

1 PWG DRAFT ~~to become an INTERNET DRAFT~~
2 draft-ietf-ipp-ops-set1-00-9809020.txt

3 R. Bergman
4 Data-Products Corp.
5 T. Hastings
6 Xerox Corporation
7 R. Herriot
8 Sun Microsystems
9 P. Moore
10 Microsoft

11 ~~July 27~~ September 20, 1998

13 Internet Printing Protocol/1.0: Additional Optional Operations - Set 1

14 Status of this Memo

15 This document is a Printer Working Group (PWG) DRAFT ~~an Internet Draft~~. The previous
16 version (980727) was posted to the IPP mailing list for comment on 7/29/98 and reviewed at the
17 August, 19-20, 1998 PWG meeting in Toronto. This version contains the changes agreed to at
18 that meeting, and the IPP telecons on 9/2/98, 9/9/98, and 9/16/98 and need to be reviewed by the
19 participants on the mailing list. Internet Drafts are working documents of the Internet
20 Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may
21 also distribute working documents as Internet Drafts.

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

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

29 **Abstract**

30 This document specifies ~~seven~~ six OPTIONAL operations for use with the Internet Printing
31 Protocol/1.0 (IPP) [ipp-mod, ipp-pro]. The defined Set 1 operations are 3 Job object operations
32 that end-users may perform on their jobs and operators/administrators may perform on any job:

33 Hold-Job
34 Release-Job
35 Restart-Job
36 ~~Reprocess-Job~~

37 and 3 Printer object operations that operators/administrators may perform on a Printer object:

38 Pause-Printer
39 Resume-Printer
40 Purge-Jobs

41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60

Table of Contents

1 Summary of Set 1 and Operation-Id Assignments 3

2 Job Operations..... 3

2.1 Hold-Job..... 4

2.1.1 "job-hold-until" (type3 keyword | name(MAX)) operation attribute 4

2.2 Release-Job..... 5

2.3 Restart-Job..... 6

~~2.4 Reprocess-Job..... 9~~

3 Additional "job-state-reasons" and the Job History concept 8

3.1 The "job history" concept 8

3.2 Add a new 'job-restartable' value to the "job-state-reasons" attribute 9

4 Printer operations 11

4.1 Pause-Printer 13

4.1.1 Add a new 'moving-to-paused' value to the "printer-state-reasons" attribute 14

4.2 Resume-Printer..... 14

4.3 Purge-Jobs 15

5 Security Considerations..... 15

~~6~~ References 15

61 1 Summary of Set 1 and Operation-Id Assignments

62 The Set 1 operations are summarized in the following table:

Operation Name	Operation-Id	Brief description
Hold-Job	0x000C	Holds a pending job so that it cannot be scheduled for processing
Release-Job	0x000D	Allows a previously held job to be scheduled for processing
Restart-Job	0x000E	Restarts a completed job as the same job on the same Printer object
Reprocess-Job	0x000F	Reserved for the future. Reprocesses a completed job as a new copy of the job on the same Printer object
Pause-Printer	0x0010	Stops the device(s) as soon as possible from processing jobs
Resume-Printer	0x0011	Resumes the device(s) processing jobs
Purge-Jobs	0x0012	Removes all jobs from the Printer regardless of job state

63 All of the ~~attributes-operations~~ in Set 1 are OPTIONAL for an IPP object to support. Unless the
 64 specification of an OPTIONAL operation requires support of another OPTIONAL operation,
 65 conforming implementations may support any combination of these operations.

66 2 Job Operations

67 ~~The job operations in Set 1 are for use by end users on their jobs and by operators and~~
 68 ~~administrators on any jobs.~~ The operation attributes in requests and responses for the job
 69 operations are the same as the standard Cancel-Job operation (see [model] 3.3.3). Additional
 70 operation attributes are specified that the client MAY supply in a request. In addition, the IPP
 71 object MUST return the "job-state" [ipp-mod 4.3.7] attribute and, if supported, the "job-state-
 72 reasons" [ipp-mod 4.3.8] attribute in the response as Group 3 Job Attributes in order to indicate
 73 the effect of the operation on the job object.

74 ~~Note: In order to keep the operations in Operation Set 1 simple, they are rejected when the job is~~
 75 ~~in the 'processing' or 'processing-stopped' states. If operations are needed to affect jobs while in~~
 76 ~~these states, they will be added as additional operations, rather than overloading these operations.~~
 77 ~~Then it is clear to clients by querying the Printer object's "operations-supported" [ipp-mod~~
 78 ~~4.4.13] what the behavior is.~~

79 2.1 Hold-Job

80 This operation allows a client to hold a pending job in the queue so that it is not eligible for
81 scheduling. If the Hold-Job operation is supported, then the Release-Job operation MUST be
82 supported, and vice-versa.

