRING-LENGTH.
048900000000
049000000000          ADD 1 TO AFP-STRING-LENGTH.
049300000000          MOVE LEFT-ALIGN TO AFP-ALIGNMENT-OPTION.
049400000000          MOVE TRU TO AFP-POSITION-OPTION.
049500000000          PERFORM AFPWRITE.
049600000000
049700000000          *-----*
049800000000          * Write the customer address.                                *
049900000000          *-----*
050000000000          PERFORM AFPNLIN.
050100000000
050200000000          CALL "TRIM" USING CUST-ST-ADDR,
050300000000                      BY CONTENT LENGTH OF CUST-ST-ADDR,
050400000000                      BY REFERENCE AFP-CHARACTER-STRING,
050500000000                      AFP-STRING-LENGTH.
050600000000
050700000000          ADD 1 TO AFP-STRING-LENGTH.
051000000000          MOVE LEFT-ALIGN TO AFP-ALIGNMENT-OPTION.
051100000000          MOVE TRU TO AFP-POSITION-OPTION.
051200000000          PERFORM AFPWRITE.
051300000000
051400000000          PERFORM AFPNLIN.
051500000000
051600000000          CALL "TRIM" USING CUST-CITY-STATE,
051700000000                      BY CONTENT LENGTH OF CUST-CITY-STATE,
051800000000                      BY REFERENCE AFP-CHARACTER-STRING,
051900000000                      AFP-STRING-LENGTH.
052000000000
052100000000          ADD 1 TO AFP-STRING-LENGTH.
052300000000          MOVE LEFT-ALIGN TO AFP-ALIGNMENT-OPTION.
052400000000          MOVE TRU TO AFP-POSITION-OPTION.
052500000000          PERFORM AFPWRITE.
052600000000
052700000000          PERFORM AFPNLIN.
052800000000
052900000000          /-----*
053000000000          *-----*
053100000000          *
053200000000          *   PROCESS TRANSACTIONS.                                *
053300000000          *
053400000000          *-----*
053500000000
053600000000          PROCESS-TRANSACTIONS.
053700000000          MOVE "NO " TO NEW-CUSTOMER.
053800000000          MOVE 5442 TO AFP-Y-COORDINATE.
053900000000          PERFORM AFPVMOVETO.
054000000000          PERFORM WRITE-TRANSACTIONS UNTIL
054100000000          NEW-CUSTOMER = "YES" OR DATA-REMAINS-SWITCH = "NO".
054200000000          PERFORM AFPNLIN.
054300000000          MOVE 793 TO AFP-X-COORDINATE.
054400000000          MOVE ABSOLUTE-POS TO AFP-X-REF-COORD-SYS.
054500000000          PERFORM AFPHMOVETO.
054600000000          SET AFP-RULE-HANDLE TO RULEHNDL1.
054700000000          PERFORM AFPERULE.
054800000000          MOVE 850 TO AFP-X-COORDINATE.

```

```

054900000000    PERFORM AFPHMOVE.
055000000000    SET AFP-RULE-HANDLE TO RULEHNDL2.
055100000000    PERFORM AFPERULE.
055200000000    MOVE 570 TO AFP-X-COORDINATE.
055300000000    PERFORM AFPHMOVE.
055400000000    SET AFP-RULE-HANDLE TO RULEHNDL3.
055500000000    PERFORM AFPERULE.
055600000000    MOVE 1427 TO AFP-X-COORDINATE.
055700000000    PERFORM AFPHMOVE.
055800000000    SET AFP-RULE-HANDLE TO RULEHNDL4.
055900000000    PERFORM AFPERULE.
056000000000    MOVE 4422 TO AFP-X-COORDINATE.
056100000000    PERFORM AFPHMOVE.
056200000000    SET AFP-RULE-HANDLE TO RULEHNDL5.
056300000000    PERFORM AFPERULE.
056400000000    MOVE 1247 TO AFP-X-COORDINATE.
056500000000    PERFORM AFPHMOVE.
056600000000    SET AFP-RULE-HANDLE TO RULEHNDL6.
056700000000    PERFORM AFPERULE.
056800000000    MOVE 1984 TO AFP-X-COORDINATE.
056900000000    PERFORM AFPHMOVE.
057000000000    SET AFP-RULE-HANDLE TO RULEHNDL7.
057100000000    PERFORM AFPERULE.
057200000000    MOVE 850 TO AFP-X-COORDINATE.
057300000000    MOVE ABSOLUTE-POS TO AFP-X-REF-COORD-SYS.
057400000000    PERFORM AFPHMOVE.
057500000000    MOVE 10500 TO AFP-RULE-LENGTH.
057600000000    MOVE 10 TO AFP-RULE-THICKNESS.
057700000000    PERFORM AFPPHRUL.
057800000000
057900000000
058000000000    *-----*
058100000000    * If there is room, build a personalized paragraph.      *
058200000000    *-----*
058300000000    PERFORM AFPQUERY.
058400000000    IF AFP-Y-COORDINATE < 8000
058500000000    MOVE 4040 TO AFP-PARA-WIDTH
058600000000    MOVE JUSTIFY-ALIGN TO AFP-FORMAT-OPTION
058700000000    MOVE DEFAULT-LSP TO AFP-LINE-SPACING
058800000000    MOVE 10000 TO AFP-PARA-MAXDEPTH
058900000000    PERFORM AFPBPARA
059000000000
059100000000    MOVE LOW-VALUES TO AFP-CHARACTER-STRING
059200000000    STRING AFP-STRING-1
059300000000    DELIMITED BY SIZE INTO AFP-CHARACTER-STRING
059400000000    CALL "STRING-LENGTH" USING AFP-CHARACTER-STRING,
059500000000    BY CONTENT LENGTH OF AFP-CHARACTER-STRING,
059600000000    BY REFERENCE AFP-STRING-LENGTH
059700000000    ADD 1 TO AFP-STRING-LENGTH
060000000000    PERFORM AFPPTTEXT
060100000000
060200000000    CALL "TRIM" USING CUST-NAME,
060300000000    BY CONTENT LENGTH OF CUST-NAME,
060400000000    BY REFERENCE AFP-CHARACTER-STRING,
060500000000    AFP-STRING-LENGTH
060600000000
060700000000    ADD 1 TO AFP-STRING-LENGTH
061000000000    PERFORM AFPPTTEXT
061100000000
061200000000    MOVE LOW-VALUES TO AFP-CHARACTER-STRING
061300000000    STRING AFP-STRING-2
061400000000    DELIMITED BY SIZE INTO AFP-CHARACTER-STRING
061500000000    CALL "STRING-LENGTH" USING AFP-CHARACTER-STRING,
061600000000    BY CONTENT LENGTH OF AFP-CHARACTER-STRING,
061700000000    BY REFERENCE AFP-STRING-LENGTH
061800000000    ADD 1 TO AFP-STRING-LENGTH
062100000000

```

```

062200960322          PERFORM AFPPTXT
062300960322          PERFORM AFPEPARA
062400960322          END-IF.
062500000000
062600000000
062700000000          *-----*
062800000000          * Write the transaction rows for this customer.          *
062900000000          *-----*
063000000000          WRITE-TRANSACTIONS.
063100000000          MOVE 1000 TO AFP-X-COORDINATE.
063200000000          PERFORM AFPHMOVETO.
063300000000          PERFORM AFPNLIN.
063400000000          PERFORM AFPQUERY.
063500000000
063600000000          *-----*
063700000000          *           This is where we check to see if we need another          *
063800000000          *           page.  if so, count it, and start a new one.          *
063900000000          *           don't end page until all done so we can print          *
064000000000          *           'page n of m' string .          *
064100000000          *-----*
064200000000          IF AFP-RET-CODE = 13 OR AFP-Y-COORDINATE > 11520
064300000000          PERFORM END-CUST-PAGE.
064400000000
064500000000          MOVE TRANS-QUANTITY TO AFP-CHARACTER-STRING.
064700000000          MOVE 4 TO AFP-STRING-LENGTH.
064800000000          MOVE LEFT-ALIGN TO AFP-ALIGNMENT-OPTION.
064900000000          MOVE TRU TO AFP-POSITION-OPTION.
065000000000          PERFORM AFPWRITE.
065100000000          MOVE 1000 TO AFP-X-COORDINATE.
065200000000          PERFORM AFPHMOVE.
065300000000
065400000000          MOVE TRANS-UOM TO AFP-CHARACTER-STRING.
065600000000          MOVE 2 TO AFP-STRING-LENGTH.
065700000000          MOVE CENTER-ALIGN TO AFP-ALIGNMENT-OPTION.
065800000000          MOVE TRU TO AFP-POSITION-OPTION.
065900000000          PERFORM AFPWRITE.
066000000000
066100000000          MOVE 417 TO AFP-X-COORDINATE.
066200000000          PERFORM AFPHMOVE.
066300000000
066400000000          MOVE TRANS-ITEMNUM TO AFP-CHARACTER-STRING.
066600000000          MOVE 8 TO AFP-STRING-LENGTH.
066700000000          MOVE LEFT-ALIGN TO AFP-ALIGNMENT-OPTION.
066800000000          MOVE TRU TO AFP-POSITION-OPTION.
066900000000          PERFORM AFPWRITE.
067000000000
067100000000          MOVE 1417 TO AFP-X-COORDINATE.
067200000000          PERFORM AFPHMOVE.
067300000000          MOVE TRANS-DESCRIPTION TO AFP-CHARACTER-STRING.
067500000000          MOVE 25 TO AFP-STRING-LENGTH.
067600000000          MOVE LEFT-ALIGN TO AFP-ALIGNMENT-OPTION.
067700000000          MOVE TRU TO AFP-POSITION-OPTION.
067800000000          PERFORM AFPWRITE.
067900000000
068000000000          MOVE 8390 TO AFP-X-COORDINATE.
068100000000          PERFORM AFPHMOVETO.
068200000000          MOVE TRANS-AMOUNT-IN TO TRANS-AMOUNT-OUT.
068300000000          MOVE TRANS-AMOUNT-OUT TO AFP-CHARACTER-STRING.
068500000000          MOVE 7 TO AFP-STRING-LENGTH.
068600000000          MOVE RIGHT-ALIGN TO AFP-ALIGNMENT-OPTION.
068700000000          MOVE TRU TO AFP-POSITION-OPTION.
068800000000          PERFORM AFPWRITE.
068900000000
069000000000          MOVE 1500 TO AFP-X-COORDINATE.
069100000000          PERFORM AFPHMOVE.
069200000000          MULTIPLY TRANS-QUANTITY BY TRANS-AMOUNT-IN GIVING
069300000000          EXTENSION-AMOUNT-COMP ROUNDED.
          MOVE EXTENSION-AMOUNT-COMP TO EXTENSION-AMOUNT-OUT.

```

```

069400000000          MOVE EXTENSION-AMOUNT-OUT TO AFP-CHARACTER-STRING.
069500000000          MOVE 9 TO AFP-STRING-LENGTH.
069700000000          MOVE RIGHT-ALIGN TO AFP-ALIGNMENT-OPTION.
069800000000          MOVE TRU TO AFP-POSITION-OPTION.
069900000000          PERFORM AFPWRITE.
070000000000
070100000000          PERFORM READ-DATA.
070200000000
070300000000          *-----*
070400000000          *           This is for more than one page per customer.           *
070500000000          *-----*
070600000000          END-CUST-PAGE.
070700000000
070800000000          MOVE NUM-CUSTOMER-PAGES TO NUM-CUSTOMER-PAGES-OUT.
070900000000          MOVE NUM-CUSTOMER-PAGES-OUT TO AFP-PAGE-NAME.
071100000000          PERFORM AFPBPAGE.
071200000000
071300000000          PERFORM CREATE-THE-HEADER.
071400000000          ADD 1 TO NUM-CUSTOMER-PAGES.
071500000000
071600000000          *-----*
071700000000          *                                           *
071800000000          *           PROCESS THE FOOTER.           *
071900000000          *                                           *
072000000000          *-----*
072100000000
072200000000          CREATE-THE-FOOTER.
072300000000          *-----*
072400000000          * Position below the tear strip.           *
072500000000          *-----*
072600000000          MOVE 0 TO AFP-X-COORDINATE.
072700000000          MOVE ABSOLUTE-POS TO AFP-X-REF-COORD-SYS.
072800000000          MOVE 0 TO AFP-Y-COORDINATE.
072900000000          MOVE ABSOLUTE-POS TO AFP-Y-REF-COORD-SYS.
073000000000          PERFORM AFPSPPOS.
073100000000
073200000000          /-----*
073300000000          *-----*
073400000000          *                                           *
073500000000          *           TERMINATE THE AFPAPI AND THE PROGRAM           *
073600000000          *                                           *
073700000000          *-----*
073800000000
073900000000          END-PROCESSING.
074000000000          PERFORM AFPEDOC.
074100000000          PERFORM AFPEND.
074100000001          *****
074100000002          *                                           *
074200000000          * Include the provided copy files that contains           *
074200000001          * the AFP Toolbox subroutines           *
074200000002          *                                           *
074200000003          *****
074300000000          COPY AOCLCBLP.
074400960322          COPY APQTRIM.
074500960322          COPY APQSTRL.
074600000000
074700000000          END PROGRAM SUNSEEDS.

```

Description of COBOL Program Listing

There are several important differences in the way this application can be done with the Toolbox. First, typographic fonts can be used, because the Toolbox enables text strings to be formatted in several alignments. Second, the boxes around the item transactions can be drawn to the exact size needed instead of always being a fixed size. Third, a personalized paragraph of text is included for each customer order. Notice that the text in this paragraph is formatted and that the lines are flowed by

Toolbox, so that the application program does not have to measure the strings of characters. Bar codes can be generated directly from your application. Finally, you can number pages “n of m” and make decisions from your application about when a page of text is full.

All Toolbox applications begin by initializing the session (as shown in line 016400000000). If the initialization call is successful, the next thing to do is begin a document. Lines 017200000000 through 018000000000 show how to specify the output file name and other information about the document for the output AFP data stream.

After the document is started, each page must be built (see lines 028300000000 and 028500000000). The application reads the customer data from an input file. The name and address are formatted at the top of the page, following the overlay which contains the Super Sun Seeds logo and other fixed text (see lines 030300000000 through 035300000000).

The Toolbox provides the capability to format text as left-, right-, center- or character-aligned strings. Left-aligned strings such as the customer name and address are shown in lines 048400000000 through 052700000000 of the sample. Right-aligned strings for the account and invoice numbers are shown in lines 038000000000 through 040300000000. Centered strings are shown in the UOM column (lines 065300000000 through 065800000000), and character-alignment is used for the Total Due field.

One of the most powerful functions provided by the Toolbox is the ability to produce formatted paragraphs of text. The personalized paragraph in the middle of the page is produced by the code shown in lines 057900000000 through 062400960322.

The variable size rules for the box are started in lines 032000000000 through 035300000000 and ended in lines 054300000000 through 057700000000.

Sample RPG Program using AFP Toolbox

The following is a sample of the source code of an RPG program that was created with AFP Toolbox. This sample code creates a paragraph of formatted text in 14-point Times New Roman font.

