

Robert Herriot (editor)
Sun Microsystems, Inc.
Tom Hastings
Xerox Corporation
Norm Jacobs
Sun Microsystems, Inc.
Jay Martin
Underscore, Inc.
November 20, 1997

Mapping between LPD and IPP Protocols draft-ietf-ipp-lpd-ipp-map-02.txt

Copyright © The Internet Society (date). All Rights Reserved.

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 the mapping between (1) the commands and operands of the "Line Printer Daemon (LPD) Protocol" specified in RFC 1179 and (2) the operations and parameters of the Internet Printing Protocol (IPP). One of the purposes of this document is to compare the functionality of the two protocols. Another purpose is to facilitate implementation of gateways between LPD and IPP.

[This document is an informational document that is not on the standards track. It is intended to help implementors of gateways between IPP and LPD. It also provides an example, which gives additional insight into IPP.](#)

WARNING: RFC 1179 was not on standards track. While RFC 1179 was intended to record existing practice, it fell short in some areas. However, this specification maps between (1) the actual current practice of RFC 1179 and (2) IPP. This document does not attempt to map the numerous divergent extensions to the LPD protocol that have been made by many implementers.

TABLE OF CONTENTS

35		
36		
37	1. Introduction.....	3
38	2. Terminology.....	3
39	3. Mapping from LPD Commands to IPP Operations	4
40	3.1 Print any waiting jobs	4
41	3.2 Receive a printer job	4
42	3.2.1 Abort job	5
43	3.2.2 Receive control file	5
44	3.2.3 Receive data file.....	6
45	3.3 Send queue state (short)	6
46	3.4 Send queue state (long).....	8
47	3.5 Remove jobs	9
48	4. Mapping of LPD Control File Lines to IPP Parameters	9
49	4.1 Required Job Functions	10
50	4.2 Optional Job Functions	10
51	4.3 Required Document Functions.....	11
52	4.4 Recommended Document Functions.....	11
53	5. Mapping from IPP operations to LPD commands	12
54	5.1 Print-Job	12
55	5.2 Print-URI	13
56	5.3 Validate-Job.....	13
57	5.4 Create-Job	13
58	5.5 Send-Document	13
59	5.6 Send-URI	13
60	5.7 Cancel-Job	14
61	5.8 Get-Attributes	14
62	5.9 Get-Jobs	15
63	6. Mapping of IPP Parameters to LPD Control File Lines	15
64	6.1 Required Job Functions	15
65	6.2 Optional Job Functions	16
66	6.3 Required Document Functions.....	16
67	7. Copyright Notice.....	17
68	8. References	17
69	9. Author's Addresses	17
70	10. Appendix A: ABNF Syntax for response of Send-queue-state (short)	18
71	11. Appendix B: ABNF Syntax for response of Send-queue-state (long).....	18
72	12. Appendix C: Unsupported LPD functions.....	19
73		

74

Mapping between the LPD and IPP Protocols

75

1. Introduction

76

77

The reader of this specification is expected to be familiar with the IPP Model and Semantics specification [ipp-mod], the IPP Protocol specification [ipp-pro], and the Line Printer Daemon (LPD) protocol specification [rfc1179] as described in RFC 1179.

78

79

80

81

82

RFC 1179 was written in 1990 in an attempt to document existing LPD protocol implementations. Since then, a number of undocumented extensions have been made by vendors to support functionality specific to their printing solutions. All of these extensions consist of additional control file commands. This document does not address any of these vendor extensions. Rather it addresses existing practice within the context of the features described by RFC 1179. Deviations of existing practice from RFC 1179 are so indicated.

83

84

85

Other LPD control file commands in RFC 1179 are obsolete. They are intended to work on "text" only formats and are inappropriate for many contemporary document formats that completely specify each page. This document does not address the support of these obsolete features.

86

87

88

89

90

In the area of document formats, also known as page description languages (PDL), RFC 1179 defines a fixed set with no capability for extension. Consequently, some new PDL's are not supported, and some of those that are supported are sufficiently unimportant now that they have not been registered for use with the Printer MIB[rfc1759] and IPP[ipp-mod] [ipp-pro], though they could be registered if desired. See the Printer MIB specification [rfc1759] and/or the IPP Model specification [ipp-mod] for instructions for registration of document-formats with IANA. IANA lists the registered document-formats as "printer languages".

91

92

93

This document addresses the protocol mapping for both directions: mapping of the LPD protocol to the IPP protocol and mapping of the IPP protocol to the LPD protocol. The former is called the "LPD-to-IPP mapper" and the latter is called the "IPP-to-LPD mapper".

94

95

This document is an informational document that is not on the standards track. It is intended to help implementors of gateways between IPP and LPD. It also provides an example, which gives additional insight into IPP.

96

2. Terminology

97

98

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [abnf].

99

100

101

RFC 1179 uses the word "command" in two contexts: for over-the-wire operations and for command file functions. This document SHALL use the word "command" for the former and the phrase "functions" for the latter. The syntax of the LPD commands is given using ABNF [abnf].

