

1 Subj: IPP Bake Off 2 Issues
2 From: Peter Zehler, Tom Hastings, and Bob Herriot
3 File: Issues-raised-at-Bake-Off2.doc
4 Version: 1.2
5 Date: 3/22/1999
6

7 We've taken the issues that Peter published in the Bake Off 2 Summary and started a separate file.
8 We've add some additional information that we gathered at the Bake Off with the people raising the
9 issues. We've also added to each issue, either a list of "possible alternatives" or a "suggested
10 clarification", "suggested change", or "suggested addition" for the discussion, so that we can reach
11 agreement as soon as possible. Please feel free to add additional alternatives or disagree with our
12 suggested clarifications or additions via e-mail so that the group may have the widest possible set of
13 alternatives to choose from. All the additional material is indicated with revision marks from the issues
14 list that Peter Zehler published last week.

15 **TABLE OF CONTENTS**

16	1) ISSUE: Is 'application/octet-stream REQUIRED?	3
17	Suggested change:	3
18	2) ISSUE: How can client force identified mode?	3
19	Possible alternatives:	3
20	3) ISSUE: How reject down stream auto-sensed unsupported PDL?	3
21	Suggested addition (similar addition for "compression" in Issue 6):.....	4
22	4) ISSUE: Client closes slow channel	4
23	Suggested clarification (same as Issues 5 and 20):.....	4
24	5) ISSUE: Client closes stopped device	4
25	Suggested clarification (same as Issues 4 and 20):.....	5
26	6) ISSUE: What error if wrong compressed data supplied?	5
27	Suggested addition (similar addition for document-format in Issue 3; see related Issue 28):.....	5
28	7) ISSUE: Please implement Manufacturer make and model printer attribute and send the .INF file	
29	model name of the printer.	5
30	Suggested clarification for the IIG:	5
31	8) ISSUE: In IPP/1.0 Model and semantics 3.2.6.1, the definition for "limit", "which-jobs" and "my-	
32	jobs" is contradicting each other.	6
33	Suggested clarification:	6
34	9) ISSUE: Customers become very unhappy when they go to the printer to pick up their job and a ream	
35	of PostScript source code is sitting in the output bin.	6
36	Suggested clarification:	6
37	10) ISSUE: How distinguish between submit vs processing auto-sense?	7
38	Suggested clarification in [ipp-mod] and [ipp-iig]:	7
39	11) ISSUE: Return what attributes with document-format-not-supported?	7
40	Suggested clarification (see also Issues 18 and 23):.....	7
41	12) ISSUE: length fields for the "UNSUPPORTED" tag	8
42	Suggested clarification (same as Issue 15):	8
43	13) ISSUE: What job-state value should be returned in the Create-Job response?	8
44	Suggested clarification:	8
45	14) ISSUE: Job-state for a forwarding server?	9
46	Suggested addition:	9
47	15) ISSUE: 'unknown' and 'unsupported' Out of band values.	9

48 Suggested clarification (same clarification as Issue 12):9

49 16) ISSUE: Get-Printer-Attributes Polling 10

50 Suggested clarification in the IIG:10

51 17) ISSUE: Client display of absolute time for job attributes? 10

52 Possible alternatives:10

53 18) ISSUE: Return all errors on Print-Job fidelity=true 11

54 Suggested clarification (same clarification as Issue 27):11

55 19) ISSUE: User Performing the Send-Document Operation 11

56 Suggested clarification:11

57 20) ISSUE: Non-spooling printers accept/reject additional jobs 11

58 Suggested clarification (same as Issues 4 and 5):11

59 21) ISSUE: Does 'none' "uri-security-supported" mean Basic/Digest? 12

60 Suggested clarification:12

61 22) ISSUE: Status code on variable-length attributes that are 'too short' 12

62 Suggested clarification in the IIG:12

63 23) ISSUE: There seems to be some misunderstanding about the unsupported-attributes group. 13

64 Suggested clarification (related to Issues 11 and 18):13

65 24) ISSUE What status does Get-Jobs return when no jobs? 13

66 Suggested clarification:13

67 25) ISSUE - MAY an IPP object return more Operation attributes? 13

68 Suggested clarification:13

69 26) ISSUE: MAY an IPP object return additional groups? 13

70 Suggested clarification:14

71 27) ISSUE: Return first or all unsupported attributes in Unsupported Group? 14

72 Suggested clarification (same clarification as Issue 18):14

73 28) ISSUE: What if compression is supplied but not supported? 14

74 Possible Alternatives (related to Issues 3 and 6):14

75 29) ISSUE: Should "queued-job-count" be REQUIRED? 15

76 Suggested change:15

77

78 1) ISSUE: Is 'application/octet-stream REQUIRED?