```

HDATAFMT(*USA)
*
FQSYSPRT 0 F 132 PRINTER OFLIND(*INOV)
*
D* These arrays hold constants used in the program
DWDS S 100A DIM(3) PERRCD(1) CTDATA
DDTA S 45A DIM(12) PERRCD(1) CTDATA
* Font ID's
D TIM14NAME S 5I 0
D TIM14TYPE S 5I 0
* Return code
D AFPRNCDE S 10I 0
*
D/COPY AOCLRPGH
*
* * * * *
*
* MAINLINE
*
* The mainline logic is as follows:
*
* Set-up the API

```

```

*           Begin a Page
*           Set Position
*           Write a string of text in 14 point bold TNR
*           End the Page
*           End the API
*

```

```

* * * * *

```

```

C           EXSR      SETUPAPIR
*
C           DTA(1)    CAT(P)  x'00':0      AFPPAGNAM
* The above line places the DTA(1) data in AFPPAGNAM and adds a Hex'00' as
C           EXSR      AFPBPAGER
*
C           Z-ADD     1440      AFPXCRD
C           Z-ADD     1440      AFPYCRD
C           Z-ADD     ABSPOS     AFPXRFCRDS
C           Z-ADD     ABSPOS     AFPYRFCRDS
C           EXSR      AFPSPOSR
*
C           Z-ADD     TIM14TYPE  AFPFNTID
C           EXSR      AFPSFONTR
*
C           DTA(2)    CAT(P)  X'00':0      AFPCHRSTR
C           MOVE      LEFTALGN  AFPALNOPT
C           MOVE      FALS      AFPPOSOFT
C           Z-ADD     -1         AFPXTRVAR
C           Z-ADD     -1         AFPXTRICS
C           MOVE      '.'       AFPALNCHR
C           EXSR      AFPWRITER
*
C           Z-ADD     1440      AFPYCRD
C           EXSR      AFPVMOVER
*
C           Z-ADD     20        AFPRULTHK
C           Z-ADD     4320     AFPBOXWID
C           EXSR      AFPBBOXR
*
C           EXSR      AFPNLINER
*
C           Z-ADD     140       AFPXCRD
C           EXSR      AFPHMOVER
*
C           Z-ADD     4040     AFPPARWTH
C           MOVE      JUSTFYALN AFPFMTOPT
C           MOVE      DEFLSP   AFPLINSPC
C           Z-ADD     10000    AFPPARMAX
C           EXSR      AFPBPARAR
*
C           WDS(1)    CAT(P)  WDS(2):0     AFPCHRSTR
C           CAT(P)    WDS(3):0     AFPCHRSTR
C           CAT(P)    X'00':0     AFPCHRSTR
C           EXSR      AFPPTXTR
*
C           EXSR      AFPEPARAR
*
C           Z-ADD     80        AFPYCRD
C           EXSR      AFPVMOVER
*
C           Z-ADD     1440     AFPXCRD
C           EXSR      AFPHMOVTOR
*
C           EXSR      AFPEBOXR
*
C           EXSR      AFPEPAGER
*
C           EXSR      AFPEDOCCR

```

```

C          EXSR      APPENDR
*
C      FINIS      TAG
C          SETON
LR

```

```

* * * * *
*          Subroutines for AFP Toolbox
* * * * *

```

```

C/COPY AOCLRPGS

```

```

* * * * *
*
* Set Up the API
*   Initialize the AFP API
*   Set the output characteristics
*   Begin a document
*   Define the fonts
*

```

```

C      SETUPAPIR  BEGSR
C          EXSR      AFPINITR

```

```

* Set the output characteristics
C      DTA(3)      CAT(P)  x'00':0      AFPDOCNAM
C      DTA(4)      CAT(P)  x'00':0      AFPDOCCMT
C          Z-ADD      FILED      AFPOUTTYP
C      DTA(5)      CAT(P)  x'00':0      AFPOUTFLE
C          EXSR      AFPBDOCR

```

```

* Define the fonts
C      DTA(6)      CAT(P)  x'00':0      AFPDESNAM
C      DTA(7)      CAT(P)  x'00':0      AFPCDEPAG
C          Z-ADD      140      AFPPNTSIZ
C          Z-ADD      MEDIUM  AFPWGT
C          Z-ADD      NORMAL  AFPFNTWTH
C          Z-ADD      ROMAN   AFPSTL
C          EXSR      AFPDFNTATR

```

```

*
C          Z-ADD      AFPFNTID  TIM14TYPE
C*      DTA(8)      CAT(P)  x'00':0      AFPCHRSET
C*          EXSR      AFPDFNTNMR
C*          Z-ADD      AFPFNTID  TIM14NAME

```

```

C          ENDSR

```

```

*
OQSYSVRT  E          ERROUT      1  5
O          AFPERRDTA      27
O          'FAILED'
O          'Toolbox Return Code'
O          AFPRTNODE      52

```

```

**CTDATA WDS

```

The Toolbox lets you produce customized output pages that utilize typographic fonts, shading, graphics, and image. You can focus your customer's attention on important information that might be overlooked.