102

The following tokens are used in order to make the syntax more readable:

103

104

105

LF stands for %x0A (linefeed)

SP stands for %x20. (space)

DIGIT stands for %x30-39 ("0" to "9")

106

3. Mapping from LPD Commands to IPP Operations

107

108

This section describes the mapping from LPD commands to IPP operations. Each of the following sub-sections appear as sub-sections of section 5 of RFC 1179.

109 The following table summarizes the IPP operation that the mapper uses when it receives an LPD command. Each section below
 110 gives more detail.

LPD command	IPP operation
print-any-waiting-jobs	<i>ignore</i>
receive-a-printer-job	Print-Job or Create-Job/Send-Document
send queue state (short or long)	Get-Attributes (printer) and Get-Jobs
remove-jobs	Cancel-Job

111 **3.1 Print any waiting jobs**

112 Command syntax:

113 print-waiting-jobs = %x01 printer-name LF

114 This command causes the LPD daemon check its queue and print any waiting jobs. An IPP printer handles waiting jobs without
 115 such a nudge.

116 If the mapper receives this LPD command, it SHALL ignore it and send no IPP operation.

117 **3.2 Receive a printer job**

118 Command syntax:

119 receive-job = %x02 printer-name LF

120 The control file and data files mentioned in the following paragraphs are received via LPD sub-commands that follow this
 121 command. Their mapping to IPP commands and attributes is described later in this section.

122 The mapper maps the 'Receive a printer job' command to either:

- 123 • the Print-Job operation which includes a single data file or
- 124 • the Create-Job operation followed by one Send-Document operation for each data file.

125 If the IPP printer supports both Create-Job and Send-Document, and if a job consists of:

- 126 • a single data file, the mapper SHOULD use the Print-Job operation, but MAY use the Create-Job and Send-
 127 Document operations.
- 128 • more than one data file, the mapper SHALL use Create-Job followed by one Send-Document for each received
 129 LPD data file.

130 If the IPP printer does not support both Create-Job and Send-Document, and if a job consists of:

- 131 • a single data file, the mapper SHALL use the PrintJob operation.
- 132 • more than one data file, the mapper SHALL submit each received LPD data file as a separate Print-Job
 133 operation (thereby converting a single LPD job into multiple IPP jobs).

134 If the mapper uses Create-Job and Send-Document, it MUST send the Create-Job operation before it sends any Send-Document
 135 operations whether the LPD control file, which supplies attributes for Create-Job, arrives before or after all LPD data files.

136 NOTE: This specification does *not* specify how the mapper maps: the LPD Printer-name operand to the IPP "printer-uri"
137 parameter.

138 The following 3 sub-sections gives further details about the mapping from LPD receive-a-printer-job sub-commands. Each of
139 the following sub-sections appear as sub-sections of section 6 of RFC 1179.

140 ~~ISSUE: the mapper needs to maintain information such as the mapping of each job-number to its corresponding job-URI. It~~
141 ~~would be nice for IPP to support an "scratch-pad" attribute for the mapper to encode such information. Then it wouldn't have to~~
142 ~~maintain this information separately.~~

143 3.2.1 Abort job

144 Sub-command syntax:

145 abort-job = %x1 LF

146 This sub-command of receive-a-printer-job is intended to abort any job transfer in process.

147 If the mapper receives this sub-command, it SHALL cancel the job that it is in the process of transmitting.

148 If the mapper is in the process of sending a Print-Job or Create-Job operation, it terminates the job either by closing the
149 connection, or performing the Cancel-Job operation with the job-uri that it received from the Print-Job or Create-Job operation.

150 NOTE: This sub-command is implied if at any time the connection between the LPD client and server is terminated before an
151 entire print job has been transferred via an LPD Receive-a-printer-job request.

152 3.2.2 Receive control file

153 Sub-command syntax:

154 receive-control-file = %x2 number-of-bytes SP name-of-control-file LF
155 number-of-bytes = 1*DIGIT
156 name-of-control-file = "cfA" job-number client-host-name
157 ; e.g. "cfA123woden"
158 job-number = 3DIGIT
159 client-host-name = <a host name>

160 This sub-command is roughly equivalent to the IPP Create-Job operation.

161 The mapper SHALL use the contents of the received LPD control file to create IPP parameter and attribute values to transmit
162 with the Print-Job or Create-Job operation.

163 3.2.3 Receive data file

164 Sub-command syntax: %x3 number-of-bytes-in-data-file Name-of-data-file

165 receive-data-file = %x03 number-of-bytes SP name-of-data-file LF
166 number-of-bytes = 1*DIGIT
167 name-of-data-file = "df" letter job-number client-host-name
168 ; e.g. "dfA123woden for the first file"

169 letter = %x41-5A / %x61-7A ; "A" to "Z", "a" to "z"
 170 ; first file is "A",
 171 ; second "B", and 52nd file is "z"
 172 job-number = 3DIGIT
 173 client-host-name = <a host name>

174 This sub-command is roughly equivalent to the IPP Send-Document operation.

175 The mapper SHALL use the contents of the received LPD data file as the data to transmit with the IPP Print-Job or Send-
 176 Document operation.

177 Although RFC-1179 alludes to a method for passing an unspecified length data file by using an octet-count of zero, no
 178 implementations support this feature.. The mapper SHALL reject a job that has a value of 0 in the number-of-bytes field.