83 2.1.1 "job-hold-until" (type3 keyword | name(MAX)) operation attribute

84 The client OPTIONALLY supplies this attribute. The IPP object MUST support this operation
85 attribute in a Hold-Job request, if it supports the "job-hold-until" Job template attribute in create
86 operations. See [ipp-mod] section 4.2.2. Otherwise, the IPP object NEED NOT support the
87 "job-hold-until" operation attribute in a Hold-Job request. If supplied and supported, the IPP
88 object copies the attribute to the Job object, replacing the job's previous "job-hold-until"
89 attribute, if present, and makes the job a candidate for scheduling during the supplied named time
90 period.

91 As with all operations, if the client supplies the "job-hold-until" (or any OPTIONAL) Operation
92 attribute that is unknown or unsupported or the value is unsupported, the IPP object MUST
93 accept and perform the operation, ignoring the unknown or unsupported operation attribute and
94 returning the ignored or unsupported attributes and/or values in Group 2 Unsupported Attributes
95 (see [ipp-mod] sections 3.3.3.2 and 16.3.6).

96 If the client (1) supplies a value that specifies a time period that has already started or the 'no-
97 hold' value [ipp-mod 4.2.2] (meaning don't hold the job) and (2) the IPP object supports the "job-
98 hold-until" operation attribute and there are no other reasons to hold the job, the IPP object
99 MUST reject-accept the operation and make the job be a candidate for processing immediately
100 (see [ipp-mod] Section 4.2.2)~~return the 'client-error-bad-syntax' error status code.~~

101 The following new keyword value is defined for use with the "job-hold-until" Job Template
102 attribute in job create operations and the "job-hold-until" operation attribute in Hold-Job and
103 Restart-Job operations:

104 'indefinite': - the job is held indefinitely, until a client performs a Release-Job or Restart-
105 Job operation

106 If the client does not supply a "job-hold-until" operation attribute in the requestHold-Job
107 operation, the IPP object MUST populate the job object with a "job-hold-until" attribute with the
108 'indefinite' value (if IPP object supports the "job-hold-until" attribute) and hold the job
109 indefinitely, until a client performs a Release-Job or Restart-Job operation.

110 The IPP object SHOULD support the "job-hold-until" Job Template attribute for use in job create
111 operations with at least the 'indefinite' value, if it supports the Hold-Job operation. Otherwise, a

112 client cannot create a job and hold it immediately (without picking some supported time period
 113 in the future).

114 The IPP object MUST accept or reject the request based on the job's current state, transition the
 115 job to the indicated new state, and return the indicated new "job-state" attribute and status code
 116 as follows:

Current "job-state"	New "job-state"	IPP object's response status code and action:
'pending'	'pending-held'	'successful-ok' See Note 1
<u>'pending'</u>	<u>'pending'</u>	<u>'successful-ok' See Note 2</u>
'pending-held'	'pending-held'	'successful-ok' See Note 1
<u>'pending-held'</u>	<u>'pending'</u>	<u>'successful-ok' See Note 2</u>
'processing'	'processing'	'client-error-not-possible'
'processing-stopped'	'processing-stopped'	'client-error-not-possible'
'completed'	'completed'	'client-error-not-possible'
'canceled'	'canceled'	'client-error-not-possible'
'aborted'	'aborted'	'client-error-not-possible'

117 Note 1: If the OPTIONAL "job-state-reasons" attribute is supported and if the implementation
 118 supports multiple reasons for a job to be in the 'pending-held' state, the IPP object MUST add the
 119 'job-hold-until-specified' value to the job's "job-state-reasons" attribute.

120 Note 2: If the IPP object supports the "job-hold-until" operation attribute, but the specified time
 121 period has already started (or is the 'no-hold' value) and there are no other reasons to hold the job,
 122 the IPP object MUST make the job be a candidate for processing immediately (see [ipp-mod]
 123 Section 4.2.2) by putting the job in the 'pending' state.

124 Note: In order to keep the Hold-Job operation simple, such a request is rejected when the job is
 125 in the 'processing' or 'processing-stopped' states. If an operation is needed to hold jobs while in
 126 these states, it will be added as an additional operation, rather than overloading the Hold-Job
 127 operation. Then it is clear to clients by querying the Printer object's "operations-supported" [ipp-
 128 mod 4.4.13] and the Job object's "job-state" [ipp-mod 4.3.7] attributes which operations are
 129 possible.