```

**CTDATA DTA
Page 1 of Many      1
HELLO WORLD          2
RPG TEST DOC         3
And a comment        4
/QSYS.LIB/MYLIB.LIB/AFPOUT.FILE/RPGTEST.MBR  5
TIMES NEW ROMAN LATIN1  6
TIV10500             7
CON20000             8

```

Appendix E. Migrating from OfficeVision/400

OfficeVision/400 (OV/400) uses AFP function to significantly enhance the quality of documents it can produce. OV/400 can incorporate images, graphic data files, and overlays into its documents. Combined with OV/400's capabilities to merge application data into documents, this provides some very powerful application possibilities.

For example, company logos, company letterhead, and business forms can all be merged into a document which is, in turn, merged with records from a database file. You can use OV/400 to create sales letters or dunning letters. You can create "form-intensive" applications, such as insurance or mortgage applications, that combine text, overlays, and application data into customer documents.

OV/400 is no longer supported and it will not run on any release of OS/400 beyond V4R5.

Today, there is no single product that offers all the services available in OfficeVision/400. However, there is often more than one possible replacement for each OfficeVision/400 function.

Lotus® Domino™

An applications and messaging server with an integrated set of services that let you create secure, interactive business solutions for the Internet and corporate intranets. Lotus Domino is also a client/server platform for developing and deploying groupware applications. IBM has migration tools to move your users from OV/400 to Lotus Domino.

DTM/400

Offers native OS/400 word processing and data/text merge functions, based on XML architecture. Existing OV/400 documents can be reused and existing OS/400 applications continue to run without any modification. Integration into Lotus Domino environments includes a graphical editor and integration with the powerful Domino groupware and workflow functions. DTM/400 is an IBM business partner product.

EverGreen/400

A 5250 e-mail and calendaring client that allows users of traditional OS/400 terminals (or PC's with emulation sessions) to be integrated into e-mail and calendar environments. EverGreen/400 protects your investment in 5250 hardware and twinax cabling. EverGreen/400 is an IBM business partner product.

For more information about DTM/400 and EverGreen/400, refer to the Inventive Designers Web site: www.inventivedesigners.com/.

The Redbook *How to Replace OfficeVision/400 in Your Applications: Looking at Domino for AS/400 and AS/400 Alternatives* contains information about migrating from OV/400. It describes the above products in greater detail and has instructions about when and how to migrate to these products. It also provides several sample applications, including OfficeVision/400 sample applications, as a starting point. It is intended for use by those AS/400 customers, Business Partners, and software vendors who have developed applications that use the functions included in, or associated with, OfficeVision/400. It is also suitable for those who wish to develop similar new applications without previous experience with OfficeVision/400.

Appendix F. Performing Advanced Font Tasks with iSeries Navigator

There are several tasks you can perform on fonts with iSeries Navigator that require advanced knowledge of fonts. Changing values for the font mapping tables and specifying font capture should only be done by a qualified system administrator. You can perform these advanced tasks with the AFP Manager component of iSeries Navigator. For information about installing AFP Manager, see “Installing AFP Manager” on page 268.

- “Viewing and Changing Font Mapping Tables”
- “Creating a Font Mapping Table” on page 350
- “Activating Font Capture” on page 351

Viewing and Changing Font Mapping Tables

You can view system font mapping tables or view and change user-defined font mapping tables with AFP Manager by following these steps:

1. Double-click the iSeries Access icon.
2. Double-click the iSeries Navigator icon.
3. Open a connection to an OS/400.
4. Expand **AFP Manager** then **Font Mapping Tables**.
5. To view the system tables, open **System Tables**. To view a user-defined table, open **User Tables**.
6. Double-click the table you want to view. Figure 199 shows the user-defined HPFCS table.

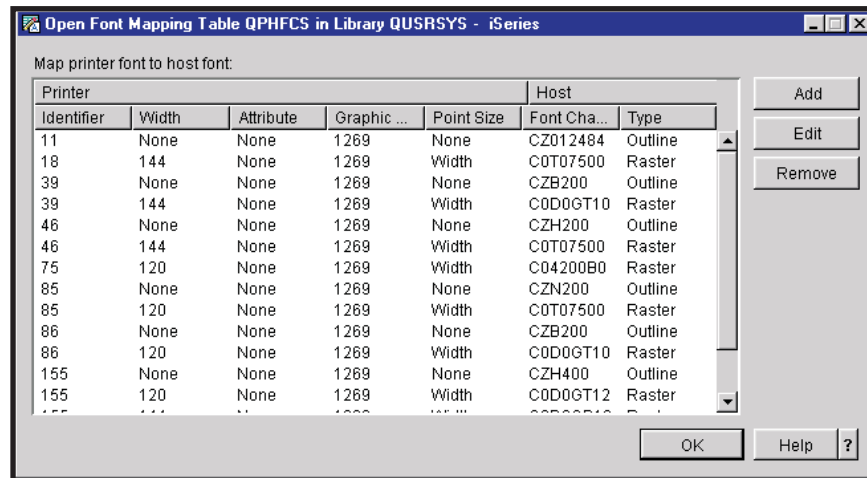


Figure 199. Open Font Mapping Table Window

7. (Optional) To add an entry to a user-defined table, click **Add**. This opens a new window. Fill in the appropriate values and click **OK**.
8. (Optional) To edit an entry in a user-defined table, select the entry and click **Edit**. This opens a new window. Fill in the new values for the font to map to and click **OK**.
9. (Optional) To remove an entry from a user-defined table, select the entry and click **Remove**.

10. For help press F1 or click **Help**.
11. Click **OK**.

Creating a Font Mapping Table

You can create user-defined font mapping tables with AFP Manager by following the steps below. You can only create one font mapping table of each kind (except printer-to-printer font). The table is named by the system *PHFCS, *HPFCS, *PHCP, *HPCP as appropriate, and stored in QUSRSYS. You can name printer resident font to printer resident font mapping tables and specify the library to store them in.

1. Double-click the iSeries Access icon.
2. Double-click the iSeries Navigator icon.
3. Open a connection to an OS/400.
4. Expand **AFP Manager** then **Font Mapping Tables**.
5. Right-click **User Tables** and select **New....** The window shown in Figure 200 opens.

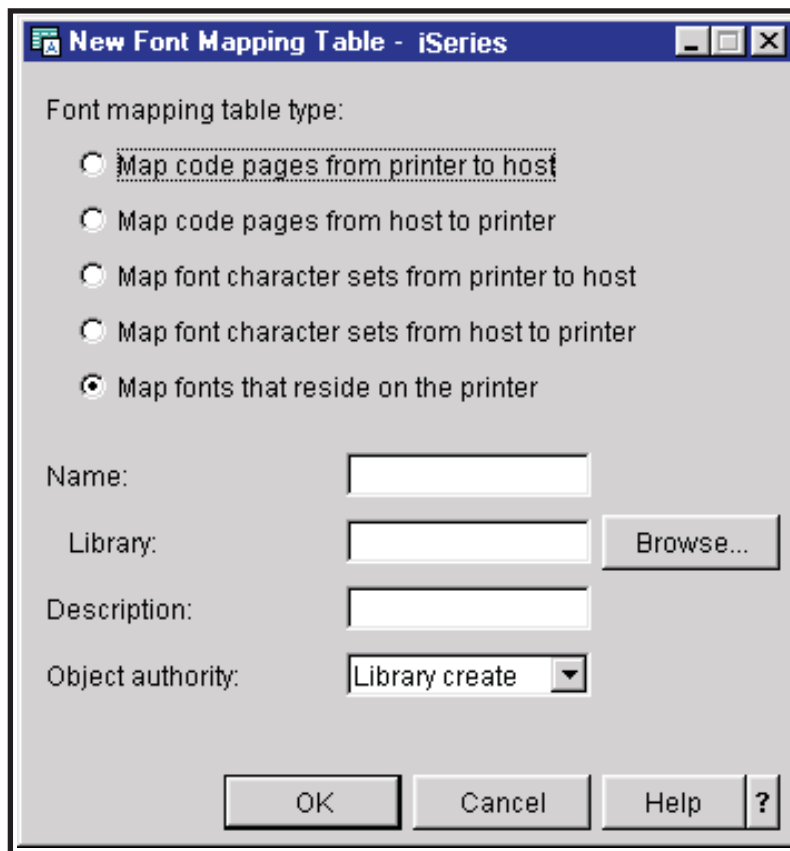


Figure 200. New Font Mapping Table Window

6. Specify the type of font mapping table to create. If you want to map printer fonts to printer fonts, give it a name and specify what library to put it in. You cannot specify the name or library for any other kind of font mapping table because you can only create one of each type. For help press F1 or click **Help**.
7. Click **OK**.
8. To add entries to the table, follow steps 5 through 7 in “Viewing and Changing Font Mapping Tables” on page 349.

Activating Font Capture

To use AFP Manager to activate font capture, follow these steps:

1. Mark the font as eligible for capture by following these steps. You can only mark a font that you are importing to your OS/400 as eligible for capture.
 - a. Follow steps 1 through 4 in “Importing Resources to the OS/400” on page 273.
 - b. Right-click **Code pages** or **Font character sets**, and select **Import**. A window like the one shown in Figure 201 opens, depending on the type of resource you selected.

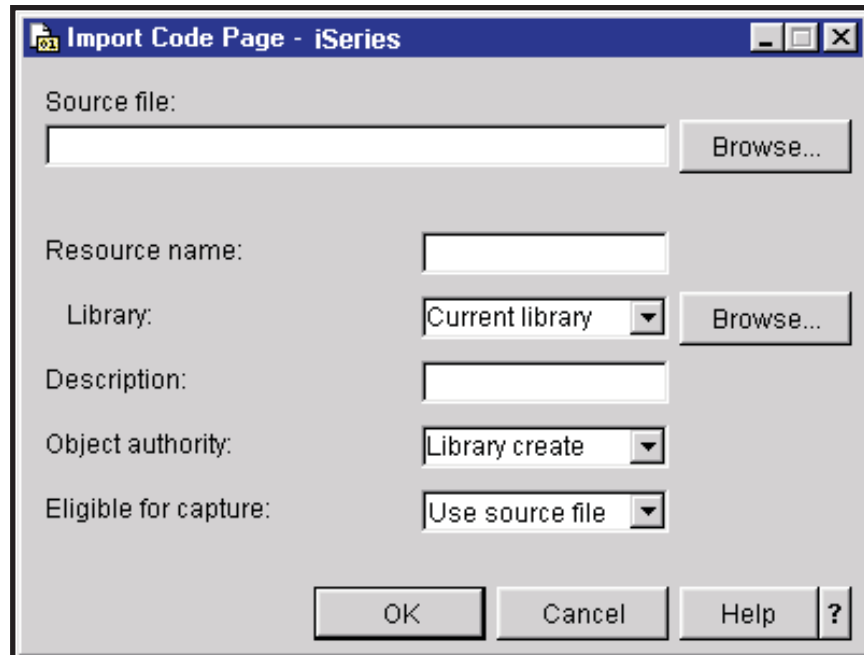


Figure 201. Import Code Page Window

- c. Fill in the fields as appropriate. By **Eligible for capture**, select **Yes**. For help press F1 or click **Help**.
 - d. Click **OK**.
2. Activate font capturing for eligible fonts by following these steps.

Note: These instructions are for changing an existing PSF configuration object. To create a PSF configuration object, follow the steps in “Creating a PSF Configuration Object” on page 271.

- a. Double-click the iSeries Access icon.
- b. Double-click the iSeries Navigator icon.
- c. Open a connection to an OS/400.
- d. Expand **AFP Manager** then **PSF Configurations**.
- e. Right-click the PSF configuration object you want to change and select **Configure**.

- f. On the **Resources** page, make sure that **Save downloaded fonts on printer** is selected as shown in Figure 202. For help, press F1 or click **Help**.

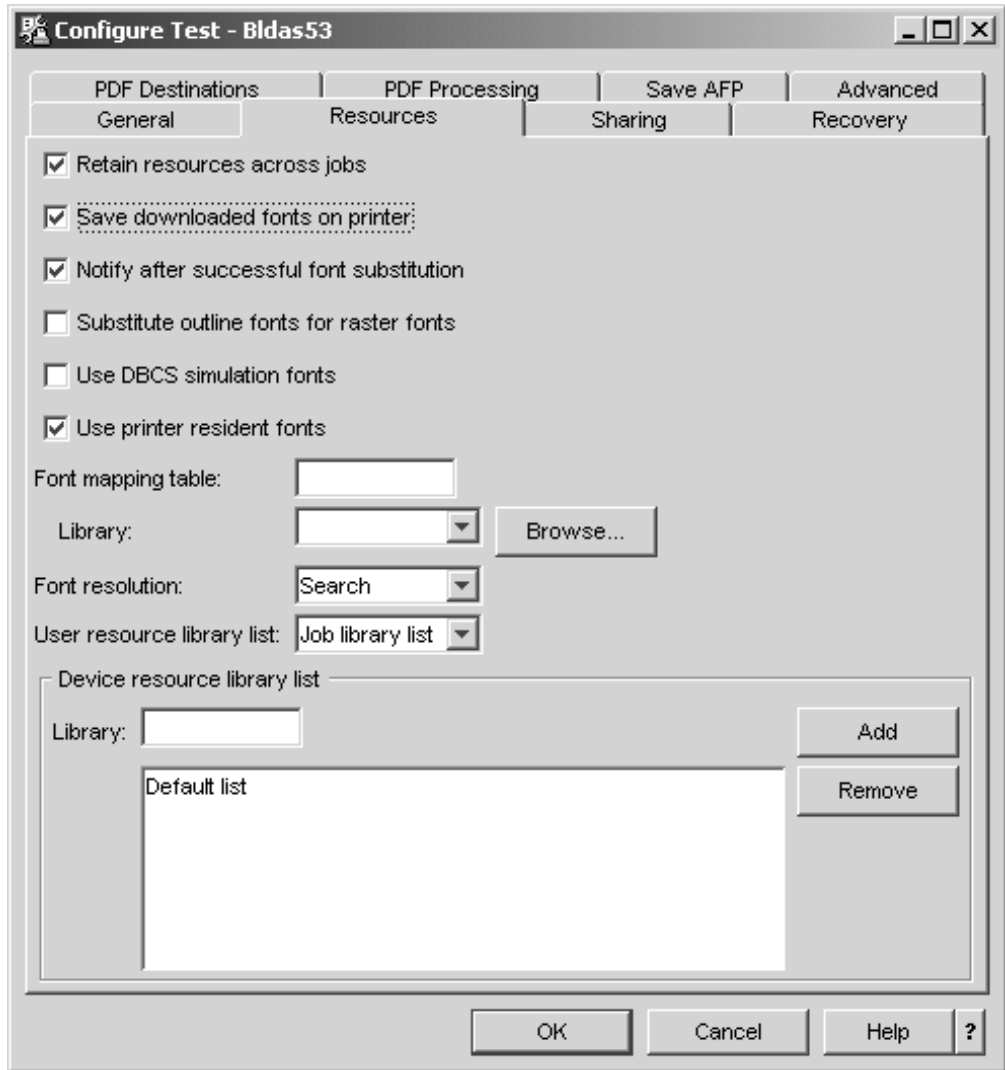


Figure 202. Enabling Font Capture

- g. Click **OK**.

Appendix G. Finding OS/400 Objects in the Integrated File System

This appendix introduces the integrated file system and discusses how find your stored OS/400 objects. “How PSF Searches for Resources” on page 357 describes how PSF searches for the resources you request in your documents.

Introducing the Integrated File System

The integrated file system is a part of OS/400 that supports stream input and output, and storage management similar to PC and UNIX[®] operating systems, while providing a structure over all information stored in your server. The integrated file system:

- Has support for storing information in stream files that can contain long continuous strings of data. These strings of data might be the text of a document or the picture elements in a picture. The stream file support is designed for efficient use in client/server applications.
- Has a hierarchical directory structure that lets you organize objects like fruit on the branches of a tree. You specify the path through the directories to accesses an object.
- Has a common interface that lets users and applications access these when they are stored in your server:

Stream files	Documents	Libraries	Objects
Database files	Resources	Folders	

- Provides a common view of stream files that are stored in these locations:

locally on your server	a Network File System server
an integrated xSeries™ server	a remote iSeries server
a Novell NetWare server	remotely on a Local Area Network (LAN) server
a remote Windows NT server	

A file system helps you access specific segments of storage that are organized as logical units. The logical units on a server are libraries, directories, files, and objects.

The integrated file system contains these file systems. Unless otherwise noted, they are stream-file based:

Network File System (NFS)

Lets you access a remote NFS server.

Document Library Services (QDLS)

Contains documents and folders.

QFileSvr.400

Lets you access other file systems on remote iSeries servers.

Netware File System (QNetWare)

Lets you access local and remote data on Novell.

Windows NT (QNTC)

Lets you access data and objects stored on Windows NT servers.

QOpenSys

Lets you access UNIX-based information in formats such as POSIX and XPG. This file system is case sensitive.

Optical file System (QOPT)

Lets you access stream data stored on optical media.

Library File System (QSYS.LIB)

Contains iSeries database files and other objects, with objects limited to 16 MB. This is a record-based file system. However, when you access it through the integrated file system, the database file members in this file system look like stream files to your application.

"root" (/) File System ("root")

Gives DOS and OS/2[®] stream file support.

User-Defined File System (UDFS)

File system that you create and manage.

Finding an Object in the Integrated File System

You use paths to store and locate objects in the integrated file system. There are two basic ways to specify paths:

- **Absolute (Full) Path Name:** An absolute path name starts at the highest ("root") level. For example, /Dept2/Photos/Smith.
- **Relative Path Name:** A relative path name begins at the current directory. For example, if you are in the Dept2 directory, you can get to the same place specified with the absolute path name above by specifying Photos/Smith.

File systems can define how deep a path can go. For example, these common OS/400 file systems have the given levels of hierarchy:

QSYS.LIB

Lets you specify a maximum of three levels of directory hierarchy. For example, QSYS.LIB/MyLib.LIB/MyFile.FILE/MyMember.MBR

/ ("root")

Has no maximum number of levels of directory hierarchy. For example, you could specify DIR1/DIR2/DIR3/...DIRn/object.

QDLS Lets you specify a maximum of 32 levels of directory hierarchy.

All file systems have unique characteristics. For more information about the file systems in the integrated file system, refer to the *File Systems and Management* topic in the iSeries Information Center.

Moving Data between File Systems

There are several different ways you can move data between file systems, depending on your needs. These are some different tools you can use:

iSeries Navigator

iSeries Navigator is a graphical user interface that lets you work with your iSeries system from your workstation. To work with iSeries files and objects with iSeries Navigator, follow these steps:

1. Double-click the iSeries Access icon.
2. Double-click the iSeries Navigator icon.
3. Open a connection to an OS/400.

- Expand **File Systems** to display a list of all OS/400 file systems to which you are authorized, as shown in Figure 203.

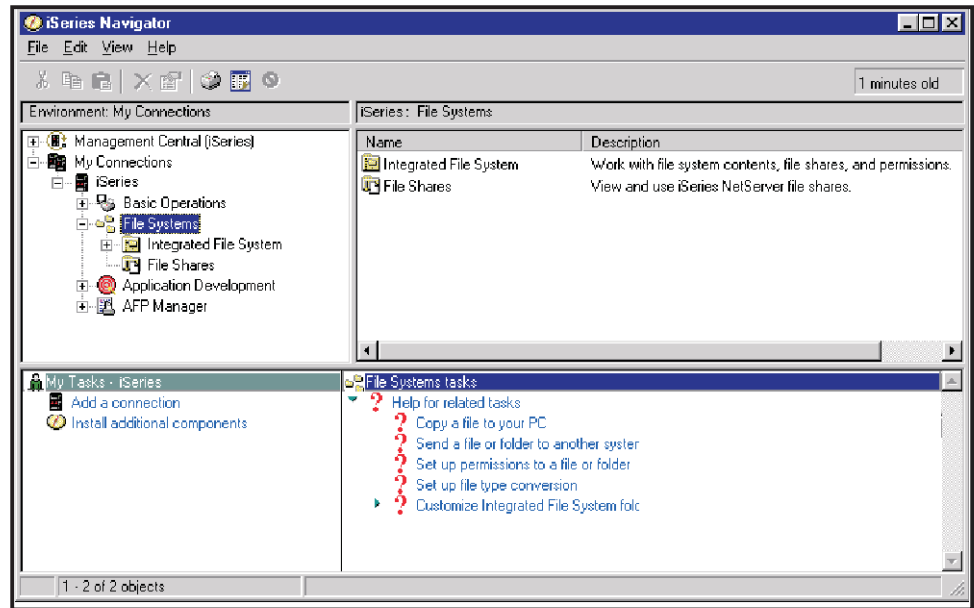


Figure 203. iSeries Navigator File Systems

- You can access OS/400 objects by expanding through the directories to the object. When you reach the object you want to work with, right-click for a list of options.

You can also drag and drop files within and between iSeries servers and the PC desktop and send files and folders to multiple systems using Management Central.

- For help press F1 or click **Help**.

OS/400 Command Line

OS/400 has a set of CL commands for working with files in the integrated file system. To view all of the available commands, enter Go DATA on the command line then select 5 “Integrated File System”. This screen appears:

```

FILESYS                               Integrated File System                               System:  ISERIES

Select one of the following:

    1. Directory commands
    2. Object commands
    3. Security commands

Selection or command
===>

F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel  F13=Information Assistant
F16=AS/400 Main menu
(C) COPYRIGHT IBM CORP. 1980, 2002.

```

From this screen you can browse the available commands. For example, if you enter 2 “Object commands”, you are given this list of object commands:

```

FSOBJ                               Object Commands                               System:  ISERIES

Select one of the following:

    1. Work with object links
    2. Display object links
    3. Copy object
    4. Rename object
    5. Move object
    6. Add link
    7. Remove link
    8. Check out object
    9. Check in object
   10. Copy to stream file
   11. Copy from stream file
   12. Save object
   13. Restore object

Selection or command
====>

F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel  F13=Information Assistant
F16=AS/400 Main menu

```

Choosing any of the commands from this screen brings up the window for that command. For example, selecting 3 “Copy object” brings up this window:

```

                                Copy Object (CPY)

Type choices, press Enter.