179 3.3 Send queue state (short)

180 Command syntax:

181 send-queue-short = %x03 printer-name *(SP(user-name / job-number)) LF

182 The mapper's response to this command includes information about the printer and its jobs. RFC 1179 specifies neither the
 183 information nor the format of its response. This document requires the mapper to follow existing practice as specified in this
 184 document.

185 The mapper SHALL produce a response in the following format which consists of a printer-status line optionally followed by a
 186 heading line, and a list of jobs. This format is defined by examples below. Appendix A contains the ABNF syntax.

187 For an printer with no jobs, the response starts in column 1 and is:

188 no entries

189 For a printer with jobs, an example of the response is:

```

190 killtree is ready and printing
191 Rank   Owner   Job      Files          Total Size
192 active fred    123     stuff          1204 bytes
193 1st    smith   124     resume, foo    34576 bytes
194 2nd    fred    125     more           99 bytes
195 3rd    mary    126     mydoc          378 bytes
196 4th    jones   127     statistics.ps  4567 bytes
197 5th    fred    128     data.txt       9 bytes
  
```

199 The column numbers of above headings and job entries are:

```

200
201 |         |         |         |
202 01      08      19      35      63
203
  
```

204 The mapper SHALL produce each field above from the following IPP attribute:

LPD field	IPP attribute	special conversion details
printer-status	printer-state and printer-state-reasons	For a printer-state of idle or processing, the mapper SHALL use the formats above. For stopped, the mapper SHALL use printer-state-reasons to produce an unspecified format for the

		error.
rank	number-of-intervening-jobs	the mapper SHALL the format above
owner	job-originating-user- <u>name</u>	unspecified conversion; job-originating-user- <u>name</u> may be the mapper's user-name
job	job- id <u>uri</u>	<u>the mapper shall use the job-id unspecified conversion</u>
files	document-name	the mapper shall create a comma separated list of the document-names and then truncate this list to the first 24 characters
total-size	job-k-octets* <u>copies</u> *1024	the mapper shall multiple the value of job-k-octets by 1024 <u>and by the value of the "copies" attribute.</u>

205

206 A mapper SHOULD use the job attribute number-of-intervening-jobs rather than the job's position in a list of jobs to determine
 207 'rank' because a Printer may omit jobs that it wants to keep secret. If a printer doesn't support the job attribute number-of-
 208 intervening-jobs, a mapper MAY use the job's position.

209 ~~ISSUE: is job-k-octets the sum of the bytes of each document times the number of copies? If so, "total-size" is 1024 times job-k-~~
 210 ~~octets. The model document needs clarification.~~

211 ~~ISSUE: the value of job-originating-user-name is still somewhat murky in IPP especially where the IPP client is a proxy for some~~
 212 ~~user coming from elsewhere.~~

213 In order to obtain the information specified above, The LPD-to-IPP mapper SHALL use the Get-Attributes operation of the
 214 printer to get printer-status and SHOULD use the Get-Jobs operation to get information about all of the jobs. If the LPD
 215 command contains job-numbers or user-names, the mapper MAY handles the filtering of the response ~~because Get-Jobs has no~~
 216 ~~way to limit jobs to those of a particular user.~~ If the LPD command contains job-numbers but no user-names, the mapper MAY
 217 use Get-Attributes on each converted job-number rather than Get-Jobs. If the LPD command contains a single user-name but no
 218 job-numbers, the mapper MAY use Get-Attributes with the my-jobs option if the server supports this option and if the server
 219 allows the client to be a proxy for the LPD user.

220 NOTE: This specification does *not* define how the mapper maps the LPD Printer-name operand to the IPP "printer-uri"
 221 parameter.

222 3.4 Send queue state (long)

223 Command syntax:

224 send-queue-long = %x04 printer-name *(SP(user-name / job-number)) LF

225 The mapper's response to this command includes information about the printer and its jobs. RFC 1179 specifies neither the
 226 information nor the format of its response. This document requires the mapper to follow existing practice as specified in this
 227 document.

228 The mapper SHALL produce a response in the following format which consists of a printer-status line optionally followed a list
 229 of jobs, where each job consists of a blank line, a description line, and one line for each file. The description line contains the
 230 user-name, rank, job-number and host. This format is defined by examples below. Appendix B contain the ABNF syntax.

231 For an printer with no jobs the response is:

232 no entries

233 For a printer with jobs, an example of the response is:

```

234 killtree is ready and printing
235
236 fred: active                [job 123 tiger]
237     2 copies of stuff      602 bytes
238
239 smith: 1st                  [job 124 snail]
240     2 copies of resume     7088 bytes
241     2 copies of foo        10200 bytes
242
243 fred: 2nd                   [job 125 tiger]
244     more                    99 bytes
245

```

246 The column numbers of above headings and job entries are:

```

247
248 |           |           |
249 01         09         41
250

```

251 Although the format of the long form is different from the format of the short form, their fields are identical except for **a)** the
 252 copies and host fields which are only in the long form, **and b)** the “size” field contains the single copy size of each file. Thus the
 253 sum of the file sizes in the “size” field times the value of the “copies” field produces the value for the “Total Size” field in the
 254 short form. For fields other than the host and copies fields, see the preceding section. For the host field see the table below.