79 Is application/octet-stream REQUIRED. IPP/1.0 appears not to require it, while IPP/1.1 indicates
80 "REQUIRED".

81 Suggested change:

82 Change IPP/1.1 Model and Semantics document back to agree with IPP/1.0 not to require support of the
83 'application/octet-stream' document format.

84 2) ISSUE: How can client force identified mode?

85 If an IPP Printer supports both authenticated and unauthenticated access, there is no way for a client to
86 force itself to be authenticated, i.e., be in identified mode, since it is the server that forces authentication
87 by issuing a challenge to the client. It is very useful for a client to be able to get into identified mode as
88 soon as possible. Today you have to wait to be challenged by the server, which may never happen – or
89 happens at an unpredictable time. The security conformance requires that the authentication for
90 operations be the same for all operations. So for authenticated Cancel-Job, the Print-Job has to be
91 authenticated as well. We would like to add another operation that forces the server to generate a 401
92 authentication challenge which the client would submit before submitting the print job in the first place.
93 Unless somebody has a different solution (Microsoft)

94 Possible alternatives:

- 95 1. Add the operation as an OPTIONAL operation to IPP/1.0 and IPP/1.1 that forces the IPP object to
96 issue a challenge to the client.
- 97 2. Use two URLs for the same IPP Printer object, one requires authentication and the IPP server always
98 issues a challenge and the other never does. So the client that wants to be authenticated submits
99 requests to the URL that requires authentication. ISSUE: How does the client discover which URL
100 to use, since "uri-security-supported" is about security, not authentication?
- 101 3. Use two IPP Printer objects that fan-in to the same device. One IPP Printer object requires
102 authentication and always issues the challenge and the other never does. ISSUE: How does the
103 client discover which IPP Printer to use for authenticated access?
- 104 4. Request that the HTTP WG add some kind of header that allows the client to request that the HTTP
105 server issue a challenge. ISSUE: It is unlikely that the HTTP group would do such a thing, since it
106 is not needed for the usual use of HTTP which is to access documents on a server.

107 3) ISSUE: How reject down stream auto-sensed unsupported PDL?

108 If auto-sensing happens AFTER the job is accepted (as opposed to auto-sensing at submit time before
109 returning the response), what does the implementation do?

110 Presumably, it is similar to encountering a mal-formed PDL. So the implementation aborts the job, puts
111 the job in the 'aborted' state and sets the 'aborted-by-system' value in the job's "job-state-reasons", if
112 supported. If the "job-state-reasons" attribute is supported, the 'aborted-by-system' value seems
113 appropriate, but it would be good to have a more specific reason to indicate the reason that the job was
114 aborted by the system.

115 ***Suggested addition (similar addition for "compression" in Issue 6):***

116 Add 'unsupported-document-format' as a "job-state-reasons" value for use when the job is aborted
117 because the document format that is auto-sensed is not a supported document format. Also add a
118 'document-format-error' as a "job-state-reasons" value for use when the job is aborted because any kind
119 of PDL error is encountered while processing the document.

120 **4) ISSUE: Client closes slow channel**

121 Some IPP Printer implementations, such as forwarding servers, want to accept an IPP job, even though
122 the down stream channel is being used at the moment by another job stream that the device supports.
123 Rejecting the job would mean that an IPP job might never get in, since these other protocols queue the
124 request.

125 However, some clients close the channel when it is flowed controlled off for too long a time?

126 ***Suggested clarification (same as Issues 5 and 20):***

127 Clarify the IPP/1.1 Model and Semantics document that Clients MUST NOT close the channel when
128 flowed controlled off. Clients SHOULD do Get-Printer-Attributes and determine state of the device.
129 Alert user if the printer is stopped. Let user decide whether to abort the job transmission or not.

