

Internet-Draft

Roger deBry
IBM Corporation

T. Hastings
Xerox Corporation
R. Herriot
Sun Microsystems
Scott Isaacson
Novell, Inc.
November 18, 1996

Internet Printing Protocol - IPP/1.0
Version 0.92, November 18, 1996

NOTE: This document still has many rough spots which will need further editing. At this time, the reader should read it for major concepts.

Status of this Memo

This document is an Internet -Draft. Internet -Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet -Drafts.

Internet -Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet -Drafts as reference material or to cite them other than as ``work in progress.''

To learn the current status of any Internet -Draft, please check the ``lid-abstracts.txt'' listing contained in the Internet -Drafts Shadow Directories on ftp.is.co.za (Africa), nic.nordu.net (Europe), munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or ftp.isi.edu (US West Coast).

Abstract

This Internet-Draft specifies an Internet Printing Protocol (IPP). This protocol is heavily influence by the semantic operations and attributes defined in ISO/IEC 10175 Document Printing Application (DPA) parts 1 and 3. It also incorporates some of the implementation and interoperability lessons learned from other printing related standards such as POSIX System Administration - Part 4 (POSIX 1378.4) and X/Open A Printing System Interoperability Specification(PSIS).

IPP is defined as a set of abstract data types and operations. The operations are implemented using a simple request and response

51 mechanism built on top of HTTP. The abstract data types are
 52 encoded as simple ASCII text strings.

53 The IPP protocol initially covers only end user operations on
 54 basic print service objects. Future versions of the protocol will
 55 cover operator and administrator operations. Authentication is
 56 realized by mechanisms outside the scope of the protocol, but the
 57 protocol does introduce some access control functionality so that
 58 only authorized end users are allowed to submit print jobs to
 59 devices whose implementation and site policy support access
 60 control. Also, the Cancel Job operation requires some
 61 authentication so that jobs can only be canceled by the end user
 62 who submitted the job. Extended monitoring and management is
 63 possible through other protocols such as the SNMP Printer MIB
 64 [1].. In the areas where there are no existing standards, some
 65 proposed and emerging standards are being worked (management,
 66 security, etc.). As these services become more stable, this
 67 document (and hence the protocol) can be updated to reflect the
 68 integration and relationships with these other standards.

69 Table of Contents

| | | |
|-----|--|----|
| 70 | 1. Introduction | 5 |
| 71 | 2. Distributed Printing | 5 |
| 72 | 2.1 Generic Print System Components | 6 |
| 73 | 2.2 IPP Components | 6 |
| 74 | 3. IPP Objects | 7 |
| 75 | 3.1 Printer | 7 |
| 76 | 3.2 Job | 9 |
| 77 | 3.3 Job Template | 9 |
| 78 | 3.4 Object Relationships | 9 |
| 79 | 3.5 Object Identity | 9 |
| 80 | 4. Naming | 10 |
| 81 | 4.1 Directory Services | 10 |
| 82 | 4.2 Directory Entry Schema | 11 |
| 83 | 4.2.1 Name | 11 |
| 84 | 4.2.2 Description | 11 |
| 85 | 4.2.3 Location | 11 |
| 86 | 4.2.4 Print Quality | 11 |
| 87 | 4.2.5 Cost | 11 |
| 88 | 4.2.6 This indicates a somewhat subjective evaluation of the | |
| 89 | overall cost of printing at this printer: "high", "medium", or | |
| 90 | "low".Resolution | 11 |
| 91 | 4.2.7 Color Supported | 12 |
| 92 | 4.2.8 Fonts Supported | 12 |
| 93 | 4.2.9 Maximum Speed | 12 |
| 94 | 4.2.10 Device Id | 12 |
| 95 | 4.2.11 Model | 12 |
| 96 | 4.2.12 Manufacturer | 12 |
| 97 | 4.2.13 Type | 12 |
| 98 | 4.2.14 Document Formats s Supported | 13 |
| 99 | 4.2.15 Sides Supported | 13 |
| 100 | 4.2.16 Finishings Supported | 13 |
| 101 | 5. IPP Operations | 13 |
| 102 | 5.1 IPP Operations Using HTTP | 13 |
| 103 | 5.1.1 HTTP Overview | 13 |
| 104 | 5.1.2 IPP Operation Encoding | 14 |
| 105 | 5.1.2.1 HTTP Request-Header Fields | 15 |
| 106 | 5.1.2.2 Print | 15 |

| | | |
|-----|--|----|
| 107 | 5.1.2.3 Cancel Job | 15 |
| 108 | 5.1.2.4 Get Attributes | 16 |
| 109 | 5.1.2.5 Get Jobs | 16 |
| 110 | 5.1.3 The Print-Job | 16 |
| 111 | 5.1.3.1 Print Job Object Header | 16 |
| 112 | 5.1.3.2 Document Header | 16 |
| 113 | 5.1.3.3 Document-Content Header | 17 |
| 114 | 5.1.3.4 Job Attributes | 17 |
| 115 | 5.1.3.5 Document Attributes | 17 |
| 116 | 5.2 Print Operation | 17 |
| 117 | 5.2.1 Print Request | 18 |
| 118 | 5.2.2 Print Response | 18 |
| 119 | 5.3 Cancel Job Operation | 18 |
| 120 | 5.3.1 Cancel Job Request | 18 |
| 121 | 5.3.2 Cancel Job Response | 19 |
| 122 | 5.4 Get Attributes Operations | 19 |
| 123 | 5.4.1 Get Attributes Request | 19 |
| 124 | 5.4.2 Get Attributes Response | 19 |
| 125 | 5.5 Get Jobs Operation | 19 |
| 126 | 5.5.1 Get Jobs Request | 19 |
| 127 | 5.5.2 Get Jobs Response | 19 |
| 128 | 6. Object Attributes | 20 |
| 129 | 6.1 Attribute Syntaxes | 20 |
| 130 | 6.2 Job Attributes | 21 |
| 131 | 6.2.1 Job Informational Attributes (Set by a Client/End User) .. | 21 |
| 132 | 6.2.1.1 job-name (string) | 21 |
| 133 | 6.2.2 Job Informational Attributes (Set by a Printer) | 21 |
| 134 | 6.2.2.1 job-identifier (url) | 21 |
| 135 | 6.2.2.2 job-originator (name) | 22 |
| 136 | 6.2.2.3 job-originating-host (name) | 22 |
| 137 | 6.2.2.4 notification-address (name) | 22 |
| 138 | 6.2.2.5 job-locale (type3Locale) | 22 |
| 139 | 6.2.3 Printer Selection Attributes (Set by Client/End User) | 22 |
| 140 | 6.2.3.1 printer-name-requested (URL) | 23 |
| 141 | 6.2.4 Job Status Attributes (Set by Printer) | 23 |
| 142 | 6.2.4.1 current-job-state (type1Enum) | 23 |
| 143 | 6.2.4.2 printer-assigned (name) | 25 |
| 144 | 6.2.4.3 submission-time (dateTime) | 25 |
| 145 | 6.2.4.4 job-message-from-administrator (string) | 26 |
| 146 | 6.2.4.5 completion-time (dateTime) | 26 |
| 147 | 6.2.4.6 job-state-reasons (1#type2Enum) | 26 |
| 148 | 6.2.4.7 impressions-completed (cardinal) | 27 |
| 149 | 6.2.4.8 media-sheets-completed (cardinal) | 27 |
| 150 | 6.2.5 Job Sheet Attributes (Set by Client/End User) | 27 |
| 151 | 6.2.6 Notification Attributes (Set by a Client/End User) | 27 |
| 152 | 6.2.6.1 notification-events (#type1Enum) | 27 |
| 153 | 6.2.7 Job Scheduling Attributes (Set by Client/End User) | 28 |
| 154 | 6.2.7.1 job-priority (type1Enum) | 28 |
| 155 | 6.2.7.2 job-print-after (dateTime) | 28 |
| 156 | 6.2.7.3 job-print-off-peak (type3Enum) | 29 |
| 157 | 6.2.7.4 job-retention-period (deltaTime) | 29 |
| 158 | 6.2.8 Job Production Attributes (Set by Client/End User) | 29 |
| 159 | 6.2.8.1 medium-select (type2Enum) | 30 |
| 160 | 6.2.8.2 number-up (positiveInteger) | 32 |
| 161 | 6.2.8.3 finishing (type2Enum) | 33 |
| 162 | 6.2.8.4 sides (type2Enum) | 34 |
| 163 | 6.2.8.5 copies (positiveInteger) | 34 |
| 164 | 6.2.8.6 printer-resolution-select (positiveIntegerCross) | 34 |
| 165 | 6.2.8.7 print-quality (type2Enum) | 34 |
| 166 | 6.2.8.8 page-select (positiveIntegerRange) | 34 |

| | | |
|-----|--|----|
| 167 | 6.2.8.9 files-are-one-document (Boolean) | 34 |
| 168 | 6.2.8.10 files-are-interleaved (Boolean) | 35 |
| 169 | 6.2.9 Attributes for Conversion of Text Files (Set by Client/End | |
| 170 | User) | 35 |
| 171 | 6.2.9.1 width (cardinalUnits) | 35 |
| 172 | 6.2.9.2 length (cardinalUnits) | 35 |
| 173 | 6.2.9.3 left-margin (cardinalUnits) | 35 |
| 174 | 6.2.9.4 right-margin (cardinalUnits) | 35 |
| 175 | 6.2.9.5 top-margin (cardinalUnits) | 35 |
| 176 | 6.2.9.6 bottom-margin (cardinalUnits) | 36 |
| 177 | 6.2.9.7 repeated-tab-stops (cardinalUnits) | 36 |
| 178 | 6.2.9.8 header-text (string) | 36 |
| 179 | 6.2.9.9 footer-text (string) | 36 |
| 180 | 6.2.9.10 font-size (cardinalUnits) | 36 |
| 181 | 6.2.9.11 number-pages (Boolean) | 36 |
| 182 | 6.2.9.12 default-code-set (type3Enum) | 36 |
| 183 | 6.2.9.13 content-orientation (type2Enum) | 36 |
| 184 | 6.2.10 Job Resource Attributes (Set by the program that produces | |
| 185 | or senses the PDL) | 37 |
| 186 | 6.2.10.1 document-format-used (1#type2Format) | 38 |
| 187 | 6.2.10.2 fonts-used (1#string) | 38 |
| 188 | 6.2.10.3 code-sets-used (1#type3Enum) | 38 |
| 189 | 6.2.10.4 media-used (1#type2Enum) | 38 |
| 190 | 6.2.10.5 sides-used (type2Enum) | 39 |
| 191 | 6.2.10.6 print-quality-used (type2Enum) | 39 |
| 192 | 6.2.10.7 finishing-used (type2Enum) | 39 |
| 193 | 6.2.10.8 printer-resolution-used (positiveIntegerCrossState) | 39 |
| 194 | 6.2.10.9 total-job-octets (positiveInteger) | 39 |
| 195 | 6.2.10.10 job-impression-count (positiveInteger) | 39 |
| 196 | 6.2.10.11 job-media-sheet-count (positiveInteger) | 39 |
| 197 | 6.2.11 Number of Documents (Set by Client) | 39 |
| 198 | 6.2.11.1 number-of-documents (positiveInteger) | 39 |
| 199 | 6.2.12 Document Data (Set by a Client/End User) | 40 |
| 200 | 6.2.12.1 document-format (type2Format) | 40 |
| 201 | 6.2.12.2 document-name (name) | 40 |
| 202 | 6.2.12.3 document-URL (name) | 40 |
| 203 | 6.2.12.4 document-content (octetString) | 40 |
| 204 | 6.3 Operation Attributes (Set by Client) | 40 |
| 205 | 6.3.1 operation-locale (type3Locale) | 41 |
| 206 | 6.3.2 operation-notification-address (name) | 41 |
| 207 | 6.3.3 operation-user-name (name) | 41 |
| 208 | 6.3.4 operation-host-name (name) | 41 |
| 209 | 6.4 Printer Attributes (Set by the Administrator) | 41 |
| 210 | 6.4.1 printer-name (name) | 42 |
| 211 | 6.4.2 printer-location (string) | 42 |
| 212 | 6.4.3 printer-model (string) | 42 |
| 213 | 6.4.4 printer-types (type2Enum) | 42 |
| 214 | 6.4.5 printer-state (type1Enum) | 43 |
| 215 | 6.4.6 printer-state-message (string) | 44 |
| 216 | 6.4.7 message (string) | 44 |
| 217 | 6.4.8 locale (type3Locale) | 44 |
| 218 | 6.4.9 notification-events (#type2Enum) | 45 |
| 219 | 6.4.10 notification-addresses (#name) | 45 |
| 220 | 6.4.11 end-user-acl (#name) | 45 |
| 221 | 6.4.12 maximum-printer-speed (positiveIntegerUnits) | 45 |
| 222 | 6.4.13 fonts-substitutions (#stringPair) | 45 |
| 223 | 6.4.14 fonts-supported (1#stringState) | 46 |
| 224 | 6.4.15 media-supported (1#nameState) | 46 |
| 225 | 6.4.16 document-formats-supported (1#type2FormatState) | 46 |
| 226 | 6.4.17 numbers-up-supported (1#positiveIntegerState) | 46 |