Object . . . . .
To directory . . . . . '.'
To object . . . . .

Symbolic link . . . . . *NO          *NO, *YES
From CCSID . . . . . *OBJ          1-65533, *OBJ, *PCASCI...
To CCSID . . . . . *OBJ          1-65533, *OBJ, *CALC...
Data Format . . . . . *BINARY      *BINARY, *TEXT
Directory subtree . . . . . *NODIR    *NODIR, *NONE, *ALL
Replace object . . . . . *NO          *NO, *YES
Owner . . . . . *NEW           *NEW, *KEEP

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

File Transfer Protocol (FTP)

The default file format for FTP is the traditional iSeries library system (QSYS.LIB). To FTP to and from the integrated file system, issue the command QUOTE SITE NAMEFMT 1 at the FTP command prompt. This changes the FTP protocol defaults to operate within the integrated file system.

If you choose not to change the FTP file format and use FTP within traditional iSeries library system, or if you change the file format but try to FTP within QSYS.LIB, you cannot use it to directly transfer objects such as programs and AFP resources. To use FTP to transfer non-supported objects, save the object in a save file (SAVF) and transfer the save file.

Windows Mapped Drive

If your PC is connected to an iSeries server, you can interact with the directories and objects of the integrated file system as if they were stored on the your PC. You can copy objects between directories by using the drag-and-drop capability of Windows Explorer. You can also copy an object from your iSeries server to the PC by selecting the object in the mapped server drive and dragging the object to the PC directory.

How PSF Searches for Resources

OS/400 resources, such as fonts and overlays, are stored in libraries. It is important that PSF be able to find the correct resources. This section describes how PSF searches for resources and how you can affect which libraries are searched.

PSF searches libraries specified in library lists and the system font libraries for resources. You can accept the default library list or specify a user resource library list and device resource library list.

Note: PSF does not use library lists to search for object containers. For information about how PSF searches for object containers, see “How PSF Searches for Object Containers” on page 358.

To make a resource library available to all users at all times, you have two options. You must stop and re-start PSF for these changes to take affect:

- Specify the library in a specific library list
- Rename the library QFNTnn, where *nn* is in the range 01 through 19.

When a resource is named without a library qualifier, PSF looks in these libraries in the order listed:

1. The job (interactive or batch) library list
2. System font libraries, as listed below:
 - QFNTCPL
 - QFNT01-QFNT19
 - QFNT61-QFNT69

You can override the system library list and specify a library list for a particular printer (device resource library list) or a user resource library list with a PSF configuration object. If you specify both a user library list and a device library list, PSF searches the lists in this order when looking for a resource:

1. User resource library list
2. Device resource library list

There is a two-fold value in using the user resource library and device resource library lists. They enhance usability because, for example, it lets you specify one set of libraries for a 240-pel printer and another set of libraries for a 300-pel printer (with the device resource library list). You also can ensure that the libraries that contain the resources needed by the specified job are in the library list that PSF will use. Using the library lists also can improve performance because you can have PSF search only the necessary libraries.

Figure 204 on page 358 illustrates how PSF searches for resources.

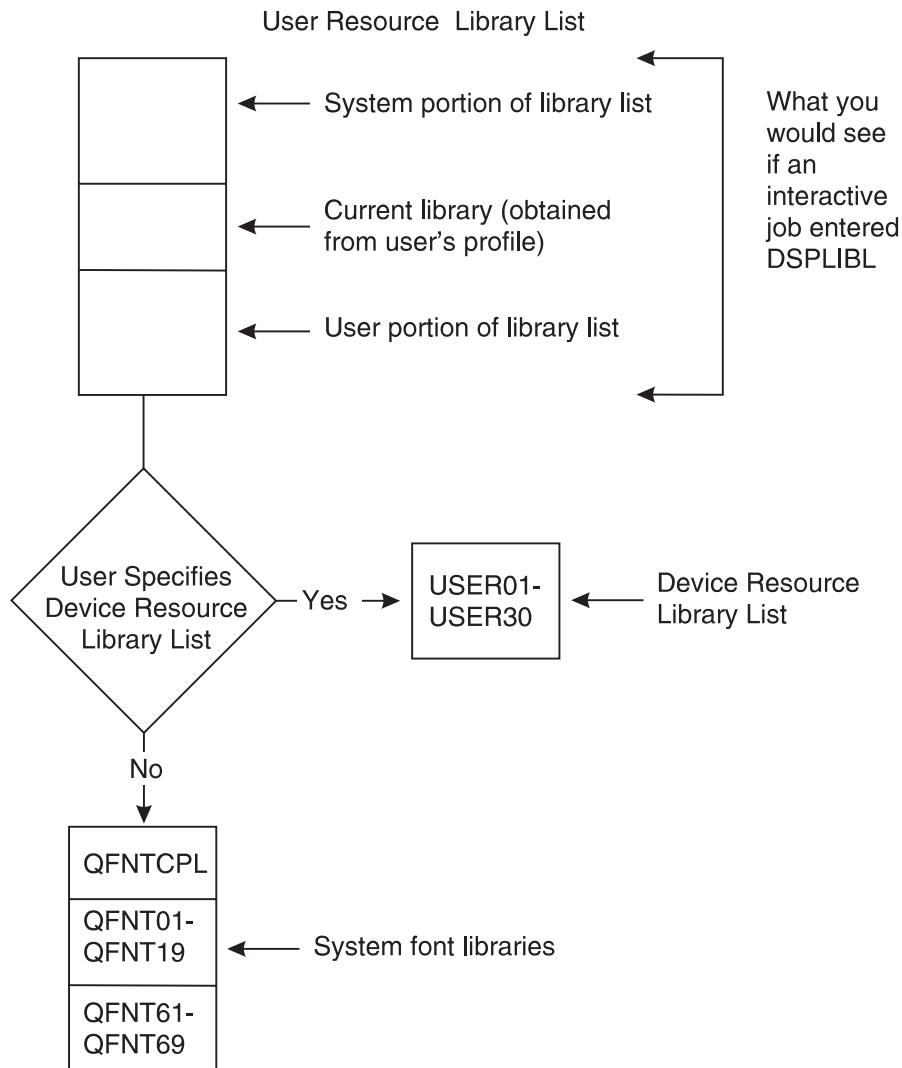


Figure 204. How PSF Searches for Resources

How PSF Searches for Object Containers

This is how PSF searches for object containers. Notice that it does not use library lists:

1. PSF first determines whether the object is inline. If so, the inline resource is used and no further searching is done.
2. If the DDS AFPRSC keyword specifies a path, that path is searched for the resource.
3. The path specified with the system-level value for environment variable QIBM_AFP_RESOURCES_PATH is searched.
4. If the resource is not found, and the spooled file resides on an independent disk pool, the /<independent-disk-pool-name>/QIBM/UserData/OS400/AFPresources directory, if it exists, is searched. You are responsible for creating directory /QIBM/UserData/OS400/AFPresources on an independent disk pool. Subdirectories are not searched.

- If the resource is not found, or the spooled file resides on *SYSBAS, the /QIBM/UserData/OS400/AFPresources directory on the system ASP is searched. Subdirectories are not searched.

Specifying Resource Library Lists

You specify both user and device resource library lists on a PSF configuration object.

Specify a user resource library list in a PSF configuration object on the User resource library list (USRRSCLIBL) parameter. To use a device resource library list, specify it on the Device resource library list (DEVRSCLIBL) parameter. This is the screen that opens when you enter CRTPSFCFG on the OS/400 command line:

```

Create PSF Configuration (CRTPSFCFG)

Type choices, press Enter.

PSF configuration . . . . . Name
Library . . . . . *CURLIB Name, *CURLIB
User resource library list . . *JOBLIBL *JOBLIBL, *CURLIB, *NONE
Device resource library list . *DFT Name, *DFT
      + for more values
IPDS pass through . . . . . *NO *NO, *YES
Activate release timer . . . . *NORDYF *NORDYF, *IMMED...
Release timer . . . . . *NOMAX 1-1440, *NOMAX, *SEC15...
Restart timer . . . . . *IMMED 1-1440, *IMMED
APPC and TCP/IP retry count . . 15 1-99, *NOMAX
Delay between APPC retries . . . 90 0-999
Acknowledgment frequency . . . 100 1-32767
Printer response timer . . . . *NOMAX 5-3600, *NOMAX
Generate PDF output . . . . . *NONE *NONE, *SPLF, *STMF, *MAIL

Bottom
F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys

```

Example:

To change the PSF configuration object MY_CONFIG so that current library for the job that created the spooled file is used to search for AFP resources first (the user resource library list), then libraries LIB1 and LIB2 are searched (the device resource library list), enter this command:

```
CHGPSFCFG PSFCFG(MY_CONFIG) USRRSCLIBL(*CURLIB) DEVRSCLIBL(LIB1 LIB2)
```

Using AFP Manager to Specify Resource Library Lists

You can use AFP Manager to create or change a PSF configuration object, including specifying resource library lists. AFP Manager is part of iSeries Navigator.

Follow these steps to specify a user resource library list and a device resource library list on an existing PSF configuration object using AFP Manager. You can also use AFP Manager to create a new PSF configuration object. For steps see "Creating a PSF Configuration Object" on page 271.

- Double-click the iSeries Access icon.
- Double-click the iSeries Navigator icon.
- Open a connection to a 5.1 or higher OS/400.
- Expand **AFP Manager** then **PSF Configurations**.

5. From the list of PSF configuration objects, right-click the one you want to modify and select **Configure....**
6. On the Resources page, select a value for **User resource library list**.
7. The default device resource library list is automatically specified. To specify different libraries, put the library name in the empty field and click **Add**. Add up to 30 libraries, one at a time.
8. For help press F1 or click **Help**.
9. Click **OK**.

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 about 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
500 Columbus Avenue
Thornwood, NY 10594-1785
U.S.A.

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
Mail Drop 001W
Boulder, CO 80301
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 information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

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.

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. 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 IBM's application programming interfaces.

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 publication is intended to help you to (INSERT TASK HERE).

This publication also documents General-Use Programming Interface and Associated Guidance Information.

General-Use programming interfaces allow the customer to write programs that obtain the services of (INSERT PRODUCT NAME HERE).

General-Use Programming Interface and Associated Guidance Information is identified where it occurs, either by an introductory statement to a chapter or section or by the following: **General-Use Programming Interface:**

Trademarks

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

400	IPDS
Advanced Function Presentation	iSeries
AFCCU	MVS
AFP	OfficeVision
AIX	OfficeVision/400
APL2	Operating System/400
Application System/400	OS/2
AS/400	OS/390
BookMaster	OS/400
COBOL/400	Print Services Facility
DB2 Universal Database	Redbooks
GDDM	RPG/400
IBM	WebSphere
ImagePlus	xSeries
Infoprint	z/OS
Intelligent Printer Data Stream	

Lotus and Domino are trademarks of Lotus Development Corporation in the United States, other countries, or both.

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

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

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

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

Glossary

This glossary defines technical terms and abbreviations that are used in this document. If you do not find the term you are looking for, refer to the index of this publication or view *IBM Dictionary of Computing*, located at: <http://www.ibm.com/networking/nsg/nsgmain.htm>.

Definitions reprinted from the *American National Dictionary for Information Processing Systems* are identified by the symbol (A) following the definition.

Definitions reprinted from a published section of the International Organization for Standardization (ISO) *Vocabulary—Information Processing* or from a published section of *Vocabulary—Office Machines* developed by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Committee (ISO/IEC JTC1/SC1) are identified by the symbol (I) following the definition. Because many ISO definitions are also reproduced in the *American National Dictionary for Information Processing Systems*, ISO definitions may also be identified by the symbol (A).

Definitions that are specific to IBM products are so labeled—for example, “In SNA,” or “In the 3820 printer.”

These cross-references are used in this glossary:

- **Contrast with.** Refers to a term that has an opposite or substantively different meaning.
- **See.** Refers to multiple-word terms in which this term appears.
- **See also.** Refers to related terms that have similar, but not synonymous, meanings.
- **Synonymous with.** Appears in the commentary of a preferred term and identifies less desirable or less specific terms that have the same meaning.

A

ACIF. AFP Conversion and Indexing Facility.

addressable point. Any point in a presentation surface that can be identified by a coordinate from the coordinate system of the presentation medium. See also pel.

Advanced Function Common Control Unit. A high-speed print controller. It can quickly resolve AFP data streams and resources into completed pages.

Advanced Function Presentation (AFP). A set of licensed programs that use the all-points-addressable concept to print data on a wide variety of printers or display data on a variety of display devices. AFP also includes creating, formatting, archiving, viewing, retrieving, and distributing information.

Advanced Function Presentation Application Programming Interface. An AFP program shipped with PSF/MVS 2.1.1 and later and PSF/VM 2.1.1 that creates the AFP data stream from the COBOL and PL/1 high-level programming languages.

Advanced Function Presentation data stream. A presentation data stream that is processed in the AFP environment. MO:DCA-P is the strategic AFP interchange data stream. IPDS is the strategic AFP printer data stream.

Advanced Function Presentation Utilities for iSeries (AFP Utilities). An IBM licensed program that includes a group of utilities that work together to provide Advanced Function Presentation on the iSeries.

Advanced Print Utility (APU). Part of the AFP PrintSuite family of application enablers that allow end-users to migrate existing application output to advanced electronic documents.

AFCCU. Advanced Function Common Control Unit.

AFP. Advanced Function Presentation.

AFP API. Advanced Function Presentation Application Programming Interface.

AFP Conversion and Indexing Facility. An AFP program you can use to convert a print file into a MO:DCA-P document, to retrieve resources used by the document, and to index the file for later retrieval and viewing.

AFPDS. AFP data stream; a synonym for the composed page, MO:DCA-P based data stream interchanged in AFP environments.

AFP Manager. A component of iSeries Navigator. AFP Manager lets you work with these objects associated with AFP output: PSF configuration objects, font mapping tables, and resources. iSeries Navigator is a feature of iSeries Access.