130 Also clarify the IPP/1.1 Model and Semantics document that the following actions are conforming for
131 non-spooling IPP Printer objects: After accepting a create job operation, a non-spooling IPP Printer
132 MAY either:

- 133 1. Reject any subsequent create job operations while it is busy transferring and/or processing an
134 accepted job request and return the 'server-error-busy (0x0507).
- 135 2. Accept up to some implementation-defined subsequent create job operations and flow control
136 them to prevent buffer overflow. When the implementation-defined number of jobs is exceeded,
137 the IPP Printer MUST return the 'server-error-busy' status code and reject the create job request
138 as in 1 above.

139 Client MUST NOT close the channel when flow controlled off. Clients that are rejected with a 'server-
140 error-busy' status code MAY retry periodically, try another IPP Printer, and/or subscribe for a 'ready-for-
141 job' event when we have notification specified. **5) ISSUE: Client closes stopped device**

142 When a non-spooling printer is accepting data and putting it on media and runs into a problem, such as
143 paper out or paper jam, what should it do?

144 Returning an error is not user friendly, if fixing the problem would allow the job to complete normally.

145 ***Suggested clarification (same as Issues 4 and 20):***

146 Clarify the IPP/1.1 Model and Semantics document that IPP Printers MUST not return an error status
147 code during a Print-Job operation when a device problem, such as jam or out of paper. Instead, the IPP
148 Printer object flow controls the data off. Otherwise, only a partial job will be produced, when a whole
149 job would be produced when the problem is attended to.

150 Clients MUST not close the channel when flow controlled off. Clients SHOULD do Get-Printer-
151 Attributes and determine state of the device. Alert user if the printer is stopped. Let user decide whether
152 to abort the job transmission or not.

153 **6) ISSUE: What error if wrong compressed data supplied?**

154 Problem: IPP server supports 'deflate' and 'gzip'. If client sets "compression attribute" = 'deflate' but
155 sends gzipped data, what error does IPP server return to client? Cannot use the existing 'client-error-
156 attributes-or-values-not-supported' (0x040B). But returning the operation attribute with the value that
157 was sent ('deflate') would be incorrect, because 'deflate' is supported!

158 ***Suggested addition (similar addition for document-format in Issue 3; see related Issue***
159 ***28):***

160 Add a new error status code: 'client-error-compression-error' that the IPP object can return if the
161 compression error is detected before the create job response is returned. Also add 'compression-error' as
162 a "job-state-reason" value for use when the job is aborted because any kind of compression error is
163 detected while decompressing the data after the create job response has been returned to the client.

164 **7) ISSUE: Please implement Manufacturer make and model printer**
165 **attribute and send the .INF file model name of the printer.**

166 If you do this we will automatically install the correct driver (if we have it) (Microsoft)

167 ***Suggested clarification for the IIG:***

168 At the front of the Implementer's Guide, indicate that implementation considerations that relate to
169 particular operating system and NOS will be incorporated as they become known. Add recommendation
170 to the IPP/1.1 Implementer's Guide that printer vendors are encouraged to configure the IPP Printer's
171 "printer-make-and-model" attribute with the make and model name that matches the .INF file on
172 Microsoft platforms. When so configured, the Microsoft driver install program will skip asking the user
173 for the make and model of the printer being installed and use the value of the "printer-make-and-model"
174 attribute.

175 **8) ISSUE: In IPP/1.0 Model and semantics 3.2.6.1, the definition for "limit",**
176 **"which-jobs" and "my-jobs" is contradicting each other.**

177 The problem is that the definition for "which-jobs" and "my-jobs" states that "all" jobs MUST be
178 returned, while "limit" restricts the number of jobs to be returned. (Stefan Andersson Axis
179 Communication AB)

180 ***Suggested clarification:***

181 Clarify IPP/1.1 Model and Semantics "which-jobs" and "my-jobs" operation attributes to indicate that
182 the number of jobs returned is limited by the "limit" attribute if supplied by the client.

183 **9) ISSUE: Customers become very unhappy when they go to the printer to**
184 **pick up their job and a ream of PostScript source code is sitting in the**
185 **output bin.**

186 Cause: A PostScript datastream is accidentally sent to a PCL printer.

187 IPP Issue: IPP needs to clarify the standard in section 3.2.1.1 of the Model and Semantics document.
188 Lines 1219-1221 defining the "document-format" operation attribute state that:

189 If the client does not supply the [document format] attribute, the Printer object assumes that the
190 document data is in the format defined by the Printer object's "document-format-default"
191 attribute.