| | | | |
|-----|--------|--|-------------------------------------|
| 227 | 6.4.18 | finishings-supported (#type2EnumState) | 46 |
| 228 | 6.4.19 | sides-supported (1#type2EnumState) | 47 |
| 229 | 6.4.20 | print-qualities-supported (1#type2EnumState) | 47 |
| 230 | 6.4.21 | printer-resolutions-supported | |
| 231 | | (1#positiveIntegerCrossState) | 47 |
| 232 | 6.4.22 | code-sets-supported (1#type3EnumState) | 47 |
| 233 | 6.4.23 | off-peak-times-supported (#type3EnumState) | 47 |
| 234 | 6.4.24 | events-supported (#type2EnumState) | 48 |
| 235 | 6.4.25 | locales-supported (1#type3LocaleState) | 48 |
| 236 | 6.4.26 | job-sheets-supported (#type3EnumState) | 48 |
| 237 | 6.4.27 | maximum-copies (positiveInteger) | 48 |
| 238 | 6.4.28 | maximum-job-octets (positiveInteger) | 48 |
| 239 | 6.4.29 | maximum-impressions (positiveInteger) | 49 |
| 240 | 6.4.30 | maximum-media-sheets (positiveInteger) | 49 |
| 241 | 6.4.31 | maximum-job-retention-period (deltaTime) | 49 |
| 242 | 6.4.32 | maximum-end-user-priority (type1Enum) | 49 |
| 243 | 6.5 | Job Templates | 49 |
| 244 | 6.6 | Conformance | 50 |
| 245 | 7. | Security Considerations | 50 |
| 246 | 8. | References | 50 |
| 247 | 9. | Author's Address | 51 |
| 248 | 10. | Appendix A: Extended Operations | 53 |
| 249 | 10.1 | Modify Job Operation | 53 |
| 250 | 10.1.1 | Modify Job Argument | 53 |
| 251 | 10.1.2 | Modify Job Result | 53 |
| 252 | 10.2 | Resubmit Job Operation | 54 |
| 253 | 10.2.1 | Resubmit Job Argument | Error! Bookmark not defined. |
| 254 | 10.2.2 | Resubmit Job Result | 54 |
| 255 | | | |

256 1. Introduction

257 The Internet Printing Protocol (IPP) is an application level
 258 protocol that can be used for distributed printing on the
 259 Internet. The protocol is heavily influenced by the printing model
 260 introduced in the Document Printing Application (ISO/IEC 10175
 261 DPA) standard, which describes a distributed printing service. DPA
 262 identifies the end user and administrative roles associated with a
 263 distributed printing service, and defines the set of operations
 264 supported by the service. This IPP specification deals initially
 265 only with the end user role. These ideas and concepts, when
 266 unified with other Internet protocols and services, realize a
 267 distributed print service for the Internet.

268 2. Distributed Printing

269 This document assumes a distributed computing environment where
 270 requesters of print services (clients, applications, PC drivers,
 271 etc.) cooperate and interact with print service providers.
 272 Although the underlying configuration may be a complex n-tier
 273 client/server system, an important simplifying step in this
 274 protocol is that the only object the requester of the print
 275 service ever sees is a "printer". It is important, however, to
 276 understand that in a real system, other components of a print
 277 service exist.

278 2.1 Generic Print System Components

279 Every distributed print service, including those using the
280 Internet Printing Protocol, includes elements from the following
281 list.

- 282 - End Users: End Users are humans (or agents who work on behalf
283 of a human) who submit print jobs.
- 284 - Print clients: Print clients are computer network nodes with
285 which humans interact in order to manipulate the distributed
286 print service. A print client uses some protocol to invoke
287 print service operations on another node. Each operation has
288 arguments and results associated with it. The print client
289 provides arguments which add information about the operation
290 requested, and receives results which describe the status and
291 outcome of the operation.
- 292 - Print servers: Printer servers may be embedded in an output
293 device or implemented in a separate system which is associated
294 with an output device. The print server receives requests from
295 the print client and sends back results which describe the
296 status and outcome of the operation requested. A print server
297 normally provides queuing, job management, and device
298 management functions.
- 299 - Queues. Print jobs may be queued or stored on a spool prior to
300 printing. This allows a print service provider to accept one or
301 more print jobs while the printer (or printers) is busy
302 processing another job. Queues, if present, may be implemented
303 in the client, in the server, in the output device, or in some
304 combination of the three.
- 305 - Output Devices. Output devices interpret the print data and
306 generate some form of output. In the case of a laser printer,
307 for example, this normally means rasterizing the print data and
308 putting the resulting marks on paper. An output device may
309 receive print data directly from a client or through a Print
310 server.

311 A specific implementation of a print service may not include all
312 of the elements described here, and the physical packaging of
313 elements is up to the implementation. For example, an output
314 device may include a queue or a print server may include a
315 rasterizer.

316 2.2 IPP Components

317 The print model defined by the Internet Printing Protocol
318 simplifies the user's view of the system components described in
319 the previous section by encapsulating the important elements of
320 the system into three simple objects:

- 321 - End Users
- 322 - Clients
- 323 - Printers (section xxx)
- 324 - Print Jobs (section xxx)
- 325 - Job Templates (section xxx)

These objects are not encapsulations of both data and behavior as in other object oriented models, but are simple collections of attribute/value pairs. [We may try to fix this in our new design, but it's not high priority.]

Clients interact with these using the following operations:

- Print (section xxx)
- Cancel Job (section xxx)
- Get Attributes(section xxx)
- Get Jobs (section xxx)

3. IPP Objects

This section describes the IPP objects.

3.1 Printer

One of the most significant objects in the IPP model is the Printer. To the end user, the Printer object represents the functionality of the actual output device along with the queuing, job management, and device management functions often associated with a print server. An IPP Printer object implements the Internet Printing Protocol. Using the protocol, end users may query the attributes of the Printer, submit jobs to the Printer, determine subsequent states of submitted and queued jobs and state of the Printer, and cancel their own print jobs. The realization of a Printer object may take on different forms for any given configuration of real components. However, the details of the configuration of real components must be transparent to the end user.

Some examples of configurations containing IPP Printer object include:

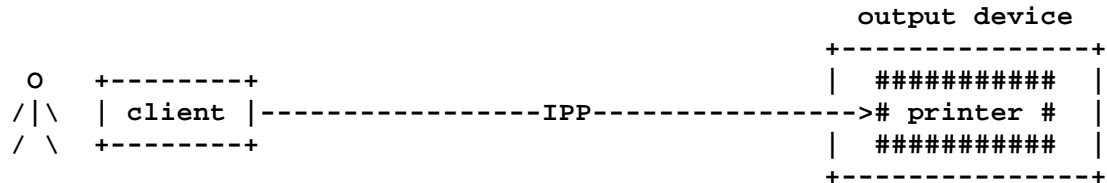
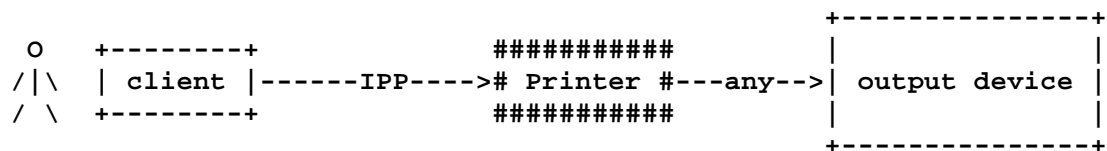
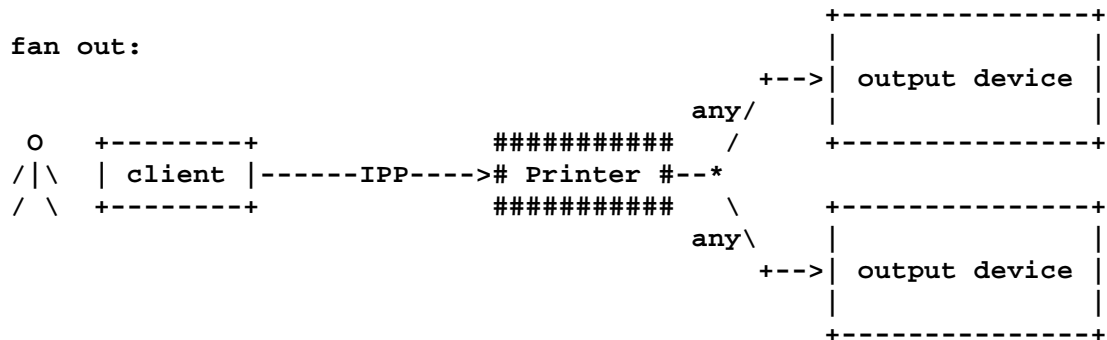
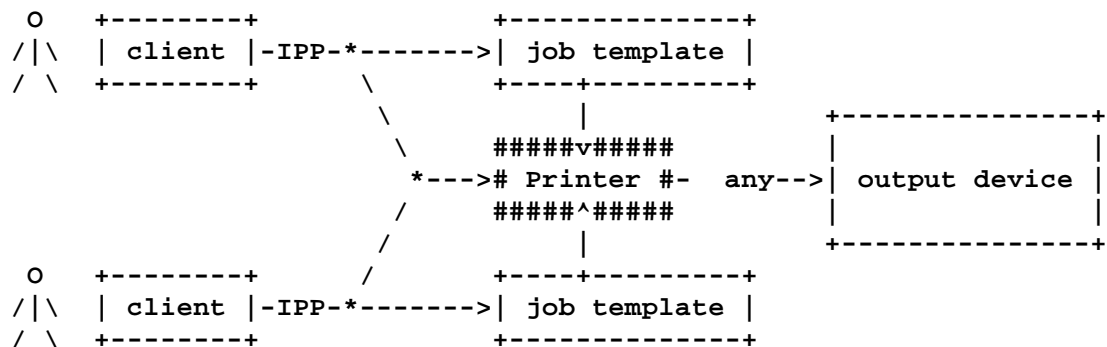
- An output device, with no spooling capabilities, supporting IPP
- An output device, with a built-in spooler, supporting IPP
- A print server with one or more associated output devices with the print server supporting IPP.
 - The associated output devices may or may not be capable of spooling jobs
 - The associated output devices may or may not support IPP
- A print server with one or more downstream print servers and/or output devices where the upstream print server supports IPP
- ISSUE: Is this previous example too much?

See the following figures for some examples on how to view IPP Printer objects on top of other printing system models:

Legend:

indicates an IPP printer object which is either embedded in an output device or is hosted in a server. An IPP printer object may or may not queue/spool.

any indicates any network protocol or direct connect, including IPP

embedded printer:**hosted printer:****fan out:****fan in:**

3.2 Job

A Job object is used to model a job. A job can consist of one or more documents. However, there are no separate document objects. The impact of this is that there are no attributes that pertain to one document in a job but not to others, except for a single attribute that specifies the document data, its location, and its format. Note: In future versions, documents may become separate objects with attributes whose scope and application are different from the corresponding job attributes.

Job attributes provide information to

- identify the print job(section xxx)
- assist in selecting the Printer (section xxx)
- report job status (section xxx)
- assist in scheduling and processing (section xxx)
- describe the documents in the job (section xxx)
- produce the document (section xxx)

ISSUE: This list needs to be fixed to match the final job attributes sections (they don't match right now).

3.3 Job Template

A Job Template object is used to model job defaults. A Job Template is essentially a set of job attributes that initialize a newly created job object.

ISSUE: a job template needs more work. It is currently believed that when a client needs to present a Print Dialog box to an end-user, it gets potential job values and default job values from a Printer. The default values are from the Job Template associated with the Printer named by the end-user. If a end-user sends a job to a Printer, the Printer may set unspecified attributes to the value of the associated Job Template.

3.4 Object Relationships

Instances of objects within the system have relationships which must be maintained persistently along with the persistent storage of the objects themselves. A Printer can contain zero, or more Job objects. A Job object contains one or more Documents. A Printer object is associated with one or more Job Template objects.

3.5 Object Identity

All instances of all objects have an identifier attribute that makes them unique so that they can be unambiguously referenced.

The following objects have the following mandatory identifier attributes:

| Object | Identifier | Containing Object |
|--------|------------|-------------------|
|--------|------------|-------------------|

| | | | |
|-----|--------------|-------------------|---------|
| 471 | | | |
| 472 | Printer | printer-name | None |
| 473 | Job | job-identifier | Printer |
| 474 | | | |
| 475 | Job Template | job-template-name | None |
| 476 | | | |
| 477 | 4. Naming | | |

478 Clients identify Printer objects by using an HTTP type URL. For
 479 example, a URL for a Printer object named "printer-1" whose
 480 network node's domain name is "some.domain.com", might look like:

481 http://some.domain.com/printer-1

482 In this case, the URL identifies the use of the HTTP protocol.
 483 The Printer is located at the node identified by the DNS name
 484 "some.domain.com" and "printer-1" is the name of the Printer.