AFP Utilities for iSeries (AFP Utilities). Advanced Function Presentation Utilities for iSeries

AFP Workbench for Windows. A platform for the integration of AFP enabling applications and services. The AFP Viewer application is a Workbench tool.

all points addressable (APA). The capability to address, reference, and position text, overlays, and images at any defined point (pel) on the printable area of the paper. See *page mode*.

American National Standard Code for Information Interchange. A standard code, using a coded character set consisting of 7-bit coded characters (8-bits, including the parity check), that is used for information interchange among data processing systems, data communication systems, and their associated equipment. The ASCII set consists of control characters and graphic characters.

APA. All points addressable

APA printers. Devices that are all points addressable; in other words, devices that print with picture elements on the printing medium at any valid location on a sheet of paper.

APPC. Advanced program-to-program communication. APPC is the implementation of the SNA LU session type 6.2 architecture.

application program. A program written for or by a user that applies to the user's work, such as a program that does inventory control or payroll.

application programmer. A programmer who is responsible for writing programs for specific applications. The application programmer takes application data and writes programs to print it on line and AFP printers.

Application System/400®. The former name of the hardware on which OS/400 runs.

APU. Advanced Print Utility

APU Monitor. A program that, when processed, applies print definitions to selected spooled files so that the spooled files are automatically converted to AFP print files.

architecture. The set of rules and conventions that govern the creation and control of data types such as text, image, graphics, font, fax, color, audio, bar code, and multimedia.

ASCII. American National Standard Code for Information Interchange

AS/400. Application System/400

B

bar code. A code representing characters by sets of parallel bars of varying thickness and separation that are read optically by transverse scanning. (l)

baseline. In a font, the imaginary line on which successive characters are aligned in the inline direction.

batch. (1) A group of records or data processing jobs brought together for processing or transmission. (2) Pertaining to activity involving little or no user action. Contrast with interactive.

batch environment. The environment in which non-interactive programs are executed. The environment schedules their execution independently of their submitter. Contrast with interactive environment.

burst. To separate continuous-form paper into separate sheets.

C

ccsid. Coded character set identifier.

character. (1) A symbol used in printing. For example, a letter of the alphabet, a numeral, a punctuation mark or any other symbol that represents information. (2) A byte of data.

character graphic. The visual representation of a character, defined by toned or untoned picture elements (pels). **Note:** An untoned pel (a reverse character) is visually represented by the toned pels around it.

character increment. The distance the current print position is increased by printing the current character graphic.

character rotation. The alignment of a character relative to the baseline, measured in degrees in a clockwise direction. Examples are 0°, 90°, 180°, and 270°.

characters per inch. A unit of measurement for the width of a printed character, reflecting the number of times a graphic character can be set in 1 inch. For example a 10-characters per inch (CPI) font can print 10 characters in one inch. Uniformly spaced fonts are measured in characters per inch.

character set. A collection of characters that is composed of some descriptive information and the character shapes themselves. Synonymous with *font character set*.

code page. A font component that associates code points and character identifiers. A code page also identifies how undefined code points are handled.

code point. A 1-byte code representing one of 256 potential characters.

coded font. A font library member that associates a code page with a font character set. For double-byte fonts, a coded font associates more than one pair of code pages and font character sets.

command. A request from a terminal or a specification in a batch processing job for the performance of an operation or the execution of a particular program.

composition. The act or result of formatting a document.

conditional processing. A page definition function that enables input data records to partially control their own formatting.

constant data. (1) Data with a value that does not change. (2) Data that has an unchanging, predefined value to be used in processing. A constant does not change during execution of a program, but the contents of a field or variable can. Contrast with variable data.

continuous-forms paper. A series of connected forms that feed continuously through a printer. The connection between the sheets is perforated to enable the user to tear them apart. Before printing, the sheets are folded in a stacked arrangement, with the folds along the perforations. (Note that some continuous forms are in rolls and are not folded.) Contrast with cut-sheet paper.

copy. The specification level of an APU print definition where most layout work, such as specifying page layout options, selecting and placing images, and defining constants and boxes, is done.

cpi. characters per inch

cut-sheet emulation. Printing sheets side-by-side on either a simplex or duplex continuous forms printer that supports cut-sheet emulation. The output, when slit and collated by post-processing equipment, looks like output generated by a cut-sheet printer.

cut-sheet paper. Paper that is cut into separate sheets before being printed on. Contrast with *continuous-forms paper*.

D

Data Description Specifications (DDS). Provides a set of OS/400 keywords that let format output external to the application program. For example, you can draw shapes, specify fonts, include overlays, and specify formatting with DDS keywords.

data object resource. Object containers or IOCA images that are sent to the printer in the IPDS home state. Object containers or IOCA images sent to the printer in this state are saved as data object resources to be used later while processing a page or overlay.

data processing. The systematic performance of operations upon data; for example, handling, merging, sorting, and computing. (I) (A)

data set. Synonym for *file*.

data stream. (1) All data transmitted through a data channel in a single read or write operation. (2) A continuous stream of data elements being transmitted, or intended for transmission, in character or binary-digit form, using a defined format. (3) Records sent to PSF from the print files and the resource libraries.

database. A set of data, part or the whole of another set of data that consists of at least one file, and that is sufficient for a given purpose or for a given data-processing system. (I) (A)

database file. A record arranged file; It has predefined subdivisions consisting of one or more fields that have specific characteristics, such as length and data type. Also called "record-based file" Commonly used when fields need to be individually accessed and manipulated. The OS/400 object type is *FILE. Contrast with *stream file*.

DBCS. See *double-byte character set*.

DDS. See *Data Description Specifications*.

direction. The print position of data on a logical page, line, or field. The ultimate reference point for all direction controls on a page is the hardware origin. Secondary and tertiary reference points are possible as well, enabling more than one print direction on a page.

ditroff. A file format consisting of device-independent data produced by the troff utility. See *troff*.

document. (1) A publication or other written material pertaining to a specific subject or related subjects. (2) In word processing, a collection of one or more lines of text that can be named and stored as a separate entity.

double-byte character set (DBCS). A font character set that can contain up to 65536 characters. Languages with non-alphabetic writing systems, such as Chinese, Japanese, and Korean, require double-byte character sets.

duplex printing. Pertaining to printing on both sides of a sheet of paper. Contrast with simplex printing.

E

electronic overlay. Synonym for *overlay*.

end user. (1) A person, device, program, or computer system that uses a computer network for the purpose of data processing and information exchange. (T) (2) A person who writes and creates documents. The end user has little or no programming training but knows

how to use a terminal for text processing. Examples of end users include secretaries, writers, and engineers.

external formatting. Controls for the placement of data on the page that are embedded outside the actual application program.

F

field. In a record, a specified area used for a particular class of data; for example, a group of character positions used to enter or display wage rates on a screen. (T)

file. (1) A named set of records stored or processed as a unit. (T) (2) The major unit of data storage and retrieval, consisting of a collection of data in one of several prescribed arrangements and described by control information to which the operating system has access.

font. A family or assortment of characters of a given size and style; for example, 9-point Sonoran Serif roman medium. (A)

font administrator. A person who is responsible for installing and maintaining the fonts stored in computer resource libraries.

font character set. See *character set*.

font metrics. Measurement information that defines individual character values such as height, width, and space as well as overall font values such as the average and maximum heights and widths of characters. Font metrics can be expressed in specified fixed units, such as pels, or in relative units that are independent of both the resolution and size of the font.

font substitution. When a requested font cannot be found and a different font is used.

form. (1) The paper on which output data is printed by a line printer or a page printer. (2) A physical sheet of paper. See *preprinted form*.

form definition. A resource that defines the characteristics of the form which include overlays to be used (if any), text suppression, the position of page data on the form, and the number and modifications of a page. Contrast with *page definition*.

format. (1) A specified arrangement of such things as characters, fields, and lines, usually used for displays, printouts, or files. (2) To arrange such things as characters, fields, and lines. (3) (v.) To prepare a document for printing in a specified format.

formatter. A computer program that prepares a source document for printing.

forms designer. A person who is responsible for designing electronic or preprinted forms that are

readable, usable, and attractive. The forms designer usually has training in graphics design and in the presentation of information.

ftp. File transfer protocol

fully resolved AFP. See *portable AFP*.

G

GDDM. Graphical Data Display Manager

graphic. Image, text, or a combination of both that can be placed on the printed page.

Graphical Data Display Manager (GDDM). An IBM licensed program containing utilities for creating, saving, editing, and displaying visual data such as page segments, charts, images, vector graphics, composites (of text, graphics, and images), and scanned data.

graphics designer. A person who is responsible for the design and appearance of graphics used in a company's documents. The graphics designer has experience in graphics design as well as in using computers to create graphics.

H

hardware. Physical equipment as opposed to programs, procedures, rules, and associated documentation. (I) (A) Contrast with *software*.

host-based computer. (1) In a computer network, a computer that provides end users with services such as computation and data bases and that usually performs network control functions. (T) (2) The primary or controlling computer in a multiple-computer installation.

I

IBM-supplied PDF mapping program. A PDF mapping program supplied with Infoprint Server and designed to be used with a map object that you create.

image. A pattern of toned and untoned pels that form a picture.

impact printer. A device in which printing results from mechanical impacts. (I) (A) Contrast with *nonimpact printer*.

index. (1) A process of segmenting a print file into uniquely identifiable groups of pages (a named collection of sequential pages) for later retrieval. (2) A process of matching reference points within a file and creating structured field tags within the MO:DCA-P document and the separate index object file.

index object file. An index-information file created by Advanced Function Presentation Conversion and

Indexing Facility that contains Index Element (IEL) structured fields, which identify the location of the tagged groups in the AFP file. The indexing tags are contained in the Tagged Logical Element (TLE) structured fields.

Intelligent Printer Data Stream (IPDS). A host-to-host printer data stream for Advanced Function Presentation subsystems. It provides an interface to all-points-addressable printers that makes possible the presentation of pages containing an architecturally unlimited mixture of different data types.

intelligent routing. The ability to distribute output from Infoprint Server's PDF subsystem in multiple ways.

interactive. Pertaining to an application in which entries call forth a response from a system or program, as in an inquiry system. An interactive system might also be conversational, implying a continuous dialog between the user and the system. Interactive systems are usually communicated with through terminals, and respond immediately to commands.

interactive environment. An environment in which a terminal user interacts with the system. Contrast with batch environment.

IPDS. Intelligent Printer Data Stream

iSeries. The name of the hardware on which OS/400 runs.

iSeries Information Center. A Web Site that is a source for technical information about the IBM iSeries 400® server. The Information Center is your starting point for all iSeries technical information. The books from the Online Library are part of the Information Center. Occasionally, the information directs you to the iSeries Technical Studio, IBM Redbooks™, and other iSeries Web sites. The iSeries Information Center can be found at <http://www.as400.ibm.com/infocenter>.

L

LAN. Local area network

LAN administrator. A person responsible for installing, configuring, and maintaining Local Area Networks on which are installed workstations and printers.

library. A file or a set of related files; for example, a page definition library containing one or more page definition files.

licensed program. A utility that performs a function for the user and usually interacts with and relies upon system control programming or some other IBM-provided control program. A licensed program contains logic related to the user's data and is usable or adaptable to meet specific requirements.

line data. Data prepared for printing on a line printer such as an IBM 3800 Model 1 Printing Subsystem. Line data is usually characterized by carriage-control characters and table reference characters. Line data adds blank lines for vertical positioning and a couple of additional carriage control characters to each print record. There are two types of line data: *traditional* and *record format*. When the term "line data" is used in this publication, the information applies to both types of line data, unless otherwise specified. Contrast with *MO:DCA-P data* and *SCS data*.

line data print file. A file that consists of line data, optionally supplemented by a limited set of structured fields.

line printer. A device that prints a line of characters as a unit. Contrast with *page printer*.

lines per inch (lpi). (1) A unit of measurement for the specification of baseline placement. (2) A measure of the number of lines per vertical inch of paper.

local area network. A data network located on the user's premises in which serial transmission is used for direct data communication among data stations. (T)

logical page. A presentation space. One or more object areas or data blocks can be mapped to a logical page. A logical page has specifiable characteristics, such as size, shape, orientation, and offset and is rectangular in shape. Orientation and offset are specified relative to a medium coordinate system.

lowercase. Pertaining to small letters, as distinguished from capitals; for example, a, b, g, rather than A, B, G.

lpi. Lines per inch

M

magnetic ink character recognition (MICR).

Recognition of characters printed with ink that contains particles of a magnetic material.

mail tag. See *routing tag*

map object. An object used with Infoprint Server that passes values to an IBM-supplied *PDF mapping program*. It is created by using OS/400 panels or APIs. Any time a PDF mapping program is used, it can be replaced with an IBM-supplied PDF mapping program and map object.

mapping program. See *PDF mapping program*

merged stream file. See *portable AFP*.

MICR. Magnetic ink character recognition

Mixed Object Document Content Architecture. A strategic, architected, device-independent data stream for interchanging documents.

mixed-pitch font. A font that simulates a proportionally spaced font. The characters are in a limited set of pitches (for example, 10 pitch, 12 pitch, and 15 pitch).

MO:DCA-P. Mixed Object Document Content Architecture

monospaced font. A font in which the graphic characters have a uniform character increment. Synonymous with uniformly spaced font. Contrast with proportionally spaced font.

multiple-up printing. The printing of more than one page on a single surface of a sheet of paper.

N

nonimpact printer. A device in which printing is not the result of mechanical impacts; for example, thermal printers, electrostatic printers, photographic printers. (I) (A) Contrast with impact printer.

O

object container. A generic MO:DCA-P envelope that can contain some non-OCA objects, such as EPS image objects, PDF single-page objects, and color mapping tables.

object format. The format of AFP resources required for use by PSF. Contrast with source format.

offset stacking. A function that enables the printed output pages to be offset for easy separation of the print jobs.

OGL/370. Overlay Generation Language/370

OnDemand for iSeries. An IBM licensed program that enables you to archive, retrieve, view, and print AFP documents. This application features a graphical user interface.