LPD field	IPP attribute	special conversion details
host	job-originating-host	unspecified conversion; job-originating-host may be the mapper’s host
copies	copies	the mapper shall assume the value of copies precedes the string “copies of”; otherwise, the value of copies is 1.

255

256 NOTE: This specification does *not* define how the mapper maps the LPD Printer-name operand to the IPP printer-uri parameter.

257 3.5 Remove jobs

258 Command syntax:

```

259   remove-jobs = %x05 printer-name SP agent
260                *(SP(user-name / job-number)) LF

```

261 The agent operand is the user-name of the user initiating the remove-jobs command. The special user-name 'root' indicates a
 262 privileged user who can remove jobs whose user-name differs from the agent..

263 The mapper SHALL issue one Cancel-Job operation for each job referenced by the remove-jobs command. Each job-number in
 264 the remove-jobs command references a single job. Each user-name in the remove-jobs command implicitly references all jobs
 265 owned by the specified user. The active job is implicitly referenced when the remove-jobs command contains neither job-
 266 numbers nor user-names. The mapper MAY use Get-Jobs to determine the job-uri of implicitly referenced jobs.

267 The mapper SHALL not use the agent name of ‘root’ when end-users cancel their own jobs. Violation of this rule creates a
 268 potential security violation, and it may cause the printer to issue a notification that misleads a user into thinking that some other
 269 person canceled the job.

270 If the agent of a remove-jobs command for a job J is the same as the user name specified with the ‘P’ function in the control file
 271 for job J, then the mapper SHALL ensure that the caller of the Cancel-Job command for job J is the same as job-originating-user
 272 for job J.

273 Note: This requirement means that a mapper must be consistent in who the receiver perceives as the caller of IPP operations. The
 274 mapper either acts as itself or acts on behalf of another user. The latter is preferable if it is possible. This consistency is necessary
 275 between Print-Job/Create-Job and Cancel-Job in order for Cancel-Job to work, but it is also desirable for other operations. For
 276 example, Get-Jobs may give more information about job submitted by the caller of this operation.

277 NOTE: This specification does *not* define how the mapper maps: (1) the LPD printer-name to the IPP "printer-uri" or (2) the
 278 LPD job-number to the IPP "job-uri".

279 NOTE: This specification does not specify how the mapper maps the LPD user-name to the IPP job-originating-user because the
 280 mapper may use its own user-name with jobs.

281 4. Mapping of LPD Control File Lines to IPP Parameters

282 This section describes the mapping from LPD control file lines (called 'functions') to IPP operation input parameters. The
 283 mapper receives the control file lines via the LPD receive-control-file sub-command.. Each of the LPD functions appear as sub-
 284 sections of section 7 of RFC 1179.

285 In LPD control file lines, the text operands have a maximum length of 31 or 99 while IPP input parameters have a maximum of
 286 255 characters. Therefore, no data is lost.

287 The mapper converts each supported LPD function to its corresponding IPP parameter as defined by tables in the subsections that
 288 follow. These subsections group functions according to whether they are:

- 289 • required with a job,
- 290 • optional with a job
- 291 • required with each document.

292 In the tables below, each LPD value is given a name, such as 'h'. If an IPP value uses the LPD value, then the IPP value column
 293 contains the LPD name, such as 'h' to denote this. Otherwise, the IPP value column specifies the literal value.

294 4.1 Required Job Functions

295 The following LPD functions MUST be in a received LPD job. The mapper SHALL receive each of the following LPD functions
 296 and SHALL include the information as a parameter with each IPP job. The functions SHOULD be in the order 'H', 'P' and they
 297 SHOULD be the first two functions in the control file, but they MAY be anywhere in the control file and in any order.

LPD function		description	IPP	
name	value		name	value
H	<i>h</i>	Originating Host		<i>h</i> (in security layer)
P	<i>u</i>	User identification		<i>u</i> (in security layer <u>and operation attribute</u>)
		<i>none</i>	best-effort	'true'

298 A mapper MAY send its own host rather than the client's host, and a mapper MAY send its own user-name as user identification
 299 rather than the client user. But in any case, the values sent SHALL be compatible with the Cancel-Job operation. The IPP
 300 operation MAY have no way to specify an originating host-name.

301 ISSUE: what do we do about job-originating-host?

302 The mapper SHALL include best-effort=true so that it doesn't have to determine which attributes a printer supports.

303 4.2 Optional Job Functions

304 The following LPD functions MAY be in a received job. These function SHOULD follow the required job functions and precede
305 the document functions, but they MAY be anywhere in the control file.

306 If the mapper receives such an LPD function, the mapper SHALL include the corresponding IPP attribute with the value
307 converted as specified in the table below. If the mapper does not receive such an LPD attribute, the mapper SHALL NOT
308 include the corresponding IPP attribute, except the 'L' LPD function whose absence has a special meaning as noted in the table.