485 Another example is the following URL:

486 http://1.2.3.4:380/printer-2

487 In this case, the URL identifies the use of the HTTP protocol.
 488 The Printer is located at the node identified by the IP address of
 489 "1.2.3.4" using port 380 for the HTTP server, and "printer-2" is
 490 the name of the Printer.

491 It is not necessary to expose the Job Template objects that might
 492 be associated with a given printer as separate objects. They can
 493 be exposed in two ways through URL naming.

494 - The Job Template can be hidden from the end user by a URL that
 495 represents just Job Template name (but does not expose the
 496 Printer object name) as the two URLs 1)
 497 http://some.domain.com/two-sided-printer and 2)
 498 http://some.domain.com/draft-printer. These look like two
 499 different locations, but underneath they represent the same
 500 Printer object but using two different Job Template default
 501 attribute sets.

502 - The Job Template can be exposed along with the name of the
 503 Printer object directly in the URL as in:
 504 http://some.domain.com/hr-printer/resumes. In this case there
 505 is a "resumes" Job Template associated with the "hr-printer"
 506 Printer.

507 ISSUE: Should IPP propose a new standard port number (say 380)
 508 for an HTTP server which has been optimized to support the IPP
 509 protocol over that HTTP implmentation? IPP should work for any
 510 valid HTTP server, however, there might be some specializations
 511 can be performed for IPP operations.

512 4.1 Directory Services

513 IPP does not require any specific directory service. However,
 514 this specification does define a generic schema that can be used
 515 for any specific instance of a directory service. That is, some
 516 of the attributes from the Printer object are called out as

attributes that may be added to a directory entry which represents that Printer. This allows directory users to find and locate IPP Printers by either a simple name look up or by some filtered attribute search.

4.2 Directory Entry Schema

The following attributes define the generic directory entry schema. All directories entries for IPP Printers in all types of directories should support at least these attributes.

Name, description, owner, location, address

4.2.1 Name

This is the printers name. It is a URL so it contains sufficient information to not only name, but to address the printer using IPP as well.

4.2.2 Description

This is a free form string that can contain any site specific descriptive information about this printer.

4.2.3 Location

This is a free form string that can contain any site specific location information.

In order for filtered searches to be more effective, a given site may use some regular structuring within the string values such as "SITE:USA-San Jose,BUILDING:A1,FLOOR:2,ROOM:555" or "department5-2ndFloor-A5-IndianHills-Chicago-IL-USA".

4.2.4 Print Quality

This indicates a somewhat subjective evaluation of the overall printing quality: "high", "medium", or "low".

ISSUE: Does this subsume the need for Resolution and Speed?

4.2.5 Cost

4.2.6 This indicates a somewhat subjective evaluation of the overall cost of printing at this printer: "high", "medium", or "low".Resolution

This is the maximum resolution of the Printer in dpi.

The syntax shall be the same as that of the printer-resolution-select job attribute. That syntax allows a single integer to specify the maximum resolution or a pair of integers to specify the maximum resolution when the x and y dimensions differ. When two integers are specified, the first is in the x direction, ie., the direction fo the shortest dimension of the medium, so that the

559 value is independent of whether the Printer feeds long edge or
560 short edge first.

561 4.2.7 Color Supported

562 This is a BOOLEAN for either yes, color printing is supported, or
563 no color printing is not supported.

564 4.2.8 Fonts Supported

565 This attribute takes on a list of fonts that are supported by the
566 printer. This is replicated from the fonts-supported attribute in
567 the Printer object.

568 4.2.9 Maximum Speed

569 This is the maximum speed of the printer ppm, ipm, lpm, or cps.
570 They syntax and values are the same as the maximum-printer-speed
571 Printer attribute.

572 ISSUE: Should this be "high", "medium", and "low"??

573

574 ISSUE: Delete??

575 4.2.10 Device Id

576 This is the IEEE P1284 Device Id. This attribute can be used for
577 automatic driver download and other automatic configuration tasks.
578 It can be used to generate a platform specific id such as the
579 Windows Plug-and-Play id.

580 4.2.11 Model

581 This is a simple text string defined by the manufacturer.

582 ISSUE: Is this needed if we use the P1284 Device Id??

583 4.2.12 Manufacturer

584 This is a simple text string defined by the manufacturer. There
585 is no registration, and there is a possibility of overlap, but the
586 goal is to keep this simple, not too complex.

587 ISSUE: Is this needed if we use the P1284 Device Id??

588 ISSUE: Would the company name registered with IAN for use in
589 specifying Internet addresses be required? Recommended?

590 4.2.13 Type

591 This is the printing mechanism of the print device: laser, ink
592 jet, thermal, etc. The syntax and values are the same as for the
593 printer-types Printer attribute. The value of the attribute shall
594 be single-valued, while the printer-types attribute is multi-

595 valued. A Printer that is of several types may appear multiple
596 times in the directory. ISSUE: no.

597 ISSUE: Is this needed if we use the P1284 Device Id??

598 4.2.14 Document Formats s Supported

599 This is a list of all of the document formats that the printer
600 and/or its interpreter(s) support. The syntax and values are the
601 same as those for the document-formats-supported Printer
602 attributes.

603 4.2.15 Sides Supported

604 This attribute specifies the capabilities of the Printer for
605 marking on sides of the medium. The syntax and values shall be
606 the same as the sides-supported Printer attribute. Standard
607 values are: 1-sided (simplex), 2-sided-long-binding-edge (duplex),
608 and 2-sided-short-binding-edge (tumble).

609 4.2.16 Finishings Supported

610 This attribute identifies the finishing operations supported by
611 the Printer. The standard finishing objects are defined in the
612 section on the finishing job attribute.

613 5. IPP Operations

614 The following symbols are used in the tables below:

615
616 P perform the operation directly
617 PF perform the operation; forward to Output Device sometimes
618 UA unsupported in an Output Device unless it supports queuing
619 U unsupported operation
620

621 IPP defines the following end user operations:

| 622 Operation | Print Server | Output Device |
|---------------------|--------------|---------------|
| Print | PF | P |
| Cancel Job | PF | P |
| Get Attributes | PF | P |
| Get Jobs | PF | P |

623

624

625 5.1 IPP Operations Using HTTP

626 All IPP operations are defined using HTTP as the underlying
627 communication protocol.

628 5.1.1 HTTP Overview

629 IPP is based on the existing HTTP standard. IPP is a lightweight
630 application-level protocol designed with the Internet in mind. It
631 is a generic, stateless, object-oriented protocol which can be
632 used for any task through extension of its request methods
633 (commands).

HTTP allows an open-ended set of methods to be used to indicate the purpose of a request. It builds on the discipline of reference provided by the Uniform Resource Location (URL) and message formats similar to those used by Internet Mail and the Multipurpose Internet Mail Extensions (MIME).

HTTP is based on a request-response paradigm. A requesting program (a client) establishes a connection with a receiving program (a server) and sends a request to the server in the form of a request method, a URL, and protocol version, followed by a MIME-like message containing request modifiers, client information, and possibly print data. The server responds with a status line, including its protocol version, and a success or failure code, followed by a MIME-like message containing server information, entity meta-information, and possibly some content.

Current practice requires that the connection be established by the client prior to each request and closed by the server after sending the response. Both clients and servers must be capable of handling cases where either party closes the connection prematurely, due to user action, automated time out, or program failure.

5.1.2 IPP Operation Encoding

IPP messages consist of requests from client to server and responses from server to client.

HTTP MESSAGE = Request | Response

Requests and responses use the generic message format of RFC 822 for transferring entities. Both messages may include optional header fields and an entity body. The entity body is separated from the headers by a null line (a line with nothing preceding the CRLF).

```
Request = Request-line
        * (General-Header
          | Request-Header
          | Entity-Header)
        CRLF
        [ Entity-Body ]
```

```
Response = Status-line
        * (General-Header
          | Request-Header
          | Entity-Header)
        CRLF
        [ Entity-Body ]
```

All IPP headers conform to the syntax

IPP Header = field name ":" [field-value] CRLF.

IPP/1.0 defines the octet sequence CR LF as the end-of-line marker for all protocol elements except the entity-body. In this document, the sequence CR LF is shown as CRLF.

686 Note that HTTP 1.1 defines a slightly different syntax, allowing
687 for dynamically generated messages to be transmitted. This would
688 be required for cases such as PC driver generated Print
689 Operations. HTTP 1.1 defines a message header which specifies a
690 transfer encoding called "chunks".

691 5.1.2.1 HTTP Request-Header Fields

692 HTTP request header fields allow the client to pass additional
693 information about the request, and about the client itself, to the
694 server. All header fields are optional and when used it is
695 assumed that IPP would use these headers in a standard way. IPP
696 requests will be completely encapsulated within the entity body of
697 an HTTP request.

```
698 HTTP Entity-Header = Content-Encoding
699                        | Content-Length
700                        | Content-Type
701                        | extension-header
702
703
```

704 The **Content-Length** field must always be a valid length. This means
705 that for any Print Operations based on HTTP 1.0, the entire
706 content must be generated before this header can be built. HTTP
707 1.1 provides the notion of "chunks" which will allow the content
708 to be generated dynamically as the data is sent.

709
710 **Content-Type** will always be "Application/IPP".

711 The http method token indicates the method to be performed on the
712 resource identified by the Request-URL. The method is case-
713 sensitive. The http methods used will be "Post" and "Get".

714 5.1.2.2 Print

715 The Print operation allows a user to submit a Print Job to the
716 print server. A Print Job contains the information needed by the
717 Print object to print a document or set of documents. When the
718 print operation is invoked, the Entity-Body included in the HTTP
719 request is an IPP Print Job. The concrete syntax of the Print Job
720 is defined in section xxx. The response to a print request
721 includes the Job Identifier (a URL) assigned by the Printer.

722 5.1.2.3 Cancel Job

723 This method allows a user to cancel one specific Print Job any
724 time after the print job has been established on the Printer
725 Object. Some pages may be printed before a job is terminated if
726 printing has already started when the Cancel Job operation is
727 received.

728 The Cancel HTTP request will be sent to the URL identifying the
729 job to be canceled.

730 5.1.2.4 Get Attributes

731 This operation allows a user to obtain information from the Print
 732 object concerning jobs, printers, and print queues, based on ISO
 733 10175. The entity-body of the Get Attributes operation contains
 734 the set of attributes that the requester is interested in.
 735 However, the attribute values may be null and are ignored by the
 736 server. The attribute list is returned in the response with the
 737 appropriate attribute values filled in. If no attribute list is
 738 supplied, then all attributes defined for that object are
 739 returned.

740 5.1.2.5 Get Jobs

741 The Get Jobs operation allows a client to retrieve attributes of
 742 the specified job.

743 5.1.3 The Print-Job

744 The entity body of a print request will contain a Print Job, as
 745 defined below. The headers defined here are IPP headers, but
 746 follow the same syntax as the basic HTTP headers.

```

747
748     Print Job = Print-Job-Object-Header      section (1.2.1)
749               [Job Attributes]              section (1.2.4)
750               *(Documents)
751
752     Job Attribute = Attribute name : Attribute value    CRLF
753
754     Document =      Document-Header            section (1.2.2)
755                   [Document attributes]        section (1.2.5)
756                   [Content-Header              section (1.2.3)
757                   content]
758
759

```

759 5.1.3.1 Print Job Object Header

```

760     Print-Job-Object Header = Content-Encoding
761                               | Content-Length
762                               | Content-Type
763                               | extension-header
764
765     Content-Type is always "IPP Print Object". Other header fields
766     are as defined for HTTP 1.0.

```

767 5.1.3.2 Document Header

768 The document header allows the insertion of multiple documents
 769 within a job. At this point only a limited number of document
 770 attributes are defined. However, this structure allows the
 771 addition of other attributes which can be specified on a document
 772 boundary.

```

773     Document Header = Content-Encoding
774                       | Content-Length
775                       | Content-Type
776                       | extension-header
777

```


778 Content type is always "IPP Document". Other header fields area as
779 defined in HTTP 1.0.

780 5.1.3.3 Document-Content Header

781 The document-content-header provides additional meta-information
782 about the document. The document content header is an optional
783 field and would not be present if the document was pointed to by a
784 document URL attribute. It is composed of a number of document
785 header fields as follows:

```
786
787 Document-Content-Header =      Content-Encoding
788                             | Content-Length
789                             | Content-Type
790                             | extension-header
791
```

792 Content-Type is defined as :

793 Content-Type = Data Stream Format "/" Version

794
795 Thus, for example, if the document to be printed was a Postscript
796 Level 2 document, the Content-Type would be specified as:

```
797
798 Content-Type: Postscript/2.0
799
```

800 Other header fields are as defined by HTTP 1.0.

801 5.1.3.4 Job Attributes

802 Job attributes are defined in section xxx. Attributes will always
803 be sent as

```
804
805 Job-Attribute = attribute name ":" Attribute value      CRLF
806
807 Attribute value = Value | *(Value ",", Value)
808
809
```

810 5.1.3.5 Document Attributes

811 Document attributes are defined in section yyy. At this point a
812 limited number of attribute may be specified on a document basis.
813 The syntax for a document attribute is

```
814
815 Document-Attribute = attribute name ":" Attribute value      CRLF
816
817 Attribute value = Value | *(Value ",", Value)
818
```

818 5.2 Print Operation

819 When an end user submits a job, the client submits a Print
820 Request according to the syntax and semantics of this standard and
821 receives a Print Response according to this standard. The end-
822 user or submitting application selects a Printer which implies a
823 Job Template.