Operating System/400® (OS/400). An IBM licensed program that can be used as the operating system for the iSeries processor series.

orientation. (1) The angle between the top or bottom edge of the page and the baselines within a column, measured in a clockwise direction. (2) The rotation of an element relative to a fixed reference.

OS/400. Operating System/400

outline font. A font whose graphic character shapes are defined mathematically rather than by raster patterns.

output device. A machine used to print, display, or store the result of data processing.

overlay. A resource that can contain text, image, graphics, and bar code data. An overlay is electronically created in the host processor, stored in a library, and can be merged electronically with variable data on a sheet during printing. See also preprinted form and forms flash.

Overlay Generation Language/370 (OGL/370). An IBM licensed program used to create overlays.

P

page. A collection of data that can be printed on a physical sheet of paper.

page definition. A resource containing a set of formatting controls for printing logical pages of data. Includes controls for number of lines per printed sheet, font selection, print direction, and mapping individual fields in the data to positions on the printed sheets.

page format. (1) A subset of a page definition, containing controls governing the arrangement of text on a page. (2) In APU, the object that contains all of the instructions for formatting a print job. A print definition can name one or more page formats.

page mode. The mode of operation in which the printer can accept a page of data at a time from a host processor to be printed on an all-points addressable output device. Data may consist of pages composed of text, images, overlays, or page segments.

page printer. Any of a class of printers that accepts MO:DCA-P pages, constructed of composed text and images, among other things. Contrast with line printer.

Page Printer Formatting Aid (PPFA). An IBM licensed program that you can use to create and store form definitions and page definitions.

page segment. A resource containing composed text and images, prepared before formatting and included during printing.

PDF mapping program. In Infoprint Server, a program that specifies what to do with the output from the PDF subsystem. This can be a user-created program or can be IBM-supplied. Using the IBM-supplied mapping program requires a *map object*.

pel. picture element

pel density. The number of pels per unit of linear measurement.

personal computer. A desktop, floor-standing, or portable microcomputer that usually consists of a system unit, a display monitor, a keyboard, one or more diskette drives, internal fixed-disk storage, and an optional printer. Personal computers are designed primarily to give independent computing power to a single user or small businesses. **Note:** Personal

computers are designed primarily for stand-alone operation but can be connected to mainframes or networks.

PFU. Print Format Utility

physical page. The side of a sheet of paper that is to be printed on.

pica. A unit of about 1/6 inch used in measuring typographical material.

picture element. The smallest area that can be individually toned on a printing medium or on a display surface.

pitch. characters per inch

plotter. An output unit that presents data in the form of a two-dimensional graphic representation. (I) (A)

point. In printing, a unit of about 1/72 of an inch used in measuring typographical material. Each pica contains 12 points.

point size. The height of a font in points.

portable AFP. AFP data packaged with the resources necessary for printing and indexing information, if any exists. The Infoprint Server for iSeries Create AFP Data command and IBM Content Manager OnDemand for iSeries create fully-resolved AFP data.

postprocessing option. A hardware device that attaches to the output side of a printer; for example, an envelope stuffer, binder, or stapler.

PostScript. A page description language with interactive graphics capabilities that was developed by Adobe Systems, Incorporated.

PPDS. Page Printer Data Stream

PPFA. Page Printer Formatting Aid

preprinted form. A sheet of paper containing a preprinted design of constant data. Variable data can be merged on such a form.

preprocessing option. A hardware device that attaches to the input side of a printer; for example, a paper-roll feed or multiple input bins.

print data set. Synonym for print file.

print definition. Contains instructions for transforming simple SCS print output to advanced AFP output. A print definition includes the specifications for remapping existing print data, defining and creating different page formats and copies, and adding document elements such as overlays, images, fonts, bar codes, and constants.

print file. A file created by an application program that contains the actual information to be printed and some of the data that controls the format of the printing. Print files can contain MO:DCA-P data, line data, or a combination of MO:DCA-P and line data.

Print Format Utility (PFU). Part of the Advanced Function Presentation Utilities for iSeries. This utility enables you to print data from database file members in various forms, such as text or bar codes.

print job. The data to be printed that is submitted to Print Services Facility by the user.

Print Services Facility (PSF). An IBM licensed program that produces printer commands from the data sent to it.

printer driver. A program that passes commands and resources with a data stream from the system spool to tell the printer how to print the page.

printer file. An OS/400 object that provides printer output attributes.

proportionally spaced font. A font in which the characters have different character increments. Graphic character widths vary with the size of each graphic character. This allows for even spacing between printed characters and eliminates excess space around narrow characters, such as the letter i. Contrast with uniformly spaced font.

PSF. Print Services Facility

PSF configuration object. OS/400 object that lets you define configuration parameters that you cannot specify with the device description CRTDEVPR or CHGDEVPR commands.

R

raster font. (1) A font created by a series of pels (picture elements) arranged in scan lines to form an image. (2) A font in which the characters are defined directly by the raster bit map.

raster graphics. Computer graphics in which a display image is composed of an array of picture elements (pels) arranged in rows and columns. (I) (A) Contrast with vector graphics.

raster pattern. A series of picture elements (pels) in scan lines to form an image. See also page segment.

record. A collection of related data or words, treated as a unit; for example, in stock control, each invoice could constitute one record.

record-based file. See *database file*.

record format line data. A form of line data where each record is preceded by a 10-byte identifier. Contrast with *traditional line data*.

Redbook. IBM Redbooks typically provide positioning and value guidance, installation and implementation experiences, typical solution scenarios, and step-by-step "how-to" guidelines. They often include sample code and other support materials that are available as downloads, along with the Redbooks, from the IBM Redbooks Web Page: www.ibm.com/redbooks.

remote printer. A device that prints in a location away from the centralized data processing center.

resource. A collection of printing instructions and sometimes data to be printed consisting entirely of structured fields. A resource can be stored as a member of a library and can be called for by Print Services Facility when needed. Coded fonts, font character sets, code pages, page segments, overlays, form definitions, and page definitions are all resources.

resource library. (1) A collection of related files. (2) A place to store resources such as form definitions, page definitions, page segments, fonts, and overlays.

rotation. The number of degrees a character is rotated relative to the print direction. One of four directions that define the orientation of text relative to a sheet, page, overlay, text block, or page segment.

routing tag. A keyword or valid e-mail address (in the form *name@domain*) used with the PDF subsystem. If the routing tag is a keyword, it must be mapped to one or more valid e-mail addresses or other information by a user-defined PDF mapping program.

S

scanner. A device that examines a spatial pattern one part after another and generates analog or digital signals corresponding to the pattern. Scanners are often used in mark sensing, pattern recognition, or character recognition. (I) (A)

SCS data. SNA character string

simplex printing. Printing on only one side of the paper. Contrast with duplex printing.

SBCS. Single-byte character set

single-byte character set (SBCS). A character set that contains up to 256 characters. Single-byte character sets for languages with alphabetic writing systems, such as English, Greek, and Arabic.

SNA. Systems Network Architecture

SNA character string (SCS) data. In SNA, a data stream composed of EBCDIC controls, optionally intermixed with end-user data, that is carried within a request/response unit.

software. Programs, procedures, rules, and any associated documentation pertaining to the operation of a system. (T) (A) Contrast with hardware.

source format. The format of an AFP resource, other than fonts, used by AFP resource management programs. Contrast with object format.

spooled file. An OS/400 file that contains output for the printer.

spooling (simultaneous peripheral operation online). (1) The use of auxiliary storage as a buffer storage to reduce processing delays when transferring data between peripheral equipment and the processors of a computer. (I) (A) (2) The reading of input data streams and the writing of output data streams on auxiliary storage devices, concurrently with job execution, in a format convenient for later processing or output operations.

stream file. A randomly accessible sequence of bytes with no further structure imposed by the system. For example, a document, PC file, or UNIX file. Common uses are images, text, audio, and video files. The OS/400 object type is *STMF. Contrast with *database file*.

structured field. A self-identifying, variable-length record, which can have a content portion that provides control information, data, or both.

syntax. The rules and keywords that govern the use of a programming language.

system printer. The printer used for any printed output that is not specifically directed to another printer.

system programmer. A programmer who is responsible for writing programs for the functions of the computer operating system and who has a thorough knowledge of the operating system. The system programmer installs and maintains AFP software in the OS/400 environment.

Systems Network Architecture (SNA). In IBM networks, the description of the layered logical structure, formats, protocols, and operational sequences that are used for transmitting information units through networks, as well as controlling the configuration and operation of networks.

T

TCP/IP. Transmission Control Protocol/Internet Protocol

terminal. A device, usually equipped with a keyboard and some kind of display, capable of sending and receiving information over a communication channel.

text. A graphic representation of information about an output medium. Text consists of alphanumeric characters and symbols arranged in paragraphs, tables, columns, or other shapes.

text-formatting program. A program that determines the manner in which data will be placed on a page.

text orientation. A description of the appearance of text as a combination of inline and baseline directions and character rotation.

traditional line data. A form of line data that is prepared for printing on a line printer, such as 6262 or 3211. Contrast with *record format line data*.

Transmission Control Protocol/Internet Protocol (TCP/IP). A set of communications protocols that support peer-to-peer connectivity functions for both local and wide area networks.

troff. A phototypesetting utility originally designed to support a Graphics Systems phototypesetting machine but that is now capable of supporting a variety of phototypesetters. The utility produces data in a format called ditroff.

type family. A collection of fonts of a common typeface that vary in size and style.

type font. Type of a given size and style; for example, 10-point Sonoran Serif roman medium. (A)

typeface. A collection of fonts all having the same style, weight, and width; each font differs from the others by point size or type family.

typeset. (1) To arrange the type on a page for printing. (2) Pertaining to material that has been set in type.

typographic font. A typeface originally designed for typesetting systems. Contrast with mixed-pitch font, uniformly spaced font. Synonym for proportionally spaced font.

U

underscore. A line printed under a character. To underline.

uniformly spaced font. A font in which the characters have the same character increment. Synonymous with monospaced fonts. Contrast with proportionally spaced font and typographic font.

uppercase. Pertaining to capital letters, as distinguished from small letters; for example, A, B, G, rather than a, b, g.

V

variable data. (1) In programming languages, a language object that may take different values, one at a time. The values of a variable are usually restricted to a certain data type. (l) (2) A quantity that can assume any of a given set of values. (A) (3) Used to represent a data item whose value can be changed while the program is running. Contrast with constant data.

vector. In computer graphics, a directed line segment.

vector graphics. Computer graphics in which display images are generated from display commands and coordinate data. (l) (A) Contrast with raster graphics.

W

word processing. The entry, modification, formatting, display, and printing of text on personal computers, microprocessors, and stand-alone word processors.

Bibliography

This bibliography lists the titles of publications containing additional information about the OS/400 operating system, Advanced Function Presentation, and related products.

The titles and order numbers may change from time to time. To verify the current title or order number, consult your IBM marketing representative.

You can obtain many of the publications listed in this bibliography from the Printing Systems Digital Library: <http://www.ibm.com/printers/r5psc.nsf/web/manuals> or the Online Publications Website: <http://publib.boulder.ibm.com/>. Some of the unlisted publications, such as the API references have been integrated into the iSeries Infocenter: <http://www.as400.ibm.com/infocenter>. You can obtain Redbooks from the Redbooks Web Page: <http://www.ibm.com/redbooks/>.

Advanced Function Presentation

Publication	Order Number
<i>Advanced Function Presentation: Printer Summary</i>	G544-3135
<i>Advanced Function Presentation: Printer Information</i>	G544-3290
<i>Advanced Function Presentation: Programming Guide and Line Data Reference</i>	S544-3884
<i>AFP Conversion and Indexing Facility: User's Guide</i>	G544-5285
<i>Guide to Advanced Function Presentation</i>	G544-3876
<i>IBM Online Library: Printing and Publishing Collection</i>	SK2T-2921

Business Graphics Utility

Publication	Order Number
<i>BGU User's Guide and Reference</i>	SC09-1408-00

Data Stream and Object Architectures

Publication	Order Number
<i>Data Stream and Object Architectures: MO:DCA Reference</i>	SC31-6802
<i>Intelligent Printer Data Stream Reference</i>	S544-3417
<i>Data Stream and Object Architectures: Bar Code Object Content Architecture (BCOCA) Reference</i>	S544-3766
<i>Data Stream and Object Architectures: Font Object Content Architecture (FOCA) Reference</i>	S544-3285
<i>Presentation Text Object Content Architecture Reference</i>	SC31-6803
<i>Graphics Object Content Architecture Reference</i>	SC31-6804
<i>Data Stream and Object Architectures: Image Object Content Architecture (IOCA) Reference</i>	SC31-6805

Facsimile Support for iSeries

Publication	Order Number
<i>Facsimile Support for AS/400 Programmer's Guide and Reference</i>	SC41-0656-01
<i>Facsimile Support for AS/400 Installation Guide</i>	SC41-0654-01
<i>Facsimile Support/400 User's Guide</i>	SC41-0655-01

GDDM

Publication	Order Number
<i>GDDM Programming Guide</i>	SC41-0536-00
<i>GDDM Reference</i>	SC41-3718-00

Fonts

Publication	Order Number
<i>IBM Infoprint Fonts: Introduction to Type Transformer and Utilities for Windows</i>	G544-5853-00
<i>Infoprint Fonts: Creating User-Defined Characters</i>	G544-5854-00
<i>IBM Infoprint Fonts: Font Summary</i>	S544-5846-00
<i>IBM AFP Fonts: Font Summary for AFP Font Collection</i>	S544-5633
<i>IBM AFP Fonts: Technical Reference for Expanded Core Fonts</i>	S544-5228
<i>AFP Font Collection: Type Transformer for Windows User's Guide</i>	G544-5726
<i>AFP Fonts: Technical Reference for Code Pages</i>	S544-3802
<i>Using OpenType Fonts in an AFP System</i>	G544-5876

IBM AFP PrintSuite for iSeries

Publication	Order Number
<i>Advanced Print Utility for iSeries: User's Guide</i>	S544-5351-03
<i>Page Printer Formatting Aid: User's Guide</i>	S544-5284-07
<i>AFP Toolbox for Multiple Operating Systems: User's Guide</i>	S544-5292-04
<i>iSeries: Printing with SAP R/3</i>	S544-5412-02