LPD function name	value	description	IPP name	value
J	<i>j</i>	Job name for banner page	job-name	<i>j</i>
L	<i>l</i>	Print banner page	job-sheets	'standard' if 'L' is present 'none' if 'L' is present
M	<i>m</i>	Mail When Printed	notification-events notification-method	<u>IPP has no notification mechanism. To support this LPD feature, the gateway must poll 'job-completion' 'mailto:/'m'@'h</u>

309 ~~Note: 'm' is the user name and not an email address. The mapper can fabricate an email address with the source host. This email~~
310 ~~address mail fail when the 'h' is some intermediary host that doesn't know about user 'm'. But there is no better solution.~~

311 4.3 Required Document Functions

312 The mapper SHALL receive one set of the required document functions with each copy of a document, and SHALL include the
313 converted information as parameters with each IPP document

314 If the control file contains required and recommended document functions, the required functions SHOULD precede the
315 recommended ones and if the job contains multiple documents, all the functions for each document are grouped together as
316 shown in the example of section 6.3 "Required Document Functions". However, the document functions MAY be in any order.

317

LPD function name	value	description	IPP name	value
f	fff	Print formatted file	document-format	<u>'application/octet-stream'³⁷</u> <u>(langAutomatic)</u>
l	fff	Print file leaving control characters	document-format	<u>'application/octet-stream'³⁷</u> <u>(langAutomatic)</u>
o	fff	Print Postscript output file	document-format copies	<u>'application/PostScript'⁶ (langPS).</u> see note

318 Note: In practice, the 'f' LPD function is often overloaded. It is often used with any format of document data including PostScript
319 and PCL data.

320 Note: In practice, the 'l' LPD function is often used as a rough equivalent to the 'f' function.

321 Note: When RFC 1179 was written, no implementation supported the 'o' function; instead 'f' was used for PostScript. Windows
322 NT now sends 'o' function for a PostScript file.

323 Note: the value 'fff' of the 'f', 'l' and 'o' functions is the name of the data file as transferred, e.g. "dfA123woden".

324 If the mapper receives any other lower case letter, the mapper SHALL reject the job because the document contains a format that
325 the mapper does not support.

326 The mapper determines the number of copies by counting the number of occurrences of each 'fff' file with one of the lower-case
327 functions above. For example, if 'f dfA123woden' occurs 4 times, then copies has a value of 4. Although the LPD protocol
328 allows the value of copies to be different for each document, the commands and the receiving print systems don't support this.

329 ~~ISSUE: should we register DVI, ditroff and troff. At least DVI and one of the troff is still in use.~~

330 4.4 Recommended Document Functions

331 The mapper SHOULD receive one set of the recommended document functions with each document, and SHOULD include the
332 converted information as parameters with each IPP document. The functions SHOULD be received in the order 'U' and 'N', but
333 they MAY arrive in any order.

334

LPD function		description	IPP	
name	value		name	value
U	<i>fff</i>	Name of source file	<i>ignored</i>	
N	<i>n</i>		document-name	<i>n</i>

335 Note: the value '*fff*' of the 'U' function is the name of the data file as transferred, e.g. "dfA123woden".

336 5. Mapping from IPP operations to LPD commands

337 If the IPP-to-LPD mapper receives an IPP operation, the following table summarizes the LPD command that it uses. Each section
338 below gives the detail. Each of the following sub-sections appear as sub-sections of section 3 in the document "Internet Printing
339 Protocol/1.0: Model and Semantics" [ipp-mod].

IPP operation	LPD command
Print-Job or Print-URI or Create-Job/Send-Document/Send-URI	receive-a-printer-job and then print-any-waiting-jobs
Validate-Job	implemented by the mapper
Cancel-Job	remove-jobs
Get-Attributes (printer or job) or Get-Jobs	send queue state (short or long)

340 ~~5.1 Get-Operations~~

341 ~~The mapper SHALL return a list of the operations that it supports. It SHALL support at least those operations that are mandatory~~
342 ~~according to the IPP model document [ipp-mod].~~

343 5.1 Print-Job

344 The mapper SHALL send the following commands in the order listed below:

- 345 • receive-a-printer-job command

- 346 • both receive-control-file sub-command and receive-data-file sub-command
347 (unspecified order, see Note below)
- 348 • print-any-waiting-jobs command,
349 except that if the mapper is sending a sequence of receive-a-printer-job commands, it MAY omit sending print-
350 any-waiting-jobs after any receive-a printer-job command that is neither the first nor last command in this
351 sequence

352 Note: it is recommended that the order of the receive-control-file sub-command and the receive-data-file sub-command be
353 configurable because either order fails for some print systems. Some print systems assume that the control file follows all data
354 files and start printing immediately on receipt of the control file. When such a print system tries to print a data file that has not
355 arrived, it produces an error. Other print systems assume that the control file arrives before the data files and start printing when
356 the first data file arrives. Such a system ignores the control information, such as banner page or copies.

357 NOTE: This specification does not define the mapping between the IPP printer-uri and the LPD printer-name.

358 The mapper SHALL send the IPP parameters and attributes received from the operation to the LPD printer by using the LPD
359 receive-control-file sub-command. The mapper SHALL create the LPD job-number for use in the control file name, but the
360 receiving printer MAY, in some circumstances, assign a different job-number to the job. The mapper SHALL create the IPP job-
361 id and IPP job-uri returned in the Print-Job response.