824 [Further work needs to done to define the above concept.]

825 5.2.1 Print Request

826 The following abstract data types are part of the Print Request.

827 Note: The Printer name is not needed since it is a URL and it is
828 the target of the entire operation.

| | |
|-----------------------------|--|
| Job and Document Attributes | A set of Job object and Document attributes as defined in section xxx |
| Job Return Attributes | The set of Job attributes to return in the response |
| Printer Return Attributes | The set of Printer attributes to return in the response |
| Document Contents | Note: What if there are multiple documents and each has a different size? How does this map on top of the HTTP header that has one size? Does it require multiple HTTP operations? |

829

830

831 5.2.2 Print Response

832 The following abstract data types are part of the Print Response:

833

| | |
|---------------|--|
| Job Id | Used for all other operations on this Job. |
| Job Status | Job state information |
| Printer State | Optional Printer state information |
| Message | Optional message Note: Is this needed? |
| Errors | Optional Error Information |

834

835

836 5.3 Cancel Job Operation

837 5.3.1 Cancel Job Request

838 The following abstract data types are part of the Cancel Job

839 Request. Note: The Job Id is not needed as data within the
840 operation since the Job URL is the target of the entire operation.

841

| | |
|-----------------|--|
| Document Number | Optional document number of the document to Issue: We don't need this if we do not allow canceling of a single document in a multi-document job, right? |
| Message | Optional message to the operator. |

842

843 5.3.2 Cancel Job Response

844 The following abstract data types are part of the Cancel Job
 845 Response:

846 Job Status Optional Job status information
 Errors Optional Error Information

847
 848 5.4 Get Attributes Operations

849 5.4.1 Get Attributes Request

850 The following abstract data types are part of the Get Attributes
 851 Request:

 Selector A Job Id or Printer Name URL ISSUE: Is this
 just the target URL of the operation and it is
 not needed here within the operation?
 Requested
 Attributes A set of attributes in which the requestor is
 interested

852
 853 5.4.2 Get Attributes Response

854 The following abstract data types are part of the Get Attributes
 855 Response:

 Result The requested attributes of the object
 Attributes
 Errors Optional error information

856
 857 5.5 Get Jobs Operation

858 5.5.1 Get Jobs Request

859 The following abstract data types are part of the Get Jobs
 860 Request:

861 Filtering A lightweight filtering mechanism, such as all
 jobs versus a particular end user's jobs.
 Requested
 Attributes A set of job attributes in which the requestor
 is interested

862
 863
 864 5.5.2 Get Jobs Response

865 The following abstract data types are part of the Get Jobs
 866 Response:

867 Result Attribute set containing the returned results.
 Attributes
 Errors Optional Error Information

868

869 6. Object Attributes

870 This section describes the attributes, syntaxes, and values that
 871 are part of IPP. The sections below show the objects and their
 872 associated attributes which are included within the scope of this
 873 protocol. The text in these sections has been heavily influenced
 874 by the ISO/IEC 10175 DPA (Final, June 1996).

875
 876
 877

878 6.1 Attribute Syntaxes

879 The sections below reference the following syntax items:

| | |
|----------------------------|--|
| string | arbitrary ASCII strings, no control characters, except <SPACE>.TBD |
| stringPair | string ":" string |
| stringState | string state |
| name | arbitrary ASCII strings, no control characters, and no <SPACE> characters.TBD |
| URL | Universal Resource LocatorTBD |
| dateTime | date and time in RFC 822 formatTBD |
| deltaTime | [hours ":"] minutes |
| cardinal | 0 .. n represented as ASCII digits |
| type1Enum | standard names, must revise the IPP standard to add a new name. No private names are allowed.TBD |
| type2Enum | standard names, but an implementor can add new TBDby proposing them to the PWG for registration (or an IANA-appointed registry advisor after the PWG is no longer certified) anytime. IANA keeps the registry. Implementors can add private (un-registered) with a suitable distinguishing prefix, such as -xxx- where xxx is the company name registered with IANA. |
| type3Enum | standard names, but an implementor can add new names by submitting a registration request directly to IANA, no PWG or IANA-appointed registry advisor review is required. Implementors can add private (un-registered) names with a suitable distinguishing prefix, such as -xxx- where xxx is the company name registered with IANA.TBD |
| type2EnumState | type2Enum state |
| type3EnumState | type3Enum state |
| state | TBD |
| Boolean | tokens: yes, y, true, or t and no, n, false, or f.TBD |
| positiveInteger | 1 .. n represented as ASCII digitsTBD |
| positiveIntegerCross | positiveInteger ["x" positiveInteger] |
| positiveIntegerCross State | positiveIntegerCross state |

| | |
|----------------------|---------------------------------------|
| positiveIntegerRange | positiveInteger ":" positiveInteger |
| positiveIntegerUnits | positiveInteger units |
| positiveIntegerState | positiveInteger state |
| units | "ppm" "ipm" "spm" "cps" "lpm" |
| type3Locale | type3Country ":" type3Language ":" |
| | type3CodeSet |
| type3Country | type3Enum |
| type3Language | type3Enum |
| type3CodeSet | type3Enum |
| type2Format | name ["/" version] |
| version | name |
| type3LocaleState | type3Locale state |

880

881

882 6.2 Job Attributes

883 A job object contains a set of job attributes and one or more
 884 documents. A client shall create a job and send it to a server
 885 using the Print operation. A client may use a job template
 886 associated with the selected printer in order to initialize the
 887 job.

888 Each section heading below contains the name of an attribute and
 889 its syntax in parentheses using the rules of RFC 822.

890 6.2.1 Job Informational Attributes (Set by a Client/End User)

891 The client may specify these attributes in the Print operation to
 892 provide information to identify a print -job.

893 The client may also specify these attributes in the operations:
 894 Get-Attributes, and Get-Jobs.

895 6.2.1.1 job-name (string)

896 This attribute supplies a human readable string for naming the
 897 print-job.

898 This attribute is intended for to be printed on a start sheet,
 899 returned in a Get-Jobs result, or used in notification messages.

900 If the client does not specify this attribute, a Printer shall set
 901 it to the name of the file of the first document in the job.

902 6.2.2 Job Informational Attributes (Set by a Printer)

903 The Print shall add all of these attributes to a job to provide
 904 information to identify a print -job.

905 The client may specify these attributes in the operations: Get-
 906 Attributes and Get-Jobs, but not in Print.

907 6.2.2.1 job-identifier (url)

908 This attribute provides the job -identifier for this job on the
 909 Printer. The Printer shall generate a job -identifier value as a
 910 URL..

911 The value of the job -identifier attribute shall be returned by the
912 Printer as part of the PrintResult in the Print operation.

913 6.2.2.2 job-originator (name)

914 This attribute specifies the name of the person submitting the
915 print job. The Printer shall set this attribute to the most
916 authentic name that it can obtain from the client. The operation-
917 user-name attribute is intended to be a source of the most
918 authentic name.

919 6.2.2.3 job-originating-host (name)

920 This attribute identifies the originating host of the job. The
921 Printer shall set this attribute to the value of the operation-
922 host-name which is intended to be the most authentic host name of
923 the client.

924 6.2.2.4 notification-address (name)

925 This address specifies the email address of the client. The client
926 specifies this attribute in the operation-notification-address
927 attribute which the Printer in turn uses to set this attribute.

928 The Printer shall use this attribute as the address for sending
929 messages to a job submitter when an event occurs that the end user
930 has registered an interest in or when certain other events occur,
931 such as Cancel-Job.

932 Note: The only type of notification is email.

933 ISSUE: can the email address be inferred with job-originator and
934 the originating-host.

935 6.2.2.5 job-locale (type3Locale)

936 This attribute identifies the locale of the job. The Printer sets
937 this attribute from the value of the operation-locale.

938 The Printer shall use this attribute to determine the locale for
939 notification messages that it sends.

940 The type3EnumTrip consists of 3 colon separated type 3 enums. The
941 first shall be the two-character country code from ISO 639. The
942 second shall be the two-character language code from ISO 3166. The
943 third is the code-set from the IANA Code Set Registry.

944 ISSUE: is there a more standard syntax for locale?

945 6.2.3 Printer Selection Attributes (Set by Client/End User)

946 The client shall specify this attribute to select a particular
947 Printer.

948 The client may also specify these attributes in the operations:
949 Get-Attributes, and Get-Jobs.

950 **Issue:** this attribute may be implicit in the specified URL in the
 951 Print operation.

952 6.2.3.1 printer-name-requested (URL)

953 This attribute identifies the printer that the client requests
 954 for printing the job.

955 ISSUE: We decided to delete this attribute!

956 6.2.4 Job Status Attributes (Set by Printer)

957 The Printer shall add these attributes to a job when a client
 958 submits a job, and the Printer shall assign appropriate values to
 959 each such job -status attribute.

960 The Printer uses these attributes to specify the job status
 961 before, during and after the processing of the print -job by the
 962 Printer.

963 The client may specify job -status attributes in: Get-Attributes
 964 and Get-Jobs, but not Print.

965 6.2.4.1 current-job-state (typeName)

966 This attribute identifies the current state of the job with the
 967 following values: unknown, pre-processing, pending, processing,
 968 printing, held, terminating, retained, completed.

| | |
|----------------|--|
| unknown | The job state is not known, or is indeterminate. |
| pre-processing | The job has been created on the server by the create-job sub-operation of the print-request, but a print-request with a TRUE value for the job-submission-complete component of the PrintArgument has not yet been received and no document has started processing. The job maybe in the process of being checked by the server for attributes, defaults being applied, a printer being selected, etc. |
| held | The job is waiting to be released for scheduling for any number of reasons as specified by the value of the job's job-state-reasons attribute. |
| pending | The job's job-submission-complete attribute is TRUE since the server has received a print-request with the job-submission-complete parameter TRUE and the job is waiting to start processing on a printer. |
| processing | The server is processing the job, or has made the job ready for printing, but the output device is not yet printing it, either because the job hasn't reached the output device or because the job is queued in the output device or some other spooler, awaiting the output device to print it. |

| | |
|-------------|--|
| printing | The server has completed processing the job and the output device is currently printing the job on at least one printer. That is, a print engine is either printing pages of the job, or failing in its attempt to print pages of the job because of some wait state, such as, start-wait, end-wait, needs-attention, etc. The complete job state includes the detailed status represented in the printers' printer-state attribute(s). |
| paused | The job has been paused as a result of a PauseJob operation. |
| interrupted | The job was interrupted by the InterruptJob request for an intervening job, and shall resume processing automatically once the intervening job has completed. |
| terminating | The job has been cancelled by a CancelJob request or aborted by the server and is in the process of terminating. The job's job-state-reasons attribute contains the reasons that the job is being terminated. |
| retained | <p>The job is being retained at the server as a result of the job's job-retention-period being non-zero. The job has (1) completed successfully or with warnings or errors, (2) been aborted while printing by the server, or (3) been cancelled by the CancelJob request before or during processing. The job's job-state-reasons attribute contains the reasons that the job has been retained.</p> <p>While in the retained state, all of the job's document data (and resources, if any) shall be retained by the server; thus a job in the retained state could be reprinted, using some means outside the scope of ISO/IEC 10175-Part 1.</p> |

completed The job has:
 (1) completed successfully or with
 warnings or errors,
 (2) been aborted by the server while
 printing, or
 (3) been cancelled by the CancelJob
 request,
 AND the job's:
 (1) job-retention-period was zero or has
 expired, or
 (2) job-discard-time has arrived.
The job's job-state-reasons attribute contains
the reason(s) that the job has been completed.
While in the completed state, a job's document
data (and resources if any) need not be
retained by the server; thus a job in the
completed state could not be reprinted. The
length of time that a job may be in this
state, before transitioning to unknown, is
implementation-dependent. However, servers
that implement the completed job-state shall
retain, as a minimum, the following attributes
for any job in the completed state: job-
identifier, job-owner, job-name, current-job-
state, printers-assigned, and job-state-
reasons.

969

970 The IPP protocol supports all values for job states, but Printers
971 are need only support those states which are appropriate for the
972 particular implementation.

973 6.2.4.2 printer-assigned (name)

974 This attribute identifies the Output Device to which the Printer
975 has assigned this job.

976 If an Output Device implements a Printer, the Printer does not set
977 this attribute.

978 If a Print Server implements a Printer, the value shall be empty
979 until the Printer assigns an Output Device to the job..

980 **ISSUE:** Is this attribute appropriate for a model in which we are
981 hiding the downstream Printer. The printers -assigned value shall
982 not be the same as the printer requested by the end user.

983 The value of the job's printer -assigned attribute shall remain
984 after the job has completed, so that end users can determine the
985 Output Device on which the job was printed.

986 6.2.4.3 submission-time (dateTime)

987 This attribute indicates the time at which this job was accepted
988 by the Printer. If the Printer does not support the notion of
989 time, the attribute is not stored as part of the job object.