192 I would like to see the following clarification:

193 If the client does not supply the [document format] attribute and the Printer object is not able to
194 auto-sense the document format at print-job request time, the Printer object assumes that the
195 document data is in the format defined by the Printer object's "document-format-default"
196 attribute.

197 If the Printer object senses that the document format is PostScript, then job should be rejected if it is
198 being sent to a PCL-only printer. The 'application/octet-stream' mechanism discussed in section 4.1.9
199 does not seem to be helpful in this case, because it appears to assume that the auto-sensing occurs at
200 document processing time. Until the document is actually "ripped", the document format remains
201 unknown. So it seems to me that lines 2453-2476 do not address the problem described above where the
202 wrong document format is submitted. These lines, rather, seem to apply to the case of a printer that
203 handles multiple document formats and assumes that the submitted document is in one of the supported
204 formats.

205 ***Suggested clarification:***

206 Add the suggested clarification that auto-sensing MAY be done at either job-submission time and/or job
207 processing time to the IPP/1.1 Model and Semantics documents. ISSUE: Still need to talk to proposer
208 of this issue, since the "document-format-default" should be set to 'application/octet-stream' if the default
209 is to auto-sense.

210 **10) ISSUE: How distinguish between submit vs processing auto-sense?**

211 There are two different implementations of auto-sensing:

- 212
- at print submit time BEFORE the Print-Job or Send-Document responds
 - at document processing (ripping) time AFTER the Print-Job or Send-Document has accepted the
213 job and returned the response.
214

215 The description of 'application/octet-stream' doesn't clarify whether one, the other or both is meant. How
216 can a client determine which is supported?

217 ***Suggested clarification in [ipp-mod] and [ipp-iig]:***

218 Clarify IPP/1.1 Model and Semantics document that 'application/octet-stream' means either auto-sensing
219 at job submission time and/or job processing time depending on implementation. Add to Implementer's
220 Guide a discussion about the advantages of auto-sensing at job submit time, rather than waiting until job
221 processing time, so that an IPP Printer can reject an unsupported document format instead of accepting
222 the job and then aborting the job sometime later. Also discuss for print by reference that an IPP Printer
223 may want to examine the file, at least the first few octets, in order to check that the document-format is
224 supported. On the other hand, network delays may make such a strategy take too long. Alternatively,
225 the client may want to supply the "document-format" explicitly when doing print-by-reference either
226 using the file extension as a hint, or actually accessing the first few octets of the data an implementing an
227 auto-sensing in the client.

228 **11) ISSUE: Return what attributes with document-format-not-supported?**

229 If a server receives a request with a document format which is not supported, it returns the client-error-
230 document-format-not-supported (0x040A) status code. Is it also necessary to include document format
231 in the unsupported attribute group?

232 We suggest adding text which says it need not be supplied in the unsupported group.

233 ***Suggested clarification (see also Issues 18 and 23):***

234 Clarify IPP/1.1 Model and Semantics document that when returning the 'client-error-document-format-
235 not-supported' in a create response or a Send-Document response, that the "document-format" attribute
236 and the supplied value NEED NOT be returned in the Unsupported Attributes group. If there are also
237 some unsupported Job Template attributes supplied in the create request, the IPP Printer MAY, but
238 NEED NOT, return them in the Unsupported Attributes Group when returning the 'client-error-
239 document-format-not-supported', since the document-format error is a higher precedence error and the
240 document is not going to be able to be processed at all on the Printer.

241 **12) ISSUE: length fields for the "UNSUPPORTED" tag**

242 IPP/1.0: Model and Semantics, 16 Nov 1998, 3.2.1.2, Group 2 (unsupported attributes) -- states that in
243 the case of an unsupported attribute name, the printer object should return a substituted out of band value
244 of "unsupported". This impression is strengthened by the reference to section 4.1, where it gives the legal
245 out of band values, none of which is an empty string.

246 This appears to conflict with Internet Printing Protocol/1.0: Encoding and Transport, 16 Nov 1998,
247 section 3.10, where it states that the value length must be 0 and the value empty. (Claudio Cordova,
248 Wade Mergenthal Xerox Corp.)

249 ***Suggested clarification (same as Issue 15):***

250 Clarify the IPP/1.1 Model and Semantics document so that it does not appear to contradict the Encoding
 251 and Transport document. However, whether each of the "out-of-band" values are encoded as distinct
 252 attribute syntaxes with no value or as a single attribute syntax with a value that indicates which out-of-
 253 band value, is purely an encoding matter and cannot be indicated in the Model and Semantics document.
 254 Therefore, indicate in the IPP/1.1 Model and Semantics document that the reader is to refer to the
 255 IPP/1.1 Encoding and Transport document for the encoding of the out-of-band values.