362 NOTE: This specification does not specify how the mapper determines the LPD job-number, the IPP job-id or the IPP job-uri of
363 a job that it creates nor does it specify the relationship between the IPP job-uri, IPP the job-id and the LPD job-number, both of
364 which the mapper creates. However, it is likely that the mapper will use the same integer value for both the LPD job-number and
365 the IPP job-id, and that the IPP Job-uri is the printer's URI with the job-id concatenated on the end.

366 The mapper SHALL send data received in the IPP operation to the LPD printer by using the LPD receive-data-file sub-command.
367 The mapper SHALL specify the exact number of bytes being transmitted in the number-of-bytes field of the receive-data-file sub-
368 command. It SHALL NOT use a value of 0 in this field.

369 If the mapper, while it is transmitting a receive-a-printer-job command or sub-command, either detects that its IPP connection has
370 closed or receives a Cancel-Job operation, the mapper SHALL terminate the LPD job either with the abort sub-command or the
371 remove-jobs command.

372 ISSUE: error code conversion.

373 5.2 Print-URI

374 The mapper SHALL handle this operation in the same way as a Print-Job operation except that it SHALL obtain data referenced
375 by the "document-uri" parameter and SHALL then treat that data as if it had been received via a Print-Job operation.

376 5.3 Validate-Job

377 The mapper SHALL perform this operation directly. Because LPD supports very few attributes, this operation doesn't have much
378 to check.

379 5.4 Create-Job

380 The mapper SHALL handle this operation like Print-Job, except

- 381 • the mapper SHALL send the control file after it has received the last Send-Document or Send-URI operation
- 382 because the control file contains all the document-name and document-format values specified in the Send-
- 383 Document and Send-URI operations.
- 384 • the mapper SHALL perform one receive-data-file sub-command for each Send-Document or Send-URI
- 385 operation received and in the same order received.
- 386 • the mapper SHALL send the control file either before all data files or after all data files.
- 387 (See the note in the section on Print-Job about the dilemma of sending the control file either before or after the
- 388 data files.

389 5.5 Send-Document

390 The mapper performs a receive-data-file sub-command on the received data. See the preceding section 5.4 "Create-Job" for the
391 details.

392 5.6 Send-URI

393 The mapper SHALL obtain the data referenced by the "document-uri" parameter, and SHALL then treat that data as if it had been
394 received via a Send-Document operation. See the preceding section 5.5 "Send-Document" for the details.

395 5.7 Cancel-Job

396 The mapper SHALL perform a remove-jobs command with the following parameters:

- 397 • the printer is the one to which the job was submitted, that is the IPP printer-uri is mapped to an LPD printer-
- 398 name by the same mechanism as for all commands, containing the job specified by the IPP job-uri,
- 399 • the agent is the authenticated user-name of the IPP client,
- 400 • the job-number is the job-id returned by the Print-Job command, that is, the LPD job-number has the same
- 401 value as the IPP job-id for likely implementations. one corresponding to the IPP job-uri parameter.

402 ~~NOTE: This specification does *not* specify how the mapper maps the IPP "job-uri" to the LPD printer-name or LPD job-number.~~

403 ~~ISSUE: the model needs to offer a solution for mapping jobs to printers either with a new job attribute "printer-uri" or with all~~
404 ~~operation targets being a printer-uri.~~

405 5.8 Get-Attributes

406 LPD severely limits the set of attributes that the mapper is able to return in its response for this operation.

407 When the mapper receives a Get-Attributes operation for a printer object, it SHALL support, at most, the following printer
408 attributes:

- 409 • printer-state
- 410 • printer-state-reasons

411 When the mapper receives a Get-Attributes operation for a job object, it SHALL support, at most, the following job attributes:

- 412 • number-of-intervening-jobs
- 413 • job-originating-user-name

- 414 • job-~~id~~uri
- 415 • ~~job-originating-host~~
- 416 • document-name
- 417 • job-k-octets
- 418 • copies

419 The mapper uses either the long or short form of the “send queue state” command. If it receives a request for the “~~job-originating-~~
420 ~~host~~” or “~~job-k-octets~~” or “copies” and supports ~~thet~~hose attribute it SHALL use the long form; otherwise, it SHALL use the
421 short form.

422 Note: the value of job-k-octets is the value in the short form divided by the number of “copies” which is on the long form only. Its
423 value can also be determined by adding the “size” field values for each document in the job in the long form., but it can be
424 computed from the copies and file size fields in the long form.

425 The mapper SHALL assume that the LPD response that it receives has the format and information specified in section 3.3 “Send
426 queue state (short)” and section 3.4 “Send queue state (long)”. The mapper SHALL determine the value of each requested
427 attribute by using the inverse of the mapping specified in the two aforementioned sections.

428 Note: when the mapper receives the Get-Attributes operation for a printer, it can determine the response from the printer-status
429 line without examining the rest of the LPD response. When the mapper receives the Get-Attributes operation for a job and uses
430 the LPD short form, it can determine the response from the single line that pertains to the job specified by the Get-Attributes
431 operation.