990 6.2.4.4 job-message-from-administrator (string)

991 This attribute provides a message from an operator, system
 992 administrator or 'intelligent' process to indicate to the end user
 993 the reasons for modification or other management action taken on a
 994 job.

995 6.2.4.5 completion-time (dateTime)

996 This attribute indicates the time at which this job completed.
 997 This time is useful for jobs which are retained after printing.
 998 If the Printer does not support the notion of time, the attribute
 999 is not stored as part of the Job object.

1000 6.2.4.6 job-state-reasons (1#type2Enum)

1001 This attribute identifies the reason or reasons that the job is in
 1002 the state that it is in (e.g., held, terminating, retained,
 1003 completed, etc.). The printer shall indicate the particular
 1004 reason(s) by setting the value of the job -state-reasons attribute.
 1005 It is valid for the printer to set the value of the
 1006 job-state-reasons attribute to the empty set.

1007 The following standard values are defined:

| | |
|------------------------------|--|
| documents-needed | The complete job has been accepted by the server (the value of the job-submission-complete element was TRUE in the last print-request for the job), but the server is waiting for its files to be transferred before the job can be scheduled to be printed. |
| job-hold-set | The value of the job's job-hold attribute is TRUE. |
| job-print-after-specified | The value of the job's job-print-after attribute has specified a time specification that has not yet occurred. |
| Required-resources-not-ready | At least one of the resources needed by the job, such as media, fonts, resource objects, etc., is not ready on any of the physical printer's for which the job is a candidate. |
| Successful completion | The job completed successfully. |
| Completed-with-warnings | The job completed with warnings. |
| Completed-with-errors | The job completed with errors (and possibly warnings too). |
| Cancelled-by-user | The job was cancelled by the user using the CancelJob request. |
| Cancelled-by-operator | The job was cancelled by the operator using the CancelJob request. |
| Aborted-by-system | The job was aborted by the system. |
| Logfile-pending | The job's logfile is pending file transfer. |
| Logfile-transferring | The job's logfile is being transferred. |

1008

1009

1010 6.2.4.7 impressions-completed (cardinal)

1011 This attribute contains the number of impressions that the Printer
1012 has completed printing. If the Printer cannot report this number,
1013 the Printer leaves this attribute unspecified.

1014 6.2.4.8 media-sheets-completed (cardinal)

1015 This attribute contains the number of media-sheets that the
1016 Printer has completed printing. If the Printer cannot report this
1017 number, the Printer leaves this attribute unspecified.

1018 6.2.5 Job Sheet Attributes (Set by Client/End User)

1019 The client shall specify these attributes to control the printing
1020 of job sheets.

1021 The client may also specify job sheet attributes in: Get-
1022 Attributes and Get-Jobs.

1023 job-sheets (type3Enum) This attribute determines what type of job-
1024 sheets the Printer shall print with the job.

1025 The standard values are: none, and default-sheet.

1026 The value 'none' means that the Printer shall print no job sheets.
1027 The value 'default-sheet' means that the Printer shall print the
1028 job sheets defined by an administrator. If the administrator's
1029 policy is not to support none, the Printer shall use the default-
1030 sheet value if the client supplies the "none" value.

1031 6.2.6 Notification Attributes (Set by a Client/End User)

1032 The client shall specify these attributes to indicate events that
1033 the client is interested in.

1034 The client may also specify notification attributes in: Get-
1035 Attributes and Get-Jobs.

1036 6.2.6.1 notification-events (#type1Enum)

1037 This attribute specifies the events about which the end user want
1038 to be notified.

1039 This attribute will support four events classes: none, job-
1040 completion, job-problems and printer-problems.

1041 If this attribute contains the event none, the Printer shall not
1042 notify. This value is useful if an administrator has set up a
1043 notification Printer default but the end user does not which
1044 notification. If the none value and other values are supplied,
1045 the Printer shall ignore the none value.

1046 This attribute will support only one delivery method, namely
1047 email. The attribute notification-address specifies the email
1048 address.

1049 If this attribute contains the event job-completion, the Printer
1050 shall notify the client when the job containing this attribute
1051 completes.

1052 If this attribute contains the event job-problem, the Printer
1053 shall notify the client when the job containing this attribute has
1054 a problem while the job is printing. Problems include: paper jam
1055 and out-of-paper.

1056 If this attribute contains the event printer-problem, the Printer
1057 shall notify the client when the job containing this attribute has
1058 a problem while the job is printing or waiting to print. Problems
1059 include: paper jam and out-of-paper.

1060

1061 **ISSUE:** Email is quite deficient for timely notification to an end-
1062 user who receives a lot of email, but there are no other choices.
1063 The internet community needs to solve this problem, perhaps with
1064 an extremely-urgent email. Intranet usage may have other
1065 solutions.

1066 6.2.7 Job Scheduling Attributes (Set by Client/End User)

1067 The client shall specify these attributes to provide the Printer
1068 with information for the scheduling a print -job.

1069 The client may also specify these attributes in: Get-Attributes
1070 and Get-Jobs.

1071

1072 6.2.7.1 job-priority (type1Enum)

1073 This attribute specifies a priority for scheduling the print -job.
1074 Printers that employ a priority -based scheduling algorithm use
1075 this attribute.

1076 There are three standard values: high, default, and low. Among
1077 those jobs that are ready to print, a Printer shall print all such
1078 jobs with a high priority before printing those with a default or
1079 low priority, and a Printer shall print all such jobs with a
1080 default priority before printing those with a low priority.

1081 If the client does not specify this attribute, the Printer assumes
1082 that the end user places no constraints concerning priority on the
1083 scheduling of the print -job, and it has a priority value of
1084 default

1085 An operator can modify a job to have any priority. An end-user is
1086 restricted by the value of the maxium-end-user-priority Printer
1087 attribute.

1088 6.2.7.2 job-print-after (dateTime)

1089 This attribute specifies the calendar date and time of day after
1090 which the print-job shall become a candidate for printing.

1091 If the value of this attribute is in the future, the Printer shall
1092 set the value of the job's **current-job-state** to **held** and add the
1093 **job-print-after-specified** value to the job's **job-state-reasons**
1094 attribute and shall not schedule the print-job for printing until
1095 the specified date and time has passed. When the specified date
1096 and time arrives, the Printer shall remove the **job-print-after-**
1097 **specified** value from the job's **job-state-reason** attribute and, if
1098 no other reasons remain, shall change the job's **current-job-state**
1099 to **pending** so that the job becomes a candidate for being scheduled
1100 to print.

1101 If this attribute is unspecified or the value is in the past, the
1102 job shall be a candidate for scheduling immediately.

1103 6.2.7.3 job-print-off-peak (type3Enum)

1104 This attribute specifies the off-peak period during which the
1105 print-job shall become a candidate for printing.

1106 Standard values are: 'evening', 'night', 'weekend', 'second-
1107 shift', 'third-shift'.

1108 If this attribute is specified, it contains a value with which an
1109 administrator has associated allowable print times. An
1110 administrator is encouraged to pick names that suggest the type of
1111 off-peak period.

1112 If this attribute is unspecified, the job shall be a candidate for
1113 scheduling immediately.

1114 6.2.7.4 job-retention-period (deltaTime)

1115 The retention time is expressed in hours and minutes, e.g. 6:00 (6
1116 hours), or 20 (20 minutes).

1117 This attribute specifies the minimum period of time following the
1118 completion of job processing and printing that the server shall
1119 keep job attributes and document data. The Printer may keep these
1120 attributes and data longer than the value of the
1121 job-retention-period attribute.

1122 ISSUE: Should this be deleted?

1123 6.2.8 Job Production Attributes (Set by Client/End User)

1124 The client shall specify these attributes to affect the rendering,
1125 production and finishing of the documents in the job. Similar
1126 types of instructions may also be contained in the document to be
1127 printed.

1128 If there is a conflict between the value of one of these
1129 attributes, and a corresponding instruction in the document
1130 (either implicit or explicit), the value of the attribute shall
1131 take precedence over the document instruction.

1132 Job Production and Resource Attributes each address a similar set
1133 of features but they have different uses.

1134 A job production attribute provides a client with a way to request
 1135 some feature that is not embedded within the document data. After
 1136 some program has merged the production attributes into the
 1137 document data After the information from these attributes has been
 1138 folded into the document data (possibly during a translation
 1139 process of the document data), these attributes are no longer
 1140 relevant and shall can be discarded from a job. Instead, the
 1141 resource attributes specify the resources needed to print the job
 1142 as modified by the job production attributes.

1143 Note: until companies that supply interpreters for PDL's, such as
 1144 PostScript and PCL allow a way to specify overrides for internal
 1145 job production instructions, a Printer may not be able to
 1146 implement these attributes for some PDL's.

1147 A job resource attribute tells a Printer what features the job
 1148 needs. A program that translates document data to a Printer's PDL,
 1149 and/or merges production attributes into the document data should
 1150 add job resource attributes to a job.

1151 For example, a job production attribute medium-select with the
 1152 value of 'letter' requests that a job be printed on letter paper,
 1153 but gives no information about what resources the job needs. A
 1154 resource production attribute media-used with the values of
 1155 'letter' and 'ledger' tell a Printer that the job needs letter and
 1156 ledger paper, but gives no information about which pages use each
 1157 medium.

1158

1159 The client may also specify document production -instruction
 1160 attributes in: Get-Attributes and GetJobs.

1161 6.2.8.1 medium-select (type2Enum)

1162 This attribute identifies the medium that the Printer shall use
 1163 for all pages of the document regardless of what media are
 1164 specified within the document.

1165 The values for medium include medium-names, medium-sizes, input-
 1166 trays and electronic forms so that one attribute specifies the
 1167 media.

1168 Standard values are defined (taken from ISO DPA and the Printer
 1169 MIB)

1170

| | |
|--------------------|--|
| default | The default medium for the output device |
| iso-a4-white | Specifies the ISO A4 white medium |
| iso-a4-coloured | Specifies the ISO A4 coloured medium |
| iso-a4-transparent | Specifies the ISO A4 transparent medium |
| iso-a3-white | Specifies the ISO A3 white medium |
| iso-a3-coloured | Specifies the ISO A3 coloured medium |
| iso-a5-white | Specifies the ISO A5 white medium |
| iso-a5-coloured | Specifies the ISO A5 coloured medium |

| | |
|-----------------|--------------------------------------|
| iso-b4-white | Specifies the ISO B4 white medium |
| iso-b4-coloured | Specifies the ISO B4 coloured medium |
| iso-b5-white | Specifies the ISO B5 white medium |
| iso-b5-coloured | Specifies the ISO B5 coloured medium |
| jis-b4-white | Specifies the JIS B4 white medium |
| jis-b4-coloured | Specifies the JIS B4 coloured medium |
| jis-b5-white | Specifies the JIS B5 white medium |
| jis-b5-coloured | Specifies the JIS B5 coloured medium |

1171

1172 The following standard values are defined for North American
1173 media:

| | |
|-----------------------|--|
| na-letter white | Specifies the North American letter white medium |
| na-letter coloured | Specifies the North American letter coloured medium |
| na-letter transparent | Specifies the North American letter transparent medium |
| na-legal white | Specifies the North American legal white medium |
| na-legal coloured | Specifies the North American legal coloured medium |

1174

1175 The following standard values are defined for envelopes:

| | |
|------------------------------|---|
| iso-b4-envelope | Specifies the ISO B4 envelope medium |
| iso-b5-envelope | Specifies the ISO B5 envelope medium |
| iso-c3-envelope | Specifies the ISO C3 envelope medium |
| iso-c4-envelope | Specifies the ISO C4 envelope medium |
| iso-c5-envelope | Specifies the ISO C5 envelope medium |
| iso-c6-envelope | Specifies the ISO C6 envelope medium |
| iso-designated-long-envelope | Specifies the ISO Designated Long envelope medium |
| na-10x13-envelope | Specifies the North American 10x13 envelope medium |
| na-9x12-envelope | Specifies the North American 9x12 envelope medium |
| monarch-envelope | Specifies the Monarch envelope |
| na-number-10-envelope | Specifies the North American number 10 business envelope medium |
| na-7x9-envelope | Specifies the North American 7x9 inch envelope |
| na-9x11-envelope | Specifies the North American 9x11 inch envelope |
| na-10x14-envelope | Specifies the North American 10x14 inch envelope |
| na-number-9-envelope | Specifies the North American number 9 business envelope |
| na-6x9-envelope | Specifies the North American 6x9 inch envelope |
| na-10x15-envelope | Specifies the North American 10x15 inch envelope |

1176

1177 The following standard values are defined for the less commonly
1178 used media (white-only):

| | |
|-----------------|--------------------------------------|
| executive-white | Specifies the white executive medium |
| folio-white | Specifies the folio white medium |
| invoice-white | Specifies the white invoice medium |
| ledger-white | Specifies the white ledger medium |
| quarto-white | Specifies the white quarto medium |
| iso-a0-white | Specifies the ISO A0 white medium |
| iso-a1-white | Specifies the ISO A1 white medium |
| iso-a2-white | Specifies the ISO A2 white medium |
| iso-a6-white | Specifies the ISO A6 white medium |
| iso-a7-white | Specifies the ISO A7 white medium |
| iso-a8-white | Specifies the ISO A8 white medium |
| iso-a9-white | Specifies the ISO A9 white medium |
| iso-l0-white | Specifies the ISO A10 white medium |
| iso-b0-white | Specifies the ISO B0 white medium |
| iso-b1-white | Specifies the ISO B1 white medium |
| iso-b2-white | Specifies the ISO B2 white medium |
| iso-b3-white | Specifies the ISO B3 white medium |
| iso-b6-white | Specifies the ISO B6 white medium |
| iso-b7-white | Specifies the ISO B7 white medium |
| iso-b8-white | Specifies the ISO B8 white medium |
| iso-b9-white | Specifies the ISO B9 white medium |
| iso-b10-white | Specifies the ISO B10 white medium |
| jis-b0-white | Specifies the JIS B0 white medium |
| jis-b1-white | Specifies the JIS B1 white medium |
| jis-b2-white | Specifies the JIS B2 white medium |
| jis-b3-white | Specifies the JIS B3 white medium |
| jis-b6-white | Specifies the JIS B6 white medium |
| jis-b7-white | Specifies the JIS B7 white medium |
| jis-b8-white | Specifies the JIS B8 white medium |
| jis-b9-white | Specifies the JIS B9 white medium |
| jis-b10-white | Specifies the JIS B10 white medium |

1179

1180 The following standard values are defined for engineering media:

| | |
|---|---|
| a | Specifies the engineering A size medium |
| b | Specifies the engineering B size medium |
| c | Specifies the engineering C size medium |
| d | Specifies the engineering D size medium |
| e | Specifies the engineering E size medium |

1181

1182 6.2.8.2 number-up (positiveInteger)

1183 This attribute specifies the number of source page -images to
1184 impose upon a single side of an instance of a selected medium.