130 *Access Rights:* The requesting user must either be the submitter of the job or an **operator or**
 131 administrator of the **Printer object (see [ipp-mod] Section 1)**. Otherwise, the IPP object MUST
 132 reject the operation and return: 'client-error-forbidden', 'client-error-not-authenticated', or 'client-
 133 error-not-authorized' as appropriate.

134 **2.2 Release-Job**

135 This operation allows a client to release a previously held job so that it is again eligible for
 136 scheduling. This operation removes the "job-hold-until" job attribute, if present, from the job

137 object that had been supplied in the create or most recent Hold-Job [or Restart-Job](#) operation and
138 remove its effect on the job.

139 If the Hold-Job operation is supported, then the Release-Job operation MUST be supported, and
140 vice-versa.

141 If the OPTIONAL "job-state-reasons" attribute is supported, the IPP object MUST remove the
142 'job-hold-until-specified' value from the job's "job-state-reasons" attribute, if present.

143 The IPP object MUST accept or reject the request based on the job's current state, transition the
144 job to the indicated new state, and return the indicated new "job-state" attribute and status code
145 as follows:

Current "job-state"	New "job-state"	IPP object's response status code and action:
'pending'	'pending'	'successful-ok' No effect on the job.
'pending-held'	'pending-held'	'successful-ok' See Note 1
'pending-held'	'pending'	'successful-ok'
'processing'	'processing'	'successful-ok' No effect on the job.
'processing-stopped'	'processing-stopped'	'successful-ok' No effect on the job.
'completed'	'completed'	'client-error-not-possible'
'canceled'	'canceled'	'client-error-not-possible'
'aborted'	'aborted'	'client-error-not-possible'

146 Note 1: If there are other reasons to keep the job in the 'pending-held' state, such as 'resources-
147 are-not-ready', the job remains in the 'pending-held' state. Thus the 'pending-held' state is not just
148 for jobs that have the 'job-hold-until' applied to them, but are for any reason to keep the job from
149 being a candidate for scheduling and processing, such as 'resources-are-not-ready'. [See the "job-
150 hold-until" attribute \(\[ipp-mod\] Section 4.2.2\).](#)

151 *Access Rights:* The requesting user must either be the submitter of the job or an [operator or](#)
152 administrator of the [Pprinter object](#). Otherwise, the IPP object MUST reject the operation and
153 return: 'client-error-forbidden', 'client-error-not-authenticated', or 'client-error-not-authorized' as
154 appropriate.

155 **2.3 Restart-Job**

156 This operation allows a client to restart a job that is retained in the queue after processing has
157 completed. [As an implementation option, a job in the 'processing' and/or 'processing-stopped'](#)
158 [states MAY be restarted.](#)

159 The job [is moved to the 'pending' job state and](#) restarts at the beginning on the same IPP Printer
160 object with the same attribute values. The Job Description attributes that accumulate job
161 progress, such as "job-impressions-completed", "job-media-sheets-completed", and "job-k-
162 octets-processed", MUST be reset to 0 so that they give an accurate record of the job from its

163 restart point. The job object MUST continue to use the same "job-uri" and "job-id" attribute
 164 values.

165 **2.3.1 "job-hold-until" (type3 keyword | name(MAX)) operation attribute**

166 The client OPTIONALLY supplies this attribute. The IPP object MUST support this operation
 167 attribute in a Restart-Job request, if it supports the "job-hold-until" Job template attribute in
 168 create operations. See [ipp-mod] section 4.2.2. Otherwise, the IPP object NEED NOT support
 169 the "job-hold-until" operation attribute in a Restart-Job request. If supplied and supported, the
 170 IPP object copies the attribute to the Job object, replacing the job's previous "job-hold-until"
 171 attribute, if present, and makes the job a candidate for scheduling during the supplied named time
 172 period.

173 See Section 2.1.1 for the common semantics of the "job-hold-until" operation attribute for the
 174 Hold-Job operation and Restart-Job operation.

175 Note: In the future an OPTIONAL Modify-Job operation may be specified that allows the client
 176 to modify other attributes before releasing the restarted job.