IBM Content Manager OnDemand for iSeries

Publication	Order Number
<i>IBM Content Manager OnDemand for iSeries Common Server Administration Guide</i>	SC27-1161-02

Infoprint Designer for iSeries

Publication	Order Number
<i>Infoprint Designer for iSeries: Getting Started</i>	G544-5773-03

Infoprint Server for iSeries

Publication	Order Number
<i>Infoprint Server for iSeries: Introduction and Planning Guide</i>	G544-5774-02
<i>Infoprint Server for iSeries: User's Guide</i>	G544-5775-03

OS/400

Publication	Order Number
<i>Printer Device Programming</i>	SC41-5713-06
<i>AFP Utilities for iSeries User's Guide</i>	S544-5349-02

Redbooks

Publication	Order Number
<i>How to Replace OfficeVision/400 in Your Applications: Looking at Domino for AS/400 and AS/400 Alternatives</i>	SG24-5406
<i>IBM AS/400 Printing II</i>	GG24-3704
<i>IBM AS/400 Printing III</i>	GG24-4028
<i>IBM AS/400 Printing IV</i>	GG24-4389
<i>IBM iSeries Printing V</i>	SG24-2160
<i>IBM @server iSeries Printing VI: Delivering the Output of e-business</i>	SG24-6250

Index

Special characters

47

Numerics

4-up printing 29

A

accessing the PDF subsystem 255

ACIF 258

Actions, Monitor

 action for input spooled file 179

 action for output spooled file 179

 how they are processed 179

 selection for input spooled file 179

Adobe fonts 45

Advanced Function Presentation

 See AFP

Advanced Print Utility

 See APU

AFP

 advantages of 155

 applications 19

 bar codes in 14

 case study 113

 compatibility fonts 60

 complimentary applications 21

 description of 11

 elements 12

 formatting resources 72

 graphics in 13

 images in 13

 indexing 217

 introduced 6

 output, understanding 12

 overlays in 14

 page architecture 5

 positioning document elements in 136

 print flow 15

 printing 7

 printing subsystem 14

 text in 13

 viewing 233

AFP Compatibility Fonts 60

AFP Data Stream

 See AFPDS

AFP font architecture 45

AFP Font Collection

 AFP Code Page Editor utility 59

 AFP Coded Font Editor utility 59

 AFP Font to Flat File utility 59

 AFP2FON utility 59

 ATMSHOW utility 59

 CID2EPS utility 59

 DBCS Core Fonts 58

 DBCS Simulation Fonts 58

AFP Font Collection (*continued*)

 DBCS Type Transformer 59

 described 20, 57

 DUVRMARK utility 59

 Expanded Core Fonts 57

 FontLab Composer tool 59

 FontLab tool 59

 installing 49, 84

 installing font libraries 84

 installing fonts 82

 libraries 49, 84

 OS/400 Uploader utility 59

 printing samples 82

 restoring font libraries 84

 SBCS Type Transformer 59

 ScanFont tool 59

 SigMaker tool 59

 tools 82

 Type Transformer 59

 Type Transformer and Utilities for Windows 59

 TypeTool 59

 Which libraries should I install? 49

AFP Font Collection Tools 82

AFP fonts 45

AFP Manager

 capturing fonts with 351

 changing a PSF configuration object with 272

 converting a spooled file to PDF with 269

 creating a PSF configuration object with 271

 creating font mapping tables with 350

 description of 262

 e-mailing a spooled file with 269

 installing 268

 specifying the user resource library list 359

 storing PC resources on the OS/400 with 273

 using 268, 271

 using to upload an AFP image 92

 viewing font mapping tables with 349

AFP printer driver

 creating overlays with 91

 creating page segments with 91

 described 21

AFP PrintSuite 153

 AFP Toolbox 19

 APU 19

 described 19

 PPFA 19

 SAP R/3 Print 19

AFP Toolbox

 case study 219

 COBOL program example 331

 described 19

 document structure 218

 formatting with 218

 page structure 218

 program flow 218

 RPG program example 343

 source for case study 331

- AFP Toolbox (*continued*)
 - using 217
 - what can I do with 217
- AFP Utilities
 - defining a printout format definition 224
 - described 20
 - Overlay Utility 20
 - Resource Management Utility 20
 - specifying bar codes with 107
- AFP Viewer 237
 - See also *wbview*
 - applications that use 233
 - described 22
- AFP Viewer Plug-In 239
- AFP Workbench Viewer
 - calling from your application 239
 - described 22
 - using PC-resident fonts to display documents 235
- AFPDS
 - architecture 11
 - compared to IPDS 6
 - introduced 6
- AFPU/400
 - See AFP Utilities
- all points addressability 11
- anamorphic font character scaling 84
- APU
 - case study 159
 - copy 156
 - data mapping 155
 - data structure 155
 - described 19
 - design phase 154
 - example 157
 - expanded case study with 171
 - field selection 155
 - page format 156
 - print definition 156
 - print flow 153
 - printing with 175
 - printing with APU Monitor 178
 - production phase 154
 - specifying bar codes with 108
 - step-by-step 157
 - typical application 153
 - using 153
 - what you can do with 154
 - why use 155
- APU Monitor
 - example of processing 179
 - how the monitor works 179
 - how used by APU 175
 - implementing customer requirements 180
 - output queue after monitor processing 181
 - printing with 178
 - processing sequence 179
- APYPRTDEF 178
- archiving output with OnDemand 279
- ASCII
 - printer drivers 8
 - printing 7

- ASCII (*continued*)
 - printing on OS/400 6
- aspect ratio 68
- associating a print definition with a spooled file 175

B

- bar code density 68
- bar code object content architecture
 - See BCOCA
- bar codes
 - basics 69
 - choosing a density for 102
 - choosing a height for 103
 - choosing a symbology 102
 - choosing an orientation for 103
 - description of 14, 67
 - elements 69
 - examples 70
 - on sample page 67
 - printing on IPDS printers 101
 - specifying in APU 108
 - specifying in Infoprint Designer 105
 - specifying in PFU 107
 - specifying with DDS 104
 - specifying with PPFA 103
 - symbologies 70
 - terminology 68
 - using on the OS/400 101
 - which to use? 102
 - working with 101
- BARCODE keyword 126
- BCOCA
 - description of 11
 - sub-architecture 6
- bearer bars 68
- BGU
 - described 21
- bibliography 375
- bilevel 63
- bitmap file format 64
- BOX keyword 126
- BRMS for iSeries
 - described 23
- business partner applications 23

C

- CA/400
 - See iSeries Access
- calling the AFP Viewer from your application 239
- captured fonts 53
- capturing fonts 83, 351
- case study
 - adding an overlay with a printer file 121
 - DDS source for initial output 117
 - DDS specifications 141
 - dynamic invoicing document 305
 - enhanced example 297
 - enhancing 148
 - floating invoicing document 305

- case study (*continued*)
 - initial application program 114
 - initial processing flow 117
 - introduction 113
 - preprinted invoice 113
 - related example with PFU 224
 - source input data for 191, 328
 - with AFP Toolbox 219
 - with APU 159
 - with APU, expanded case study 171
 - with CM OnDemand 279
 - with DDS 137
 - with FAX 241
 - with Infoprint Designer 201
 - with PPFA 191
 - with printer files 121
- CDEFNT keyword 126
- CGM file format 64
- changes to this book xix
- changing a PSF configuration object 272
- character ID 42
- character set 45
- check character 68
- check digit 68
- CHGPSFCFG CL command 39
- CHGSPLFA CL command 26
- choose an AFP font 52
- choosing the proper tool 119
- CHRSIZ keyword 126
- CL commands
 - APYPRTDEF 175, 176, 177, 178
 - CHGPSFCFG 39
 - CHGSPLFA 26
 - CRTAFPDTA 257
 - CRTPSFCFG 39
 - OVRPRTF 25
 - WRKPSFCFG 39
- clip 63
- CM OnDemand
 - defining the invoices to 279
 - described 23
 - indexing with 281
 - overview of 279
 - retrieving archived files 282
 - Super Sun Seeds 279
 - triggers 280
 - view-related applications 233
- code page 46
- code pages
 - description 48
 - example of different 47
- code point 46
- coded font 46
- COLOR keyword 126
- compile an AFP image 92
- complementary applications
 - defined 19
 - list of 21
- compression 63
- computer output reduction
 - description of 30

- Content Manager ImagePlus 23
- continuous code 68
- converting a spooled file to PDF 264, 269
- converting line data to AFP 34
- copy spooled file API 248, 255
- copying a stream file from OS/400 to your workstation 237
- COR, specifying in a printer file 30
- corner stapling 36
- Create AFP Data command 257
- creating a page segment 90, 93
- creating a printer share 274
- creating a PSF configuration object 271
- creating an overlay 90, 93
- creating font mapping tables 350
- creating PDF 255
- crop 63
- CRTAFPDTA 257
- CRTPSFCFG CL command 39
- CVTLINDTA 34

D

- Data Description Specifications
 - See DDS
- data object resources 109
- Data/Text Merge for AS/400e Servers 347
- DBCS Core Fonts 58
- DBCS fonts 45
- DBCS Simulation Fonts 58
- DDS
 - additional examples 297
 - disadvantages to using 7
 - example 129
 - finishing keywords 126
 - floating document elements 135
 - for case study overlays 141
 - for initial Super Sun Seeds application 117
 - functions 128
 - how used in printer file 125
 - introduction 6
 - keywords 126
 - page composition keywords 126
 - page layout keywords 126
 - positioning document elements with 136
 - specifying a constant back 149
 - specifying bar codes 104
 - specifying copies 149
 - Super Sun Seeds 137
 - using 125
- DDS keywords 126
 - BARCODE 126
 - BOX 126
 - CDEFNT 126
 - CHRSIZ 126
 - COLOR 126
 - DOCIDXTAG 126
 - DRAWER 126
 - DTASTMCMD 126
 - DUPLEX 126
 - ENDPAGE 126

DDS keywords *(continued)*

- ENDPAGGRP 126
- FNTCHRSET 126
- FONT 126
- FORCE 126
- GDF 126
- HIGHLIGHT 126
- INVMMAP 126
- LINE 126
- list of 126
- LPI 126
- OUTBIN 126
- OVERLAY 126
- PAGRRT 126
- PAGSEG 126
- POSITION 126
- STAPLE 126
- STRPAGGRP 126
- TXTRTT 126
- UNDERLINE 126
- ZFOLD 126

deciding on an output strategy 287

deciding what tools you need 287

density 68

discrete code 68

distributing PDF output 256

DOCIDXTAG keyword 126

downloadable fonts 60

dpi (dots per inch) 42

DRAWER keyword 126

DTASTMCMD keyword 126

DTM/400 347

DUPLEX keyword 126

E

e-business output 8

e-mail

- PDF output 256

e-mail a spooled file 269

e-mailing output 254

e-mailing PDF 256

e-output 9

edge stitch stapling 36

electronic forms

- See overlays

enabling applications

- defined 19
- list of 19

ENDPAGE keyword 126

ENDPAGGRP keyword 126

EPS file format 64

EverGreen/400 347

examples

- fonts 80

Expanded Core Fonts 57

F

Facsimile Support for iSeries

- described 23

Facsimile Support for iSeries *(continued)*

- description of 241
- using 241
- using with the case study 241

fast path

- for simple applications 155

FAX/400

- See Facsimile Support for iSeries

FGID 48

- how used 44
- used to access fonts 44

fields

- for user exits 177

file format 63

finding fonts 50

finding OS/400 objects 353

finishing 36

- nested stapling 126
- punch 36
- stapling 36
- support in form definition 75
- with DDS 126
- z-fold 36

first read rate 68

fixed-pitch fonts 42

FNTCHRSET keyword 126

FOCA

- description of 12
- sub-architecture 6

font capturing 351

Font Global Identifier

- See FGID

FONT keyword 126

font mapping tables

- working with 82

font object content architecture

- See FOCA

font products 57

font substitution

- controlling 55
- messages 55

font terminology 42

fonts

- Adobe 45
- AFP 45
- AFP Font Collection 57
- AFP MICR Fonts 61
- AFP, chosen by PSF 52
- architecture 45
- as external resource 235
- captured 53
- capturing 83
- character set 45
- code page 46
- coded font 46
- compatibility 60
- converting Adobe to AFP 59
- DBCS 45
- DBCS Core 58
- DBCS Simulation 58
- defined 41

- fonts (*continued*)
 - description of 13
 - different types on sample page 41
 - downloadable 60
 - dpi (dots per inch) 42
 - example of using 79
 - example using 80
 - Expanded Core 57
 - family 42
 - FGID 48
 - finding 50, 52
 - fixed-pitch 42
 - font character set naming convention 46
 - GDDM 60
 - getting 56
 - host-resident 44
 - ID 48
 - in overlays 96
 - Infoprint 57
 - installing 49, 82, 84
 - libraries 49, 84, 293
 - location of 49
 - mapping 82
 - mapping schemes 45
 - mapping tables 53, 55
 - MICR 61
 - monospaced 42
 - naming conventions 50
 - on international systems 48
 - orientation 42, 44
 - outline 43, 44
 - pel (picture element) 42
 - point size 43
 - posture 42
 - print a catalog of 50
 - printer-resident 44, 48
 - products 57
 - proportionally spaced 43
 - raster 43, 60
 - resident 42
 - resolution 42
 - SBCS 45
 - scalable 44
 - scaling 84
 - size 42
 - spacing 42
 - structure 43
 - substitute outline for raster 54
 - substitution 53, 55
 - terminology 42
 - typeface 42
 - typographic 43
 - uniformly spaced 42
 - using 79
 - weight 42
 - where stored 49
 - where to get 56
 - Which libraries should I install? 49
- FORCE keyword 126
- form definitions
 - creating using the PPFA compiler 187

- form definitions (*continued*)
 - creating with Infoprint Designer 201
 - creating with PPFA 187
 - described 72
 - description of 74
 - example output 190
 - functions 75
 - how to get 77
 - how works 74
 - in print flow 72
 - printer file keyword support 76
 - source code 317, 323
 - sources of 77
 - using 76, 189
- formdef
 - See form definitions