1185 In general, only certain numeric values are valid for this
1186 attribute, depending upon the Printer implementation to which the
1187 print-request is directed. Typical supported values are 2 and 4.
1188 If this attribute is unspecified or has a value of 1, then the
1189 Printer does not apply any number-up transformation to the pages.

1190 This attribute primarily controls the translation, scaling and
 1191 rotation of page images, but a site may choose to add
 1192 embellishments, such as borders to each logical page.

1193 ISSUE: should there be a separate attribute to control
 1194 embellishments, especially for the 1-up case?

1195 6.2.8.3 finishing (type2Enum)

1196 This attribute identifies the finishing operation that the Printer
 1197 should apply to each copy of the printed document. Examples
 1198 include stapling, saddle -stitching, hole -drilling, binding with
 1199 tape, etc.

1200 Standard values for this attribute are:

| | |
|---------------------|--|
| staple | This indicates that staples are to be used to bind the document. The exact number and placement of the staples is site-defined; other finishing object attributes may be included to provide this information. |
| staple-top-left | This indicates that one or more staples should be placed on the top left corner of the document |
| staple-bottom-left | This indicates that one or more staples should be placed on the bottom left corner of the document |
| staple-top-right | This indicates that one or more staples should be placed on the top right corner of the document |
| staple-bottom-right | This indicates that one or more staples should be placed on the bottom right corner of the document |
| saddle-stitch | This indicates that one or more staples (wire stitches) are to be used to bind the document along the middle fold. The exact number and placement of the stitches is site-defined. |
| edge-stitch | This indicates that one or more staples (wire stitches) are to be used to bind the document along one edge. The exact number and placement of the staples is site-defined. |
| punch | This indicates that holes are required in the finished document. The exact number and placement of the holes is site-defined. The punch specification may be satisfied (in a site- and implementation-specific manner) either by drilling/punching, or by substituting predrilled media. |
| cover | This value is specified when it is desired to select a non-printed (or pre-printed) cover for the document. This does not supplant the specification of a printed cover (on cover stock medium) by the document itself. |
| bind | This indicates that a binding is to be applied to the document; the type and placement of the binding is site-defined. |

1201 none Perform no finishing. See 9.1.2

1202 6.2.8.4 sides (type2Enum)

1203 This attribute specifies whether the document should be printed in
 1204 one of three ways: 1-sided (simplex), 2-sided-long-binding-edge
 1205 (duplex), 2-sided-short-binding-edge (tumble).

1206 The standard values are: 1-sided, 2-sided-long-edge, 2-sided-
 1207 short-edge.

1208 6.2.8.5 copies (positiveInteger)

1209 This attribute specifies the number of copies of the job to be
 1210 printed. If this attribute is unspecified, its default value is 1
 1211 copy.

1212 6.2.8.6 printer-resolution-select (positiveIntegerCross)

1213 This attribute specifies the resolution that the Printer should
 1214 use.

1215 The syntax allows a single integer to specify the resolution or a
 1216 pair of integers to specify the resolution when the x and y
 1217 dimensions differ. When two integers are specified, the first is
 1218 in the x direction, ie., in the direction fo the shortest
 1219 dimension of the medium, so that the value is independent of
 1220 whether the printer feeds long edge or short edge first..

1221 6.2.8.7 print-quality (type2Enum)

1222 This attribute specifies the print quality that the Printer should
 1223 use.

1224 The standard values are:

| | |
|-------------------|---|
| 1225 draft | Lowest quality available on the printer |
| 1226 normal | Normal or intermediate quality on the printer |
| 1227 high | Highest quality available on the printer |

1228
 1229

1230 6.2.8.8 page-select (positiveIntegerRange)

1231 This attribute specifies the pages in the document that the
 1232 Printer shall use. This attribute is unlikely to be useful for
 1233 jobs with more than one document or in Job Templates. If this
 1234 attribute is unspecified, then the Printer prints all pages in a
 1235 document.

1236 6.2.8.9 files-are-one-document (Boolean)

1237 This attribute is relevant only if a job consists of two or more
 1238 documents. It controls finishing operations, job-sheet placement,
 1239 and the order of documents when the copies attribute exceeds 1.

1240 If the files for the job are a and b and this attribute is **true**,
 1241 then files a and b are treated as a single document for finishing

1242 operations. Also, there will be no slip sheets between files a and
1243 b. If more than one copy is made, the ordering must be a, b, a,
1244 b, The attribute **files-are-interleaved** is ignored.

1245 If the files for the job are a and b and this attribute is **false**
1246 **or unspecified**, then each file is treated as a single document for
1247 finishing operations. Also, a client may specify that a slip sheet
1248 be between files a and b. If more than one copy is made, and the
1249 attribute **files-are-interleaved** false or unspecified, the ordering
1250 is a, a, b, b, If more than one copy is made, and the attribute
1251 **files-are-interleaved** true, the ordering is a, b, a, b,

1252 6.2.8.10 files-are-interleaved (Boolean)

1253 This attribute is used in conjunction with **files-are-one-document**
1254 (q.v.).

1255 ISSUE: Should the files-are-one-document and files-are-interleaved
1256 be combined into a single enum attribute?

1257 6.2.9 Attributes for Conversion of Text Files (Set by Client/End
1258 User)

1259 The client shall specify these attributes to control formatting
1260 for text documents or HTML documents. If the client does not
1261 specify any of these attributes, a Printer shall use its own
1262 defaults.

1263 A client need not specify these attributes for other types of
1264 documents, such as PostScript or PCL.

1265 6.2.9.1 width (cardinalUnits)

1266 This attribute specifies the media width for the document in
1267 characters.

1268 6.2.9.2 length (cardinalUnits)

1269 This attribute specifies the media length for the document in
1270 characters.

1271 6.2.9.3 left-margin (cardinalUnits)

1272 This attribute specifies the left-margin for the document in
1273 characters.

1274 6.2.9.4 right-margin (cardinalUnits)

1275 This attribute specifies the right-margin for the document in
1276 characters.

1277 6.2.9.5 top-margin (cardinalUnits)

1278 This attribute specifies the top-margin for the document in lines.

1279 6.2.9.6 bottom-margin (cardinalUnits)

1280 This attribute specifies the bottom-margin for the document in
1281 lines.

1282 6.2.9.7 repeated-tab-stops (cardinalUnits)

1283 This attribute specifies the tab stops for the document in
1284 characters.

1285 6.2.9.8 header-text (string)

1286 This attribute specifies the header text for the document.

1287 6.2.9.9 footer-text (string)

1288 This attribute specifies the footer text for the document.

1289 6.2.9.10 font-size (cardinalUnits)

1290 This attribute specifies the font-size in points for text in the
1291 document. The value of this attribute affects the size of the
1292 other text attributes.

1293 If this attribute is omitted, the Printer shall assume a value of
1294 10. A value of 10 with a fixed pitch font, shall produce 12
1295 characters per inch in the horizontal direction and with 6 lines
1296 per inch in the vertical direction.

1297 6.2.9.11 number-pages (Boolean)

1298 This attribute specifies that the pages should be numbered in the
1299 document.

1300 default-font (string) This attribute specifies the font to use for
1301 all text in the document.

1302 6.2.9.12 default-code-set (type3Enum)

1303 This attribute specifies the code-set in which the document is
1304 encoded.

1305 6.2.9.13 content-orientation (type2Enum)

1306 This attribute specifies the orientation of the document.

1307 The standard values are

| | |
|----------|--|
| portrait | The page orientation such that the sides are longer than the top when the page is held in the intended human reading orientation |
|----------|--|

landscape The page orientation such that the sides are shorter than the top when the page is held in the intended human readable orientation. Landscape is defined to be a rotation of the page by +90 degrees with respect to the medium (i.e. anti-clockwise) from the portrait orientation
 NOTE - The +90 direction was chosen because simple finishing on the long edge is the same edge whether portrait or landscape

reverse-
portrait The page orientation defined to be a rotation of 180 degrees with respect to portrait

reverse-
landscape The page orientation defined to be a rotation of 180 degrees with respect to landscape. Landscape is defined to be a rotation of the page by -90 degrees with respect to the medium (i.e. clockwise) from the portrait orientation
 NOTE - Reverse-landscape was added because some applications rotate landscape -90 degrees from portrait, rather than +90 degrees.

1308

1309 6.2.10 Job Resource Attributes (Set by the program that produces or
 1310 senses the PDL)

1311 A program described below shall add these attributes, which
 1312 describe the resources needed to print the job.

1313 A Printer may use these attributes to validate and schedule the
 1314 print-job without interpreting the contents of the document. This
 1315 provides the opportunity for a Printer to support a broad set of
 1316 document formats yet still support fast efficient scheduling and
 1317 validation of each job.

1318 The client/end user shall not specify these attributes. Instead,
 1319 it is the duty of the program that translates the document to the
 1320 printer's PDL (or analyzes it) to add these attributes and their
 1321 values to the job. Such a program may execute at a number of
 1322 different points in time:

1324 1. The program produces a final form document and stores it in
 1325 a file before the end-user submits the print job.

1326 2. The program produces a final form document data stream when
 1327 the end-user specifies "Print" to the application program
 1328 (Windows GDI driver).

1329 3. The program translates a revisable or final form document
 1330 into a PDL that the printer understands.

1331 If any of these attributes is unspecified, the Printer shall
 1332 assume that the all resources required by the document of the type
 1333 specified by the missing attributes are ready, ie., are available
 1334 to the Printer and/or output device without human intervention.
 1335 These attributes may be unspecified if the translation program

1336 fails to provides such values, or if no translation occurs (e.g.
1337 the document is a PostScript document.

1338 Note: The Printer does not use these attributes during the actual
1339 printing of a document.

1340 Note: these attributes allow more than one value wherever it is
1341 possible for a job to specify more than one value of the
1342 corresponding job attribute, possibly by embedded instructions.

1343 The client may specify these attributes in: Get-Attributes and
1344 Get-Jobs.

1345 See the section on job production attributes for an explanation of
1346 how the job resource attributes differ from the job production
1347 attributes.

1348 6.2.10.1 document-format-used (1#type2Format)

1349 This attribute identifies the document formats needed to print the
1350 document(s) in this job.

1351 A format consists of two elements, a name and a version. The
1352 latter element is optional.

1353 The syntax is for type2Format:

1354 name ["/" version]

1355 Examples include: PostScript, PostScript/2.0 and PCL/5e

1356 ISSUE: do we want the version to be optional?

1357 6.2.10.2 fonts-used (1#string)

1358 This attribute identifies the font resources used in the
1359 document(s) in the job.

1360 6.2.10.3 code-sets-used (1#type3Enum)

1361 This attribute identifies the code-sets used in the document(s) in
1362 the Job. This attribute is relevant only for files that are not in
1363 ASCII, such as text files and possibly PCL files. PostScript files
1364 are always ASCII. Normally there is at most 1 code-set.

1365 Standard values are defined in the section specifying the default-
1366 code-set attribute.

1367 6.2.10.4 media-used (1#type2Enum)

1368 This attribute identifies the media, media-sizes, input-trays or
1369 electronic forms needed to print the document(s) in the job.

1370 Standard values for this attribute are defined in the section
1371 specifying the medium-select attribute.

1372

1373 6.2.10.5 sides-used (type2Enum)

1374 This attribute specifies whether a job needs 1-sided, 2-sided-
1375 long-binding--edge, or 2-sided-short-binding-edge printing.