177 The IPP object MUST accept or reject the request based on the job's current state, transition the
 178 job (or new job, depending on implementation) to the indicated new state, and return the
 179 indicated new "job-state" attribute and status code as follows:

Current "job-state"	New "job-state"	IPP object's response status code and action:
'pending'	'pending'	'client-error-not-possible'.
'pending-held'	'pending-held'	'client-error-not-possible'.
<u>'processing'</u>	<u>'pending'</u>	<u>OPTION 1: 'successful-ok' - job is started over.</u> <u>See Note 1.</u>
'processing'	'processing'	<u>OPTION 2: 'client-error-not-possible'.</u>
<u>'processing-stopped'</u>	<u>'pending'</u>	<u>OPTION 1: 'successful-ok' - job is started over.</u> <u>See Note 1</u>
'processing-stopped'	'processing-stopped'	<u>OPTION 2: 'client-error-not-possible'.</u>
'completed'	'pending'	'successful-ok' - job is started over.
'canceled'	'pending'	'successful-ok' - job is started over.
'aborted'	'pending'	'successful-ok' - job is started over.

180
 181 Note 1: For OPTION 1, the IPP object SHOULD indicate to clients that a Restart-Job operation
 182 is possible while the job is in the 'processing' and/or 'processing-stopped' states, by populating the
 183 job's "job-state-reasons" attribute with the 'job-restartable' value.

184 Note: Resetting the job progress attributes, allows job monitoring applications to function
 185 unchanged for a job that has been restarted. However, there is a problem for accounting

186 ~~applications that "pull" the job accounting data from the IPP object after the job completes using~~
187 ~~the Get Job Attributes or Get Jobs operations (or SNMP MIBs). Since the "job-id" and "job-uri"~~
188 ~~for the restarted job are the same as the original job and the accounting attributes are reset, the~~
189 ~~accounting program may not be able to detect that the job was restarted and is using additional~~
190 ~~resources. It is recommended that the Reprocess Job operation (see section 2.4) be used when~~
191 ~~accurate accounting data is desired to be made available to accounting programs that pull the data~~
192 ~~from the IPP Printer after the job completes, since a new job with a new "job-id" and "job-uri" is~~
193 ~~created while the old job remains for the accounting program to query accounting attributes. On~~
194 ~~the other hand, if an IPP object "pushes" the accounting data to the accounting application when~~
195 ~~the job completes, say, using event notification [ipp-not], then support of the Restart Job~~
196 ~~operation is not in conflict with such "pull" accounting.~~

197 *Access Rights:* The requesting user must either be the submitter of the job or an operator or
198 administrator of the Printer object. Otherwise, the IPP object MUST reject the operation and
199 return: 'client-error-forbidden', 'client-error-not-authenticated', or 'client-error-not-authorized' as
200 appropriate.

201 **3 The Job History concept and additional "job-state-reasons"**

202 This section explains the so-called Printer object's "job-history" that contains the recently
203 completed, canceled, and aborted jobs. This section also specifies the 'job-restartable' value of
204 the "job-state-reasons" Job Description attribute for use with the Restart-Job operations.

205 **3.1 The "job history" concept**

206 When a job is completed, canceled, or aborted, the IPP Printer object MAY retain the job with its
207 document data in a restartable condition using the Restart-Job operation for an implementation-
208 defined time period which may be zero seconds. If the IPP object supports the "job-state-
209 reasons" attribute and the Restart-Job operation, then it SHOULD indicate that such jobs are
210 restartable by adding the 'job-restartable' value to the job's "job-state-reasons" attribute (see
211 Section 3.2) during that implementation-defined time period.

212 After the implementation-defined restartable time period expires, the Printer object deletes the
213 document data for the job and the job becomes part of the "job history". The Print object MAY
214 also delete any number of the job attributes. Since the job is no longer restartable, the Printer
215 object MUST remove the 'job-restartable' value from the job's "job-state-reasons" attribute, if
216 supported. Clients are able to query jobs in the Printer object's "job history" using Get-Job-
217 Attributes and Get-Jobs operations.

218 Subsequently, the IPP Printer removes jobs from its "job history" in an implementation-defined
219 manner, such as after a fixed time period (which MAY be zero seconds) or when the number of
220 jobs exceeds a fixed number. Thereupon, the job can no longer be queried using the Get-Job-
221 Attributes and Get-Jobs operations and the IPP object returns the 'client-error-not-found' or
222 'client-error-gone' as appropriate.

223 **3.2 Add a new 'job-restartable' value to the "job-state-reasons" attribute**

224 The following new keyword value is specified for use with the "job-state-reasons" Job
225 Description attribute and the Restart-Job operation (see Section 2.3):

226 'job-restartable' - This job is currently able to be restarted using the Restart-Job operation.
227 With which job states this values is used depends on implementation, i.e., OPTION 1 vs.
228 OPTION 2 in the job state transition table in Section 2.3. Whenever the IPP object will
229 reject a Restart-Job operation for the job with the 'client-error-not-possible' error status
230 code because the job is not restartable, the IPP object MUST remove this value from the
231 job's "job-state-reasons" attribute.