432 NOTE: for Get-Attributes of a job, the mapper can use its correspondence between the IPP job-id, job-uri and the LPD job-
433 number.

434 NOTE: For Get-Attributes of a job, this specification does *not* specify how the mapper maps the IPP “job-uri” to the LPD
435 printer-name or LPD job-number.

436 5.9 Get-Jobs

437 The mapper SHALL perform this operation in the same way as Get-Attributes of a printer except that the mapper converts the
438 job-lines, and the IPP response contains one job object for each job in the LPD response..

439 6. Mapping of IPP Parameters to LPD Control File Lines

440 This section describes the mapping from IPP operation input parameters to LPD control file lines (called ‘functions’). The
441 mapper receives the IPP operation input parameters via the IPP operation. Each of the IPP operation input parameters appear as
442 sub-sections of section 3 and 4.2 in the IPP model document [ipp-mod].

443 In the context of LPD control file lines, the text operands have a maximum length of 31 or 99 while IPP input parameters have a
444 maximum of 255 characters. Therefore, there may be some data loss if the IPP parameters exceed the maximum length of the
445 LPD equivalent operands.

446 The mapper converts each supported IPP parameter to its corresponding LPD function as defined by tables in the subsections that
447 follow. These subsections group functions according to whether they are:

- 448 • required with a job,
- 449 • optional with a job
- 450 • required with each document.

451 In the tables below, each IPP value is given a name, such as 'h'. If an LPD value uses the IPP value, then the LPD value column
 452 contains the IPP name, such as 'h' to denote this. Otherwise, the LPD value column specifies the literal value.

453 **6.1 Required Job Functions**

454 The mapper SHALL include the following LPD functions with each job, and they SHALL have the specified value. They SHALL
 455 be the first functions in the control file and they SHALL be in the order "H" and then "P".

IPP name	value	LPD function		description
		name	value	
(perhaps in security layer)	<i>h</i>	H	<i>gateway host</i>	Originating Host
(in security layer)	<i>u</i>	P	<i>u</i>	User identification

456 A mapper SHALL send its own host rather than the client's host, because some LPD systems require that it be the same as the
 457 host from which the remove-jobs command comes. A mapper MAY send its own user name as user identification rather than the
 458 client user. But in any case, the values sent SHALL be compatible with the LPD remove-jobs operation.

459 **6.2 Optional Job Functions**

460 The mapper MAY include the following LPD functions with each job. They SHALL have the specified value if they are sent.
 461 These functions, if present, SHALL follow the require job functions, and they SHALL precede the required document functions.

462

IPP attribute name	value	LPD function		description
		name	value	
job-name	<i>j</i>	J	<i>j</i>	Job name for banner page
job-sheets	'standard'	L	<i>u</i>	Print banner page
job-sheets	'none'			omit 'L' function

463 Note: 'L' has special meaning when it is omitted. ~~If 'M' is omitted, there is no notification.~~ If 'J' is omitted, some undefined
 464 behavior occurs with respect to the banner page.

465 ~~Note: the 'user' for the 'M' function comes from a substring of the notification-method's value.~~

466 **6.3 Required Document Functions**

467 The mapper SHALL include one set of the following LPD functions with each document, and they SHALL have the specified
 468 values. For each document, the order of the functions SHALL be 'f', 'U' and then 'N', where 'f' is replicated once for each copy.

469

IPP attribute name	value	LPD function		description
		name	value	
document-format	'application/octet-stream' or 'application/PostScript' ³⁷ (langAutomatic) or '6' (langPS).	f	<i>fff</i>	Print formatted file

IPP attribute name	value	LPD function		description
		name	value	
copies	<i>c</i>			replicate 'f' 'c' times
<i>none</i>		U	<i>fff</i>	Unlink data file
document-name	<i>n</i>	N	<i>n</i>	Name of source file

470 Note: the value '*fff*' of the 'f' and 'U' functions is the name of the data file as transferred, e.g. "dfA123woden".

471 Note: the mapper SHALL not send the 'o' function

472 ISSUE: should we register DVI, troff or ditroff?

473 If the mapper receives no "best-effort" or it has a value of false, then the mapper SHALL reject the job if it specifies attributes or
474 attribute values that are not among those supported in the above tables.

475 Below is an example of the minimal control file for a job with three copies of two files 'foo' and 'bar':

```
476 H tiger
477 P jones
478 f dfA123woden
479 f dfA123woden
480 f dfA123woden
481 U dfA123woden
482 N foo
483 f dfB123woden
484 f dfB123woden
485 f dfB123woden
486 U dfB123woden
487 N bar
```

488 7. Copyright Notice

489 This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise
490 explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without
491 restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative
492 works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to
493 the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which
494 case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into
495 languages other than English.

496 The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

497 This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND
498 THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING
499 BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE
500 ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR
501 PURPOSE.

502 8. References

503 [ipp-mod] R. deBry, T. Hastings, R. Herriot, S. Isaacson, P. Powell, "Internet Printing Protocol/1.0: Model and Semantics",
504 <draft-ietf-ipp-model-07.txt>, November 1997.