G

- GDDM
 - described 21
- GDDM fonts 60
- GDF keyword 126
- generating PDF 40
- GIF file format 64
- GIF to AFP transform 259
- GOCA
 - description of 11
 - sub-architecture 6
- graphics
 - description of 13, 61, 62
 - different from image 62
 - how created 62
 - in AFP output 62
 - in OS/400 applications 61
 - terminology 63
- graphics object content architecture
 - See GOCA
- grid overlay 137

H

- HIGHLIGHT keyword 126
- Host Print Transform 7
- host-resident fonts 44
- how PSF searches for object containers 358
- how PSF searches for resources 357
- HRI 68

I

- IBM Toolbox for Java
 - described 22
- IM1 file format 64
- Image Editor 199
 - touching up an image 87
- image object content architecture
 - See IOCA
- image transforms, using 91
- images
 - building 87

- images *(continued)*
 - compiling 92
 - compression 65
 - converting to IOCA 90
 - description of 13, 61, 62
 - different from graphic 62
 - file formats 64
 - in AFP output 62
 - in OS/400 applications 61
 - in overlays 96
 - OS/400 formats 64
 - scanning 87
 - terminology 63
 - touching up 87
 - uploading 92
 - using 93
 - working with 87
- IMDS file format 64
- indexing 257, 281
 - AFP output 257
 - with Infoprint Server 257
 - with OnDemand 279
- Infoprint Designer
 - creating a page segment with 90
 - described 21
 - description of 197
 - example 201
 - Image Editor 197, 199
 - Layout Editor 197, 200
 - Overlay Editor 197, 199
 - specifying bar codes with 105
 - Super Sun Seeds 201
 - uploading a page segment 92
 - uploading an overlay 92
 - using 197
 - using for layout 201
 - what you can do with 198
 - working with images 90
- Infoprint Fonts 57
 - described 20
- Infoprint Server
 - accessing the PDF subsystem 255
 - consolidating enterprise printing 250
 - Create AFP Data command 257
 - creating electronic documents interactively 252
 - CRTAFPDTA 257
 - delivering documents over Intranet 251
 - described 21
 - distributing output from 256
 - e-business customer documents 248
 - e-mailing with 254, 256
 - electronic report distribution 247
 - how can I use 245
 - image transforms 259
 - image transforms, using 91
 - indexing with 257
 - integrating Web images 253
 - PDF subsystem 254
 - printing ASCII data on IPDS printers 259
 - publish output for electronic access 246
 - scenarios 245
- Infoprint Server *(continued)*
 - segmenting customer documents 249
 - working with 245
- Information Center xvii
- insert sheet 36
- insert Z-fold sheet 36
- installing a printer share 276
- installing AFP Manager 268
- installing fonts 82
- Integrated File System
 - finding an object in 354
 - introduced 353
 - Network File System 353
 - QDLS 353
 - QFileSvr.400 353
 - QNetWare 353
 - QNTC 353
 - QOpenSys 354
 - QOPT 354
 - QSYS.LIB 354
 - root File System 354
 - UDFS 354
 - using paths in 354
- Intelligent Printer Data Stream
 - See IPDS
- intelligent routing
 - defined 254
 - examples of using 246
 - specifying 256
- intercharacter gap 68
- introduction to AFP resources 41
- introduction to printer files 25
- INVM MAP keyword 126
- IOCA
 - description of 11
 - sub-architecture 6
- IOCA file format 64
- IOCA function sets 64
- IPDS
 - compared to AFPDS 6
 - described 12
 - printing 7
- iSeries Access
 - creating a printer share with 274
 - described 22
 - description of 261
 - for Web 263
 - iSeries Navigator component 261
 - using 261
 - using to view AFP documents 233
 - viewing spooled files with 235
 - viewing stream files with 237
- iSeries Access for Web
 - converting a spooled file to PDF 264
 - print capabilities 263
 - viewing spooled files with 238
- iSeries Information Center xvii
- iSeries Navigator
 - capturing fonts with 351
 - changing a PSF configuration object with 272
 - converting a spooled file to PDF with 269

iSeries Navigator (*continued*)
 copying a stream file from OS/400 to your workstation 237
 creating a printer share with 274
 creating a PSF configuration object with 271
 creating font mapping tables with 350
 described 22
 description of 262
 e-mailing a spooled file with 269
 for Web, described 22
 installing a printer share 276
 specifying the user resource library list 359
 storing PC resources on the OS/400 with 273
 using 268
 viewing font mapping tables with 349
iSeries output architecture 4

J

Java 22
JPEG file format 64
JPEG to AFP transform 259

L

ladder orientation 68
Layout Editor 200
libraries 357
library list 52
 device resource, specifying 359
 how used 357
 introduced 357
 specifying 359
LINE keyword 126
location of fonts 49
Lotus Domino 347
LPI keyword 126

M

map object 254
mapping schemes on the iSeries 45
medium map 126
MICR fonts 61
migrating from OV/400 347
MO:DCA-P
 data stream 12
 introduced 6
 IOB 12
 platforms 11
 sub-architectures 6
monospaced fonts 42
moving data between file systems 354
multi-up, specifying in a printer file 29

N

names, fonts 50
Network File System 353

O

object container 12
object containers
 described 78
 how OS/400 searches for 109
 how PSF searches for 358
 working with 109
OfficeVision/400 347
 migrating from 347
 not supported 347
 replacement products 347
OnDemand
 See CM OnDemand
orientation, font 44
OS/400
 e-business output 8
 output architecture 4
 output from 3
 page architecture 5
 print flow 4
 print support 7
 printer writer 8
OS/400 output
 archiving 279
 e-mailing 256
 faxing 241
 indexing 257, 279
 retrieving archived 279
 transforming to PDF 254
 viewing 233
OUTBIN keyword 126
outline fonts 43, 44
OV/400 347
 See OfficeVision/400
Overlay Editor 199
OVERLAY keyword 126
Overlay Utility
 described 20
overlays
 adding with printer file 121
 as external resource 235
 creating 90, 93
 creating from existing images 97
 description of 14, 65
 designing with Infoprint Designer 97
 designing 95
 designing with DDS 97
 fonts in 96
 grid 137
 images in 96
 object size 96
 on sample page 65
 options for building 66
 performance considerations 95
 reasons for using 66
 specifying constant back 149
 specifying in a printer file 32
 sub form 95
 using on the OS/400 99
 working with 95
OVRPRTF CL command 25

P

- page architecture 5
- page definitions
 - creating with Infoprint Designer 201
 - creating with PPFA 184
 - described 72
 - description of 72
 - functions 73
 - how to get 77
 - how works 72
 - in print flow 72
 - printer file keyword support 76
 - source code 317, 323
 - sources of 77
 - using 76, 189
- Page Printer Formatting Aid
 - See PPFA
- page segments
 - as external resource 235
 - creating 90, 93
 - described 63
 - description of 61
 - in AFP output 62
 - terminology 63
- pagedefs
 - See *also* page definitions
 - associating with a spooled file 175
 - example, single page format 157
 - using with apply command 178
- PAGRRT keyword 126
- PAGSEG keyword 126
- parameters
 - APYPRTDEF 175
- PCX file format 64
- PDF
 - distributing 256
 - e-mailing 256
 - generating multiple 256
 - intelligent routing 256
 - mapping program 256
 - storing as stream file 256
- PDF mapping program 254
- pel 63
- pel (picture element) 42
- picket fence orientation 68
- pixel 63
- point size 43
- POSITION keyword 126
- PPFA 183
 - compiling form definitions 183, 187
 - compiling page definitions 183, 184
 - described 19
 - page definition source code 187
 - record formatting 194
 - sample form definition source 184
 - sample output 184
 - sample page definition source 184
 - specifying bar codes 103
 - Super Sun Seeds 191
 - using 183
 - using for layout 183
- presentation text object content architecture
 - See PTOCA
- print flow 4
- Print Format Utility
 - case study related example 224
 - described 20
 - introduction to 223
 - summary 230
 - Super Sun Seeds related example 224
 - using 223
- Print Services Facility for OS/400
 - See PSF
- printer files
 - CL commands 25
 - converting line data to AFP 34
 - described 25
 - enhancing your output with 27
 - flow 25
 - introduction 6
 - keyword support considerations 76
 - limitations of 123
 - parameters 26
 - specifying finishing 36
 - specifying font changes 27
 - specifying line changes 27
 - specifying multi-up 29
 - specifying overlays 32
 - using 121
 - using DDS with 125
- printer share 274
- printer writers 7
- printer-resident fonts 44, 48
- printing 4-up 29
- printing ASCII data on IPDS printers 259
- PrintSuite, AFP 153
- products, font 57
- products, OS/400 287
- proportionally spaced fonts 43
- PSF
 - described 8
 - description of 14
 - list of data streams supported 14
 - print flow 15
 - printers 16
 - role of 7
 - when required 16
- PSF configuration objects
 - changing 39, 272
 - creating 39
 - introduction to 39
 - using 40
 - using to generate PDF 40
- PSF search for object containers 358
- PSF searches for resources 357
- PTF numbers
 - for finishing 36
- PTOCA
 - description of 12
- punching 36

Q

- qchrid system value 48
- QDLS 353
- QFileSvr.400 353
- QNetWare 353
- QNTC 353
- QOpenSys 354
- QOPT 354
- QSYS.LIB 354
- quiet zone 68

R

- R/DARS
 - See CM OnDemand
- raster 63
- raster fonts 43, 60
- record format line data
 - creating variable-length graphics 196
 - defining color with 195
 - drawing graphics with 195
 - using with PPFA 194
 - with PPFA 194
- reengineering business documents
 - introduction 3
- related applications 19
- replacing outline fonts with rasters 54
- resolution 63
- resource library lists 359
- Resource Management Utility
 - described 20
- resources
 - bar codes 67
 - data object 109
 - described 5, 7
 - external 234
 - fonts 41
 - form definitions 72
 - formatting 7
 - how PSF searches for 357
 - introduction to 41
 - object containers 78
 - overlays 65
 - page definitions 72
 - page segments 62
 - secondary 78
- respool
 - PDF output 256
- root File System 354
- RPG
 - for case study using DDS 144
 - for DDS example 134

S

- saddle stitch 36
- SAP R/3 Print
 - described 19
- SBCS fonts 45
- scalable fonts 44

- scale 63
- scanner 63
- SCS
 - as output of iSeries 154
 - file to be transformed by APU 157
 - file to be transformed by Infoprint Designer 202
 - printing 7
 - remapping fields in 154
 - using interactively 155
 - what APU works on 154
 - what it is 154
- secondary resources 78
- sending output by e-mail 254
- sending output by fax 241
- Simulation Fonts, DBCS 58
- SNA Character String
 - See SCS
- specifying bar codes with APU 108
- specifying bar codes with DDS 104
- specifying bar codes with Infoprint Designer 105
- specifying bar codes with PFU 107
- specifying bar codes with PPFA 103
- specifying resource library lists 359
- spooled files
 - associating a print definition with 175
 - changing attributes of 26, 123
 - converting to PDF 264, 269
 - e-mailing 269
 - faxing 23
 - finding resources 359
 - resources referenced by 16
 - transforms performed on 16
 - viewing 22, 23, 233, 235, 238
- STAPLE keyword 126
- stapling 36
- stop pattern 68
- storing PC resources on the OS/400 273
- stream filea
 - PDF 256
- STRPAGGRP keyword 126
- structure, font 43
- sub form 95
- substituted fonts 53
- substituting outline fonts for rasters 54
- summary 287
- summary of changes xix
- Super Sun Seeds Case Study
 - adding an overlay with a printer file 121
 - DDS source for initial output 117
 - DDS specifications 141
 - dynamic invoicing document 305
 - enhanced example 297
 - enhancing 148
 - floating invoicing document 305
 - initial application program 114
 - initial processing flow 117
 - introduction 113
 - preprinted invoice 113
 - related example with PFU 224
 - source input data for 191, 328
 - with AFP Toolbox 219

Super Sun Seeds Case Study (*continued*)
 with APU 159
 with APU, expanded case study 171
 with CM OnDemand 279
 with DDS 137
 with FAX 241
 with Infoprint Designer 201
 with PPFA 191
 with printer files 121
symbology 68

T

third party design tools 23
TIFF file format 64
TIFF to AFP transform 259
Toolbox, AFP
 See AFP Toolbox
transforming
 images to AFP 259
 spooled file to PDF 254, 264, 269
triggers 280
TXTRTT keyword 126
Type Transformer and Utilities for Windows 59
typeface 42
typographic fonts 43

U

UDFS 354
UNDERLINE keyword 126
uniformly spaced fonts 42
uploading
 AFP images 92
 overlays 93
 page segments 93
USERASCII data type 6
using
 fonts 79
 PPFA for layout 183
using AFP Manager
 to upload and compile an AFP image 92
using form definitions 189
using images 93
using overlays 99
using page definitions 189
using PDF subsystem 40
Utilities, Advanced Function Printing
 See AFP Utilities
Utility
 Overlay 20
 Print Format 20
 Resource Management 20

V

viewing files 233
 AFP files 239
 benefits 233
 spooled files 238
 using AFP Viewer 233

viewing files (*continued*)
 with iSeries Access 237
viewing font mapping tables 349

W

WAF/400
 described 23
 what product should I use? 287
wide-to-narrow ratio 68
working with AFP output 231
working with font mapping tables 82
wrappers 63
WRKPSFCFG CL command 39

X

x-dimension 68

Z

Z-fold insertion 36
ZFOLD keyword 126

Readers' Comments — We'd Like to Hear from You

iSeries
Guide to Output

Publication No. S544-5319-05

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Complete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to find	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to understand	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Well organized	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Applicable to your tasks	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? Yes No

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

Name

Address

Company or Organization

Phone No.



Fold and Tape

Please do not staple

Fold and Tape



NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Corporation
Information Development
IBM Printing Systems Company
Department H7FE Building 004M
PO Box 1900
Boulder, CO 80301-9817



Fold and Tape

Please do not staple

Fold and Tape



Printed in USA

S544-5319-05



Spine information:



iSeries

iSeries Guide to Output