232

233 For example, after a job is completed (job state is 'completed', 'aborted', or 'canceled), the
234 implementation MAY retain the job in a restartable condition for an implementation-
235 defined time period. When that time elapses, an implementation MAY delete the
236 document data, but MAY retain some or all of the job attributes as a "job history" for an
237 additional implementation-defined time period. During this second time period, the
238 implementation removes the 'job-restartable' value from the job's "job-state-reasons"
239 attribute, since the job can no longer be restarted.

240 **2.4Reprocess-Job**

241 ~~This operation allows a client to reprocess a copy of the job that is retained in the queue after~~
242 ~~processing is completed. A copy of the job restarts at the beginning on the same IPP Printer~~
243 ~~object with possibly different Job Template attributes supplied by the client in the request. Thus~~
244 ~~the Reprocess-Job operation is another create job operation and all of the semantics that [ipp-~~
245 ~~mod] specifies for "create job operations" also apply to the Reprocess-Job operation.~~

246 ~~The client MAY supply any Job Template attributes as in a create job operation whether they~~
247 ~~were originally supplied in the job create operation or not. The Printer object performs a~~
248 ~~validation as in a create operation of the job that would be made up of any supplied attributes~~
249 ~~replacing the corresponding job's attributes in combination with any of the job's remaining Job~~
250 ~~Template attributes. If the Printer object supports the new combination of Job Template~~
251 ~~attributes, the Printer object accepts the Reprocess-Job operation, creates a new job, assigns new~~
252 ~~"job-id" and "job-uri" values, and makes a copy of the job attributes with their new values.~~

253 ~~The IPP object initializes the Job Description attributes of the new job as in a create job~~
254 ~~operation, so that attributes such as "media-sheets-completed", and "job-k-octets-processed" start~~
255 ~~at 0 and the new job enters the 'pending' or 'pending-held' state, as after a job create operation.~~
256 ~~The returned groups are the same as for the Print-Job operation including the "job-id" and "job-~~
257 ~~uri" attributes with the new values assigned by the Printer object, whether the job has one or~~
258 ~~multiple documents.~~

259 ~~The Printer object leaves the old 'completed', 'canceled', or 'aborted' job as is and does not change~~
 260 ~~any of its attributes. Therefore, the Job Description attributes are preserved for job monitoring~~
 261 ~~and accounting purposes for the specified (old) job. Whether the document data is copied or~~
 262 ~~shared between the old and the new job, depends on implementation, and cannot be detected by~~
 263 ~~the client. Either the old job or the new job may be the target of subsequent Reprocess Job~~
 264 ~~operations.~~

265 ~~The IPP object MUST accept or reject the request based on the job's current state, transition the~~
 266 ~~job to the indicated new state, and return the indicated new job's new "job-state" attribute and~~
 267 ~~status code as follows:~~

Old job's current "job-state"	New job's new "job-state"	IPP object's response status code and action:
'pending'	n/a	'client-error-not-possible'
'pending-held'	n/a	'client-error-not-possible'
'processing'	n/a	'client-error-not-possible'
'processing-stopped'	n/a	'client-error-not-possible'
'completed'	'pending' or 'pending-held'	'successful-ok'
'canceled'	'pending' or 'pending-held'	'successful-ok'
'aborted'	'pending' or 'pending-held'	'successful-ok'

268 ~~*Access Rights:* The requesting user must either be the submitter of the job or an administrator of~~
 269 ~~the printer. Otherwise, the IPP object MUST reject the operation and return: 'client-error-~~
 270 ~~forbidden', 'client-error-not-authenticated', or 'client-error-not-authorized' as appropriate.~~

271 **34 Printer operations**

272 The printer operations in Set 1 are for use by operators and administrators of each Printer object.
 273 The following figure is copied from [ipp-mod] with the addition of the following arrow:

274 any****>

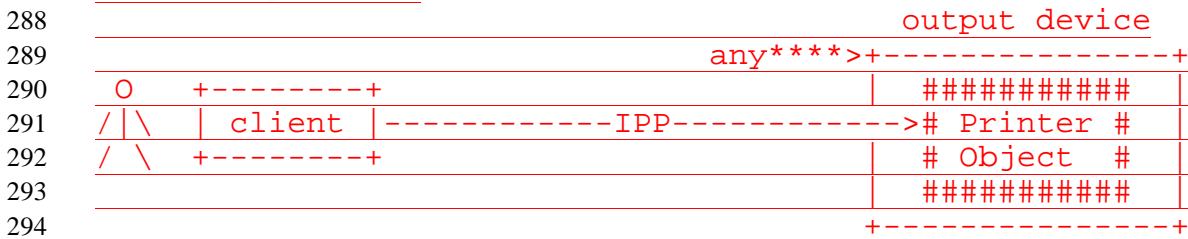
275 indicating an additional source of job submission using IPP or any other job submission protocol
 276 that passes to the device but does NOT pass through the IPP Printer object in question.

277 Legend:

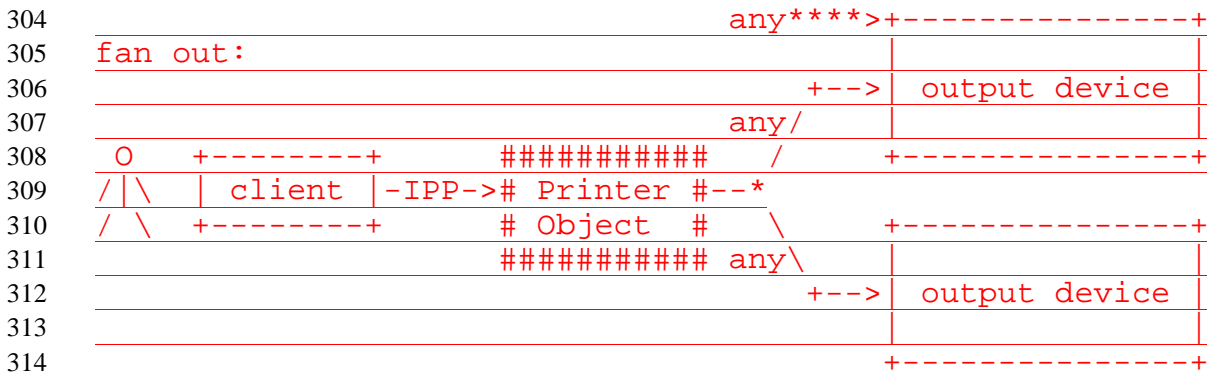
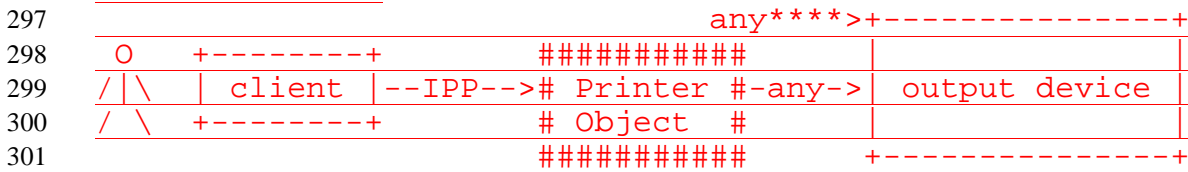
278
 279 ##### indicates a Printer object which is
 280 either embedded in an output device or is
 281 hosted in a server. The Printer object
 282 might or might not be capable of queuing/spooling.

284 any indicates any network protocol or direct
 285 connect, including IPP

286
 287 embedded printer:



295
 296 hosted printer:



315 The operation attributes for the Printer operation requests are as follows:-

316 Group 1: Operation Attributes

317 Natural Language and Character Set:

318 The "attributes-charset" and "attributes-natural-language" attributes as described in
319 section 3.1.4.1 of [ipp-mod].

320

321 Target:

322 The "printer-uri" (uri) operation attribute which is the target for this operation as
323 described in section 3.1.5 of [ipp-mod].

324

325 Requesting User Name:

326 The "requesting-user-name" (name(MAX)) attribute SHOULD be supplied by the client
327 as described in section 8.3 of [ipp-mod].

328 The operation attributes for the Printer operation responses are as follows:

329 Group 1: Operation Attributes

330 Status Message:

331 In addition to the REQUIRED status code returned in every response, the response
332 OPTIONALLY includes a "status-message" (text) operation attribute as described in
333 section 3.1.6 of [ipp-mod].

334

335 Natural Language and Character Set:

336 The "attributes-charset" and "attributes-natural-language" attributes as described in
337 section 3.1.4.2 of [ipp-mod].