1376 Standard values for this attribute are defined in the section
1377 specifying the sides attribute.

1378 6.2.10.6 print-quality-used (type2Enum)

1379 This attribute specifies what print quality the job needs.

1380 Standard values for this attribute are defined in the section
1381 specifying the print-quality attribute.

1382 6.2.10.7 finishing-used (type2Enum)

1383 This attribute specifies what finishing the job needs.

1384 Standard values for this attribute are defined in the section
1385 specifying the finishing attribute.

1386 6.2.10.8 printer-resolution-used (positiveIntegerCrossState)

1387 This attribute specifies what resolution the job needs.

1388 6.2.10.9 total-job-octets (positiveInteger)

1389 This attribute specifies the total size of the job in octets. This
1390 attribute is the first of three that a translation program can use
1391 to specify the size of a job.

1392 6.2.10.10 job-impression-count (positiveInteger)

1393 This attribute specifies the total size of the job in impressions.

1394 6.2.10.11 job-media-sheet-count (positiveInteger)

1395 This attribute specifies the total size of the job in media-
1396 sheets.

1397 6.2.11 Number of Documents (Set by Client)

1398 This group contains a single attribute which specifies the number
1399 of documents in the job.

1400 The client shall specify this attribute in Print and may specify
1401 this attribute in: Get-Attributes and Get-Jobs.

1402 6.2.11.1 number-of-documents (positiveInteger)

1403 This attribute specifies the number of documents in the job. Each
1404 document shall contain its own set of document content attributes
1405 described below.

1406 6.2.12 Document Data (Set by a Client/End User)

1407 This group of attributes describes the document data for the job.
1408 These attributes also include the document data or reference it.

1409 All job attributes in other sections of this document occur only
1410 once per job and apply to all documents in a job.

1411 The client may specify document -data attributes in Print. The
1412 client must specify either the document-URL or document-content in
1413 Print.

1414 Except for document-content, the client may specify document -data
1415 attributes in: Get-Attributes, and Get-Jobs.

1416 6.2.12.1 document-format (type2Format)

1417 This attribute identifies the document format of this document.

1418 If the client does not specify this attribute, then the Printer
1419 shall attempt to determine the format in order to decide if the
1420 document data needs to be translated.

1421 ISSUE: do we want the version to be optional?

1422 6.2.12.2 document-name (name)

1423 This attribute contains the name of the document used by the
1424 client to initially identify the document.

1425 6.2.12.3 document-URL (name)

1426 This attribute contains the URL of the document if the client
1427 specified the document with a URL.

1428 If this attribute is specified, then document-content shall be
1429 unspecified.

1430 6.2.12.4 document-content (octetString)

1431 This attribute contains the actual contents of the document.

1432 If this attribute is specified, then document-URL shall be
1433 unspecified.

1434 This attribute shall be used during the transmission of the Print
1435 operation over a network. A Printer shall save the document data
1436 to a file and reference it with the document-URL or document-path
1437 attribute. A Get-Attribute or Get-Jobs operation shall always find
1438 that this attribute is unspecified.

1439 6.3 Operation Attributes (Set by Client)

1440 NOTE: These attributes have just been introduced and they are not
1441 as stable as the attributes in the other sections. Some work is
1442 still needed to show the relationship between these attributes,

1443 job attributes, printer attributes, and authentication and
1444 authorization.

1445 The client shall set these attributes and associate them with an
1446 operation rather than an object.

1447 It is intended that a client program rather than an end-user has
1448 control over the setting of these values so that they cannot be
1449 easily forged.

1450 6.3.1 operation-locale (type3Locale)

1451 This attribute identifies the locale of the client. The Printer
1452 uses this attribute to determine the locale of messages in the
1453 result of the operation or in errors returned by the operation.

1454 The standard values are defined in the section on the job-locale
1455 attribute.

1456 If an operation does not specify this attribute, the Printer shall
1457 assume that the operation has the same locale as the Printer.

1458 6.3.2 operation-notification-address (name)

1459 This attribute identifies the email-address of the client. The
1460 Printer uses this attribute to determine the email address for any
1461 notifications that occur in the Printer.

1462 ISSUE: can this address be determined from the next two
1463 attributes: operation-user-name and operation-host-name?

1464 6.3.3 operation-user-name (name)

1465 This attribute identifies the most authenticated end user name
1466 that the client can supply. This name identifies the end user
1467 performing the operation.

1468 This value shall be set by the system rather than the end-user in
1469 order to minimize the chance of forgery.

1470 6.3.4 operation-host-name (name)

1471 This attribute identifies the most authenticated host name that
1472 the client can supply. This name identifies the host from which
1473 the operation comes.

1474 This value shall be set by the system rather than the end-user in
1475 order to minimize the chance of forgery.

1476 6.4 Printer Attributes (Set by the Administrator)

1477 A printer object may be realized in either a Print Server or
1478 Output Device. Note: How these attribute are set by an
1479 Administrator is outside the scope of this specification.

1480 A Printer Object in an Output Device contains a set of printer
1481 object attributes that represent an Output Device capable of
1482 rendering a document in visible form. Examples include electronic

1483 and electro-mechanical printers such as laser printers, ink -jet
 1484 printers, and various kinds of impact printers, but may include
 1485 other types of output devices such as microfiche imagers and
 1486 plotters as well.

1487 A Printer Object in a Print Server that supplies queuing,
 1488 spooling, and scheduling for an Output device that does not queue
 1489 or spool.

1490 A Printer Object in a Print Server contains a set of printer
 1491 object attributes that are the union of the Printer objects in the
 1492 downstream Output Devices. This object extends the capabilities
 1493 of an Output Device. For example, an administrator might define a
 1494 single Print Server to represent all of the Output Devices of the
 1495 same type and capability in a single location, associated with a
 1496 particular server. A end user would normally send a print -job to
 1497 a Print Server , and allow the Print Server to assign the job to a
 1498 particular Output Device based on the relative load and
 1499 availability of the printers under its control, thus providing a
 1500 load balancing service. However, nothing precludes an
 1501 administrator from configuring a print system so that a end user
 1502 can send a print -job directly to an Output Device .

1503 A Print Server, in the most common case, controls exactly one
 1504 downstream Output Device. The Print Server's Printer object has
 1505 attributes whose values are the same as those of the Printer
 1506 object in the downstream Output Device.

1507 The attributes defined in this section provide information about
 1508 a particular Printer.

1509 6.4.1 printer-name (name)

1510 This attribute uniquely identifies the printer on its host.

1511 6.4.2 printer-location (string)

1512 This attribute identifies the location of this printer.

1513 6.4.3 printer-model (string)

1514 This attribute identifies the make and model of the printer.

1515 6.4.4 printer-types (type2Enum)

1516 This attribute identifies the marking technology of the printer.

1517 The standard value for this attribute are the descriptive names
 1518 specified by ISO DPA which have corresponding enum symbolic and
 1519 numeric values assigned by the Printer MIB (RFC 1759).. These
 1520 standard values are:

| | |
|---------------------------|--------------------------------|
| other | Other than the standard values |
| unknown | Unknown printer type |
| electrophotographic-LED | electrophotographic LED |
| electrophotographic-laser | electrophotographic laser |

| | |
|-------------------------|---------------------------------|
| electrophotographic- | other electrophotographic |
| other | |
| impact-moving-head-dot- | 9-pin impact moving head dot |
| matrix-9-pin | matrix |
| impact-moving-head-dot- | 24-pin impact moving head dot |
| matrix-24-pin | matrix |
| impact-moving-head-dot- | neither 9-pin nor 24-pin moving |
| matrix-other | head dot matrix |
| impact-moving-head- | fully formed impact moving head |
| fully-formed | |
| impact-band | impact band |
| impact-other | impact other |
| inkjet-aqueous | aqueous inkjet |
| inkjet-solid | solid inkjet |
| inkjet-other | other inkjet |
| pen | pen |
| thermal-transfer | thermal transfer |
| thermal-sensitive | thermal sensitive |
| thermal-diffusion | thermal diffusion |
| thermal-other | other thermal |
| electro-erosion | electro-erosion |
| electro-static | electro-static |
| photographic-microfiche | photographic microfiche |
| photographic- | photographic imagesetter |
| imagesetter | |
| photographic-other | other photographic |
| ion-deposition | ion deposition |
| E-beam | E-beam |
| typesetter | typesetter |

1521

1522 **ISSUE:** Should they be from the printer MIB instead. In the printer
 1523 MIB hyphens do not exist. Instead the first letter after a hyphen
 1524 is upper case. THIS ISSUE IS CLOSED: We will use the xxx-yyy-zzz
 1525 format rather than the xxxYyyZzz format.

1526 6.4.5 printer-state (type1Enum)

1527 This attribute identifies the current state of the printer. The
 1528 protocol support all values for printer states, however a Printer
 1529 shall only generate the printer states which are appropriate for
 1530 the particular implementation.

1531 The following standard values are defined:

| | |
|-----------------|---|
| unknown | The printer state is not known, or is indeterminate, or is not returned by the operation |
| idle | The printer is ready to accept jobs, but none have been scheduled on it. |
| printing | The printer is currently printing a job |
| needs-attention | The printer needs human attention (no special skills required). This state typically includes adding paper, clearing a jam, changing the medium, etc. |
| paused | The operator has (temporarily) paused the printer, by means outside the scope of this part of ISO/IEC 10175. |

| | |
|-----------------------|---|
| shutdown | The printer has been taken out of service, (for a long time), whether for repairs or others reasons. The printer's message generic attribute may be used to record a reason and estimated time for return to service |
| job-start-wait | The currently processing job was started with the job-start-wait attribute set, and is awaiting operator intervention or time-out. |
| job-end-wait | The currently processing job was started with the job-end-wait attribute set, and is awaiting operator intervention or time-out. |
| job-password-wait | The currently processing job was started with the job-password attribute set, and is awaiting the operator or user to enter the password supplied by the job-password attribute. |
| needs-key-operator | The printer needs the attention of a key operator. Key operator functions are printer-specific, but typically include adding toner or developer, or attending to a hardware fault. |
| connecting-to-printer | The server has scheduled a job on the printer and is in the process of connecting to a shared network printer (and may not be able to actually start printing the job for an arbitrarily long time depending on the usage of the printer by other servers). |
| timed-out | The server was able to connect to the printer (or is always connected), but was unable to get a response from the printer in the time specified by the printer's printer-timeout-period attribute. |

1532

1533 6.4.6 printer-state-message (string)

1534 This attributes specifies a message that gives further information
 1535 about the current printer state. .

1536 6.4.7 message (string)

1537 This attribute provides a message from an operator, system
 1538 administrator or 'intelligent' process to indicate to the end user
 1539 information or status of the printer, such as why it is
 1540 unavailable or when it is expected to be available. .

1541 6.4.8 locale (type3Locale)

1542 This attribute specifies the locale that the Printer operates in.

1543 The standard values are defined in the section on the job-locale
 1544 attribute.

1545 6.4.9 notification-events (#type2Enum)

1546 This attribute specifies the events on whose occurrence the
1547 Printer should notify those addresses specified by the
1548 notification-addresses attribute.

1549 If the attribute is unspecified or empty, the Printer does not
1550 perform notification, though the Printer still checks the jobs'
1551 notification-events attribute.

1552 In this attribute, job-problem and printer-problem have the same
1553 meaning.

1554 The standard values are defined in the section on the job's
1555 notification-events attribute.

1556 6.4.10 notification-addresses (#name)

1557 This attribute specifies the email addresses to which the Printer
1558 should send messages when events specified by the notification-
1559 events attribute occur.

1560 If the attribute is unspecified or empty, the Printer does not
1561 perform notification, though the Printer still checks the jobs'
1562 notification-events attribute.

1563 6.4.11 end-user-acl (#name)

1564 This attribute specifies the end users who are allowed to print on
1565 the Printer.

1566 If the attribute is unspecified or empty, the Printer allows
1567 anyone to print.

1568 ISSUE: this does not fully solve the internet authorization
1569 problem because of authentication issues.

1570 6.4.12 maximum-printer-speed (positiveIntegerUnits)

1571 This attribute indicates the maximum printer speed of the Printer.
1572 A job cannot control a Printer's speed, but a Printer Browser can
1573 use printer speed as a criteria.

1574 The standard units are a type2Enum and are: ppm, ipm, spm, lpm,
1575 cps.

1576 6.4.13 fonts-substitutions (#stringPair)

1577 This attribute specifies an appropriate substitute for a font that
1578 is advertised as supported in the fonts-supported attribute, even
1579 though the Printer doesn't actually have the font available.

1580 This attribute consists of a set of font pairs: a font name and
1581 the font to use instead.

1582 6.4.14 fonts-supported (1#stringState)

1583 This attribute identifies the font resources supported by this
1584 printer and indicates the state of readiness for each font.

1585 The standard names are defined in the section on default-font.

1586 Each item in the list contains the pair consisting of a font name
1587 and a state indicating the font's readiness state.