- 505 [ipp-pro] R. Herriot, S. Butler, P. Moore, R. Turner, "Internet Printing Protocol/1.0: Protocol specification", <draft-ietf-ipp-
506 protocol-03.txt>, November 1997.
- 507 [rfc1179] L. McLaughlin, "Line Printer Daemon Protocol", RFC 1179, August 1990.
- 508 [rfc1759] Smith, R., Wright, F., Hastings, T., Zilles, S., and Gyllenskog, J., "Printer MIB", RFC 1759, March 1995.
- 509 [rfc2119] S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119 , March 1997
- 510 [abnf] D. Crocker et al., "Augmented BNF for Syntax Specifications: ABNF", draft-ietf-drums-abnf-05.txt.

511 9. Author's Addresses

Robert Herriot (editor)
Sun Microsystems Inc.
901 San Antonio.Road., MPK-17
Mountain View, CA 94043

Phone: 650-786-8995
Fax: 650-786-7077
Email: robert.herriot@eng.sun.com

Thomas N. Hastings
Xerox Corporation
701 S. Aviation Blvd., ESAE-231
El Segundo, CA 90245

Phone: 310-333-6413
Fax: 310-333-5514
EMail: hastings@cp10.es.xerox.com

Norm Jacobs
Sun Microsystems Inc.
1430 Owl Ridge Rd.
Colorado Springs, CO 80919

Phone: 719-532-9927
Fax: 719-535-0956
Email: Norm.Jacobs@Central.sun.com

Jay Martin
Underscore, Inc.
41-C Sagamore Park Road
Hudson, NH 03051-4915

Phone: 603-889-7000
Fax: 603-889-2699
Email: jkm@underscore.com

512

513 10. Appendix A: ABNF Syntax for response of Send-queue-state (short)

514 The syntax in ABNF for the response to the LPD command 'send-queue-state (long)' is:

515 status-response = empty-queue / nonempty-queue
516 empty-queue = "no-entries" LF
517 nonempty-queue = printer-status LF heading LF *(job LF)
518 printer-status = OK-status / error-status
519 OK-status = printer-name SP "ready and printing" LF
520 error-status = < implementation dependent status information >
521 heading = "Rank" 3SP "Owner" 6SP "Job" 13SP "Files"
522 23SP "Total Size" LF
523 ; the column headings and their values below begin at the columns
524 ; 1, 8, 19, 35 and 63
525 job = rank *SP owner *SP job *SP files *SP total-size "bytes"
526 ; jobs are in order of oldest to newest
527 rank = "active" / "1st" / "2nd" / "3rd" / integer "th"
528 ; job that is printing is "active"
529 ; other values show position in the queue

530 owner = <user name of person who submitted the job>
 531 job = 1*3DIGIT ; job-number
 532 files = <file name> *(“,” <file name>); truncated to 24 characters
 533 total-size = 1*DIGIT ; combined size in bytes of all documents

534 11. Appendix B: ABNF Syntax for response of Send-queue-state (long)

535 The syntax in ABNF for the response to the LPD command ‘send-queue-state (long)’ is:

536 status-response = empty-queue / nonempty-queue
 537 empty-queue = “no-entries” LF
 538 nonempty-queue = printer-status LF *job
 539 printer-status = OK-status / error-status
 540 OK-status = printer-name SP “ready and printing” LF
 541 error-status = < implementation dependent status information >
 542 job = LF line-1 LF line-2 LF
 543 line-1 = owner “:” SP rank 1*SP “[job] job SP host “[”
 544 line-2 = file-name 1*SP document-size “bytes”
 545 ; jobs are in order of oldest to newest
 546 rank = “active” / “1st” / “2nd” / “3rd” / integer “th”
 547 ; job that is printing is “active”
 548 ; other values show position in the queue
 549 owner = <user name of person who submitted the job>
 550 job = 1*3DIGIT
 551 file-name = [1*DIGIT “copies of” SP] <file name>
 552 ; truncated to 24 characters
 553 document-size = 1*DIGIT ;size of single copy of the document.

554 12. Appendix C: Unsupported LPD functions

555 The follow LPD functions have no IPP equivalent. The LPD-to-IPP mapper ignores them and the IPP-to-LPD mapper does not
 556 send them.

LPD command

name	description
C	Class for banner page
I	Indent Printing
<u>H</u>	<u>Host of client</u>
<u>M</u>	<u>Mail when printed</u>
S	Symbolic link data
T	Title for pr
W	Width of output
1	troff R font
2	troff I font
3	troff B font
4	troff S font

557

558 The follow LPD functions specify document-formats which have no IPP equivalent, unless someone registers them. The LPD-to-
 559 IPP mapper rejects jobs that request such a document format, and the IPP-to-LPD mapper does not send them.

LPD command

name	description
c	Plot CIF file
d	Print DVI file
g	Plot file
k	reserved for Kerberized clients and servers
n	Print ditroff output file
p	Print file with 'pr' format
r	File to print with FORTRAN carriage control
t	Print troff output file
v	Print raster file
z	reserved for future use with the Palladium print system

560

561 ISSUE: we may move some of these to the supported list.