338

339 Group 2: Unsupported Attributes

340 This is a set of Operation attributes supplied by the client (in the request) that are not
341 supported by the Printer object or that conflict with one another (see sections 3.2.1.2 and
342 16).

343

344 Group 3: Printer Object Attributes

345 Printer state information "~~printer-state~~" and "~~printer-state-reasons~~":

346 The Printer object MUST return the "printer-state" [ipp-mod 4.4.10] and, if supported, the
347 "printer-state-reasons" [ipp-mod 4.4.11] attributes in order to indicate the effect of the
348 operation on the Printer object.

349 **1.14.1 Pause-Printer**

350 This operation allows a client to stop the Printer object from scheduling jobs on all its devices.
 351 Depending on implementation, the Pause-Printer operation MAY also ~~and to~~ stop the Printer
 352 from processing the current job or jobs, ~~if supported~~. Any job that is currently being printed is
 353 either stopped as soon as the implementation permits or is completed, depending on
 354 implementation. The Printer object MUST still accept create operations to create new jobs, but
 355 MUST prevent any jobs from entering the 'processing' state.

356 If the Pause-Printer operation is supported, then the Resume-Printer operation MUST be
 357 supported, and vice-versa.

358 The IPP Printer stops the current job(s) on its device(s) that were in the 'processing' or
 359 'processing-stopped' states as soon as the implementation permits. If the implementation
 360 supports the "printer-state-reasons" attribute and the devices will take appreciable time to stop,
 361 the IPP Printer adds the 'moving-to-paused' value to the Printer object's "printer-state-reasons"
 362 attribute. When the device(s) have all stopped, the IPP Printer transitions the Printer object to the
 363 'stopped' state, removes the 'moving-to-paused' value, if present, and adds the 'paused' value to
 364 the Printer object's "printer-state-reasons" attribute.

365 When the current job(s) complete that were in the 'processing' state, the IPP Printer transitions
 366 them to the 'completed' state. When the current job(s) stop in mid processing that were in the
 367 'processing' state, the IPP Printer transitions them to the 'processing-stopped' state and, if the
 368 "job-state-reasons" attribute is supported, adds the 'printer-stopped' value to the job's "job-state-
 369 reasons" attribute.

370 Note: for any jobs that are 'pending' or 'pending-held', the 'printer-stopped' value of the jobs'
 371 "job-state-reasons" attribute also applies. However, the IPP Printer NEED NOT update those
 372 job's "job-state-reasons" attributes and only need return the 'printer-stopped' value when those
 373 jobs are queried (so-called "lazy evaluation").

374 Whether the Pause-Printer operation affects jobs that were submitted to the device from other
 375 sources than the IPP Printer object (see the any***> arrow in the figure above) in the same
 376 way that the Pause-Printer operation affects jobs that were submitted to the IPP Printer object
 377 using IPP, depends on implementation, i.e., on whether the IPP protocol is being used as a
 378 universal management protocol or just to manage IPP jobs, respectively.

379 The IPP Printer MUST accept the request in any state, transition the Printer to the indicated new
 380 "printer-state" before returning, and return the indicated "printer-state", "printer-state-reasons",
 381 and status code as follows:

Current "printer-state"	New "printer-state" returned	"printer-state-reasons"	IPP Printer's response status code and action:
'idle'	'stopped'	'paused'	'successful-ok'
'processing'	'processing'	'moving-to-	OPTION 1: 'successful-ok';

		paused'	Later, when all output has stopped, the "printer-state" becomes 'stopped', and the 'paused' value replaces the 'moving-to-paused' value in the "printer-state-reasons" attribute
'processing'	'stopped'	'paused'	OPTION 2: 'successful-ok'; all output stopped immediately
'stopped'	'stopped'	'paused'	'successful-ok'

382 *Access Rights:* The requesting user must be an [operator or administrator](#) of the [Printer object](#).
 383 Otherwise, the IPP Printer MUST reject the operation and return: 'client-error-forbidden', 'client-
 384 error-not-authenticated', or 'client-error-not-authorized' as appropriate.

385 **1.1.14.1.1 Add a new 'moving-to-paused' value to the "printer-state-reasons" attribute**

386 The following new keyword value is specified for use with the "printer-state-reasons" Printer
 387 Description attribute:

388 'moving-to-paused': ~~Someone~~ [The Printer object's operator or administrator](#) has paused the
 389 Printer object [using the Pause-Printer operation or other means](#), but it has not yet stopped
 390 producing output. When all the devices stop producing output, the Printer object MUST
 391 replace this value with the 'paused' value.