1588 6.4.15 media-supported (1#nameState)

1589 This attribute identifies the media, media-sizes, input trays, and
1590 electronic forms supported by this printer, and indicates the
1591 state of readiness for each medium resource.

1592 There may be just two states: ready and needs-installing, or there
1593 may be a third state: needs-purchasing.

1594 The standard names are defined in the section on the section on
1595 the medium-select.

1596 6.4.16 document-formats-supported (1#type2FormatState)

1597 This attribute identifies the document -formats, including the
1598 document -format -versions, supported by the Printer. This set
1599 includes both the formats that are native to the Printer and
1600 those formats that the Printer can translate to one that is
1601 native to the Printer. From the client's point of view, this set
1602 contains all formats in which documents can be submitted to this
1603 Printer.

1604 Proprietary document format identifiers, and versions are assigned
1605 by the owners of those formats.

1606 The state of readiness for each format is also included, though
1607 all formats should normally always be ready.

1608 6.4.17 numbers-up-supported (1#positiveIntegerState)

1609 This attribute identifies the number -up values supported by this
1610 printer..

1611 The state of readiness for each number-up value is also included,
1612 though all number-up conversions should always be ready.

1613 6.4.18 finishings-supported (#type2EnumState)

1614 This attribute identifies the finishing operations supported by
1615 this Printer and states of readiness for each finishing.

1616 The standard finishing objects are defined in the section on the
1617 finishing Job attribute.

1618 6.4.19 sides-supported (1#type2EnumState)

1619 This attribute indicates the values of the sides attribute
1620 supported by this printer and the states of readiness of each
1621 value.

1622 The standard values are defined in the section on the sides
1623 attribute.

1624 6.4.20 print-qualities-supported (1#type2EnumState)

1625 This attribute indicates the values of the printer-quality
1626 attribute supported by this printer and the states of readiness
1627 for each print-quality value.

1628 The standard values are defined in the printer-quality attribute.

1629 6.4.21 printer-resolutions-supported (1#positiveIntegerCrossState)

1630 This attribute indicates the values of the printer-resolution-
1631 select attribute supported by this printer and their states of
1632 readiness.

1633 The state of readiness for each printer resolution is also
1634 included, though normally all printer-resolutions should always be
1635 ready.

1636 The syntax is discussed in the section on the printer-resolution-
1637 select attribute.

1638 6.4.22 code-sets-supported (1#type3EnumState)

1639 This attribute indicates the values of the default-code-set
1640 attribute supported by this printer and the states of readiness
1641 for each code-set.

1642 The standard values are defined in the default-code-set attribute.

1643 6.4.23 off-peak-times-supported (#type3EnumState)

1644 This attribute indicates the values of the job-print-off-peak
1645 attribute supported by this printer and the states of readiness
1646 for each value.

1647 If this attribute is unspecified, then the Printer has no off-peak
1648 periods.

1649 The standard values are defined in the section on the job-print-
1650 off-peak attribute.

1651 Note: this document does not define how an administrator
1652 associates the off-peak names with actual time periods.

1653 6.4.24 events-supported (#type2EnumState)

1654 This attribute indicates the values of the job and printer
1655 notification-events attribute supported by this Printer and the
1656 states of readiness for each value.

1657 If this attribute is unspecified, then the Printer does not
1658 support notification.

1659 The standard values are defined in the section on the
1660 notification-events attribute.

1661 6.4.25 locales-supported (1#type3LocaleState)

1662 This attribute indicates the values of the job-locale attribute
1663 supported by this Printer and the states of readiness for each
1664 value.

1665 The standard values are defined in the section on the job-locale
1666 attribute.

1667 6.4.26 job-sheets-supported (#type3EnumState)

1668 This attribute identifies the job-sheet values supported by this
1669 printer, and the state of readiness for each job-sheet.

1670 To allow no job sheets, the system administrator shall include the
1671 value none as a value for this attribute. The client specifies
1672 that there are no job sheets by using the value none as the value
1673 of the job-sheets attribute.

1674 If the job-sheets attribute is not specified or contains a value
1675 which the Printer does not support, then the server shall select
1676 from among the values of this attribute. The server shall not
1677 select the value none unless it is the only value specified for
1678 the job-sheets-supported attribute.

1679 NOTE - It is preferable for the server to produce some job
1680 jobsheet, even if not the desired one, rather than produce none at
1681 all or reject the job.

1682 6.4.27 maximum-copies (positiveInteger)
1683

1684 This attribute indicates the maximum number of copies of a
1685 document that can be rendered by this printer in a single
1686 print-job.

1687 If the attribute is unspecified or has a value of 0, there is no
1688 limit on the maximum number of copies for this Printer.

1689 6.4.28 maximum-job-octets (positiveInteger)

1690 This attribute indicates that the Printer shall accept a job only
1691 if its size in octets is less than the value specified by this
1692 attribute.

1693 If the attribute is unspecified or has a value of 0, there is no
1694 limit on the size of a job in octets.

1695 6.4.29 maximum-impressions (positiveInteger)

1696 This attribute indicates that the Printer shall accept a job only
1697 if its size in impression is less than the value specified by this
1698 attribute.

1699 If the attribute is unspecified or has a value of 0, there is no
1700 limit on the size of a job in impressions.

1701 6.4.30 maximum-media-sheets (positiveInteger)

1702 This attribute indicates that the Printer shall accept a job only
1703 if its size in media-sheets is less than the value specified by
1704 this attribute.

1705 If the attribute is unspecified or has a value of 0, there is no
1706 limit on the size of a job in media-sheets.

1707 6.4.31 maximum-job-retention-period (deltaTime)

1708 This attribute indicates that when the Printer accepts a job, the
1709 retention period must not exceed the value of this attribute.
1710 Otherwise, the Printer sets the job's retention-period to the
1711 value of this attribute.

1712 If this attribute is unspecified, then the Printer places no limit
1713 on the retention time.

1714 ISSUE: Should this be deleted?

1715 6.4.32 maximum-end-user-priority (typeEnum)

1716 This attribute indicates that when the Printer accepts a job, the
1717 job-priority must not exceed the value of this attribute.
1718 Otherwise, the Printer sets the job's job-priority to the value of
1719 this attribute.

1720 If this attribute is unspecified, then the Printer places no limit
1721 on the job-priority time.

1722 The standard values are defined in the section on the job-priority
1723 attribute.

1724 6.5 Job Templates

1725 The attributes for a Job Template can be any of the Job object
1726 attributes defined in the sections:

1727 Job Sheet Attributes
1728 Notification Attributes
1729 Job Scheduling Attributes
1730 (except job-print-after)
1731 Job Production Attributes
1732 (except page-select)
1733 Attributes for Conversion of Text Files

6.6 Conformance

A conforming implementation shall implement all operations, objects and attributes defined in this document. IPP is explicitly designed to be extensible. This means that in addition to the attributes defined in this specification, specific implementation instances may support not only the basic protocol as defined in this specification, but might add vendor specific extensions.

Also, for the core set of attributes listed in this specification, it is not required that a conforming server support all (standard) values of all supported attributes. For example, it is not required that a printer implement all finishing methods indicated by the standard values.

The explicit requirement of the term "supported", with respect to one of the attributes that deal with printer functions or resources, is that the server shall recognize the attribute and those values that are supported, and shall be able to respond to a query about which values that printer does, in fact, support.

Additional attributes can be proposed to be registered by going through the type 2 enum process which will register their specification after approval with IANA.

7. Security Considerations

This protocol does not identify any new authentication mechanisms. The authentication mechanisms built into HTTP (such as SSL and SHTTP) are recommended.

This protocol does define a simple authorization mechanism by introducing the "end-user-acl" attribute as part of the Printer object. This ACL attribute is a multi-valued list of all of the authenticated names of end-users. This protocol does not specify what the domain is for names in this ACL attribute.

ISSUE: Will it always be possible for a Printer to obtain a meaningful authenticated name that the Printer can match against the end-user-acl, or will some other mechanism be necessary, such as a password?

8. References

- [1] Smith, R., Wright, F., Hastings, T., Zilles, S., and Gyllenskog, J., "Printer MIB", RFC 1759, March 1995.
- [2] Berners-Lee, T., Fielding, R., and Nielsen, H., "Hypertext Transfer Protocol - HTTP/1.0", RFC 1945, August 1995.
- [3] Crocker, D., "Standard for the Format of ARPA Internet Text Messages", RFC 822, August 1982.
- [4] Postel, J., "Instructions to RFC Authors", RFC 1543, October 1993.

- [5] ISO/IEC 10175 Document Printing Application (DPA), Final, June 1996.
- [6] Herriot, R. (editor), X/Open A Printing System Interoperability Specification (PSIS), August 1995.
- [7] Kirk, M. (editor), POSIX System Administration -- Part 4: Printing Interfaces, POSIX 1387.4 D8, 1994.
- [8] Borenstein, N., and Freed, N., "MIME (Multi-purpose Internet Mail Extensions) Part One: Mechanism for Specifying and Describing the Format of Internet Message Bodies", RFC 1521, September, 1993.
- [9] Braden, S., "Requirements for Internet Hosts - Application and Support", FRC 1123, October, 1989,
- [10] McLaughlin, L. III, (editor), "Line Printer Daemon Protocol" RFC 1179, August 1990.

9. Author's Address

Scott A. Isaacson
Novell, Inc.
122 E 1700 S
Provo, UT 84606

Phone: 801-861-7366
Fax: 801-861-4025
EMail: scott_isaacson@novell.com

Tom Hastings
Xerox Corporation
701 S. Aviation Blvd.
El Segundo, CA 90245

Phone: 310-333-6413
Fax: 310-333-5514
EMail: hastings@cpl0.es.xerox.com

Robert Herriot
Sun Microsystems Inc.
2550 Garcia Ave., MPK-17
Mountain View, CA 94043

Phone: 415-786-8995
Fax: 415-786-7077
Email: robert.herriot@eng.sun.com

Roger deBry
HUC/003G
IBM Corporation
P.O. Box 1900
Boulder, CO 80301-9191

Phone: (303) 924-4080
Fax: (303) 924-9889
Email: debry@vnet.ibm.com

1839

1840 Other Contributors

1841 Devon Taylor, Novell, Inc.

1842 Mike MacKay, Novell, Inc.

1843 Peter Zehler, Xerox, Corp.

1844 Keith Carter, IBM Corporation

1845 Carl-Uno Manros, Xerox, Corp.

1846 <add the list of IPP attendees and participants>

1847

1848

1849 10. Appendix A: Extended Operations

1850 The following symbols are used in the tables below:

1851 P perform the operation directly
 1852 PF perform the operation; forward to Output Device sometimes
 1853 UA unsupported in an Output Device unless it supports queuing
 1854 U unsupported operation

1855 Lower priority end user operations are:

| Operation | Print Server | Output Device |
|----------------|--------------|---------------|
| - Modify Job | P | UA |
| - Resubmit Job | P | UA |

1857

1858 Management operations are:

| Operation | Print Server | Output Device |
|--------------------|--------------|---------------|
| - Clean Queue | PF | UA |
| - Disallow Queuing | P | UA |
| - Allow Queuing | P | UA |
| - Pause Printing | P | P |
| - Resume Printing | P | P |
| - Promote Job | PF | UA |
| - Shutdown Printer | P | P |
| - Startup Printer | P | P |
| - Create Printer | P | U |
| - Delete Printer | P | U |
| - Set Attribute | P | P |

1859

1860

1861 10.1 Modify Job Operation

1862 10.1.1 Modify Job Argument

1863 The following abstract data types are part of the Modify Job
 1864 Argument (the attributes that can be modified may be severely
 1865 restricted):

| | |
|------------------|--|
| | |
| Job Id | Which job to modify. |
| | [There are no document attributes to modify.] |
| Job Attributes | Attribute set for Job attributes. Only replacement is possible; the GUI fetches the value and then sets a new one. |
| | |
| Message | Optional Message. |
| Common Arguments | |

1866

1867 10.1.2 Modify Job Result

1868 The following abstract data types are part of the Modify Job
 1869 Result:

| | |
|---------------|----------------------------|
| Modify Status | Modify result attributes. |
| Errors | Optional Error Information |

1870
1871
1872
1873
1874
1875

NOTE: job-hold can be added as a Job attribute when Modify Job is supported as an IPP operation.

10.2 Resubmit Job Operation

1876

10.2.1 Resubmit Job Argument **Error! Bookmark not defined.**

1877
1878

The following abstract data types are part of the Resubmit Argument:

| | |
|-----------------------------|---|
| | . |
| Destination Printer Name | Optional name of the destination printer. |
| | |
| Operation | MOVE or COPY |
| Job Set | A set of jobs to move or copy. Each entry in the set has: Job Id, Document Number, Job attributes, and Document attributes. |
| Message | Optional Message |
| Common Arguments | |

1879
1880
1881

10.2.2 Resubmit Job Result

1882
1883

The following abstract data types are part of the Resubmit Job Result:

| | |
|---------------------|---|
| Resubmit Job Set | A set of jobs that were resubmitted. Each element in the set has: Old Job Id, New Job Id, and an attribute set with info about the results of the move or copy. |
| Errors | Optional Error Information |

1884
1885