392 **1.24.2 Resume-Printer**

393 This operation allows a client to resume the Printer object scheduling jobs on all its devices. If
 394 the Printer object supports the "printer-state-reasons" attribute, it MUST remove the 'paused' and
 395 'moving-to-paused' values from the Printer object's "printer-state-reasons" attribute, if present. If
 396 there are no other reasons to keep a device paused (such as media-jam), the IPP Printer
 397 transitions itself to the 'processing' or 'idle' states, depending on whether there are jobs to be
 398 processed or not, respectively, and the device(s) resume processing jobs.

399 If the Pause-Printer operation is supported, then the Resume-Printer operation MUST be
 400 supported, and vice-versa.

401 The IPP Printer removes the 'printer-stopped' value from any job's "job-state-reasons" attributes
 402 contained in that Printer.

403 The IPP Printer MUST accept the request in any state, transition the Printer object to the
 404 indicated new state, and return the indicated "printer-state" and status code as follows:

Current "printer-state"	New "printer-state" returned	IPP Printer's response status code and action:
'idle'	'idle'	'successful-ok'
'processing'	'processing'	'successful-ok'
'stopped'	'processing'	'successful-ok';

		when there are jobs to be processed
'stopped'	'idle'	'successful-ok'; when there are no jobs to be processed.

405 *Access Rights:* The requesting user must be an operator or administrator of the Printer object.
 406 Otherwise, the IPP Printer MUST reject the operation and return: 'client-error-forbidden', 'client-
 407 error-not-authenticated', or 'client-error-not-authorized' as appropriate.

408 **1.34.3 Purge-Jobs**

409 This operation allows a client to remove all jobs from an IPP Printer object, regardless of their
 410 job states, including jobs in the Printer object's "job-history" (see Section 3.1). After a Purge-
 411 Jobs operation has been performed, a Printer object MUST return no jobs in subsequent Get-Job-
 412 Attributes and Get-Jobs responses (until new jobs are submitted).

413 Whether the Purge-Jobs (and Get-Jobs) operation affects jobs that were submitted to the device
 414 from other sources than the IPP Printer object (see the any***> arrow in the figure in Section
 415 4) in the same way that the Purge-Jobs operation affects jobs that were submitted to the IPP
 416 Printer object using IPP, depends on implementation, i.e., on whether the IPP protocol is being
 417 used as a universal management protocol or just to manage IPP jobs, respectively.

418 Note: if an operator wants to cancel all jobs without clearing out the job history, the operator
 419 uses the Cancel-Job operation on each job instead of using the Purge-Job operation.

420 The Printer object MUST accept this operation in any state and transition the Printer object to the
 421 'idle' state.

422 *Access Rights:* The requesting user must be an operator or administrator of the Printer object.
 423 Otherwise, the IPP object MUST reject the operation and return: client-error-forbidden, client-
 424 error-not-authenticated, and client-error-not-authorized as appropriate.

425 **5 Security Considerations**

426 For the job operations in Set 1 (Section 2), the requesting user must either be the submitter of the
 427 job or an operator or administrator of the Printer object (see [ipp-mod] Section 1). Otherwise,
 428 the IPP object MUST reject the operation and return: 'client-error-forbidden', 'client-error-not-
 429 authenticated', or 'client-error-not-authorized' as appropriate. See [ipp-mod] Section 8.3 on the
 430 two ways that the client MUST specify the user who is performing each IPP operation.

431 For the printer operations in Set 1 (Section 4), the requesting user must by an operator or
 432 administrator of the Printer object (see [ipp-mod] Section 1). The means for authorizing an
 433 operator or administrator of the Printer object are not specified in either [ipp-mod] or this
 434 document.

435 **46 References**

436 [ipp-mod]

437 Isaacson, S., deBry, R., Hastings, T., Herriot, R., Powell, P., "Internet Printing
438 Protocol/1.0: Model and Semantics" draft-ietf-ipp-mod-10.txt, June, 1998.439 ~~[ipp-not]~~440 ~~Isaacson, S., Martin, J., deBry, R., Hastings, T., "IPP Event Notifications (Very Short)"~~
441 ~~<ipp-notifications-very-short-980701.doc>, July 1, 1998.~~

442 [ipp-pro]

443 Herriot, R., Butler, S., Moore, P., Tuner, R., "Internet Printing Protocol/1.0: Encoding
444 and Transport", draft-ietf-ipp-pro-06.txt, June, 1998.445 ~~[ISO-10175]~~446 ~~ISO/IEC 10175 Document Printing Application (DPA), June 1996.~~