256 **13) ISSUE: What job-state value should be returned in the Create-Job**
 257 **response?**

258 Pending, pending-held, or either depending on implementation?

259 The problem with 'pending' is that the job is not a "candidate to start processing" as the definition states.
 260 The 'pending-held' state seems more reasonable. Its definition is:

261 'pending-held': The job is not a candidate for processing for any number of reasons but will
 262 return to the 'pending' state as soon as the reasons are no longer present. The job's "job-state-
 263 reason" attribute MUST indicate why the job is no longer a candidate for processing.

264 Also there is a "job-state-reason" value 'job-incoming' which states:

265 'job-incoming': The Create-Job operation has been accepted by the Printer, but the Printer is
 266 expecting additional Send-Document and/or Send-URI operations and/or is accessing/accepting
 267 document data.

268 But "job-state-reasons" is OPTIONAL. Do we mandate it or recommend it if supporting Create-Job?

269 ***Suggested clarification:***

270 Clarify the IPP/1.1 Model and Semantics document that an IPP Printer MAY put the job into the
 271 'pending' or 'pending-held' states after a Create-Job, depending on implementation as follows:

- 272 • 'pending' - if the job is a candidate for processing whether all of the document data is present or
 273 not. Add the 'waiting-for-data' "job-state-reasons" value to the job as an indication why this
 274 'pending' job is not being processed OR
- 275 • 'pending-held' - if the job is not a candidate for processing until the last Send-Document or Send-
 276 URI operation has been performed with the "last-document" set to 'true' and the document data
 277 transferred. Here the implementation SHOULD support the "job-state-reasons" and use the 'job-
 278 incoming' until the last data has arrived. The IPP Printer removes the 'job-incoming' value when
 279 the last data has arrived, and transitions the job from the 'pending-held' to the 'pending' job state.

280 Note: Change the bo38.test script so that either the 'pending-held' or the 'pending' job state is expected
 281 after a Create-Job operation.

282 14) ISSUE: Job-state for a forwarding server?

283 What job-state value should be returned in the Print-Job response for an IPP object that forwards the
284 data over a one-way interface, such as a parallel port or LPD? pending, processing, completed, or
285 unknown?

286 Unknown is the strict interpretation of section 4.3.7 "job-state", but it isn't very user friendly. The "job-
287 state" SHOULD reflect the actual job state, but these implementations have no idea when the job
288 actually starts or finishes.

289 How about a new "job-state-reasons" value: 'queued-in-device' (from PWG Job Monitoring MIB)?

290 Suggested addition:

291 Add to the IPP/1.1 Model and Semantics document the 'queued-in-device' value for use with the "job-
292 state-reasons" attribute. RECOMMEND that an implementation that forwards jobs, but does not have
293 any means to query the state of the down stream job, support the "job-state-reasons" attribute and the
294 new 'queued-in-device' value when returning the job in the 'completed' state.

295 15) ISSUE: 'unknown' and 'unsupported' Out of band values.

296 It is very unclear from the spec as to whether or not you should use the word 'unknown' (or unsupported
297 in that case) as the value for attributes that are unknown.

298 You can read it that you set the length equal to zero and set the type to 'unknown'. You can also read it
299 as saying you set the value to the string 'unknown'.

300 This is not helped by the Transport and Encoding spec saying – you must set the length to zero and then
301 telling a client what to do with a non-zero length. (Microsoft)

302 Suggested clarification (same clarification as Issue 12):

303 Clarify the IPP/1.1 Model and Semantics document so that it does not appear to contradict the Encoding
304 and Transport document. However, whether each of the "out-of-band" values are encoded as distinct
305 attribute syntaxes with no value or as a single attribute syntax with a value that indicates which out-of-
306 band value, is purely an encoding matter and cannot be indicated in the Model and Semantics document.
307 Therefore, indicate in the IPP/1.1 Model and Semantics document that the reader is to refer to the
308 IPP/1.1 Encoding and Transport document for the encoding of the out-of-band values.

309 16) ISSUE: Get-Printer-Attributes Polling

310 Some client polls printer periodically by Get-Printer-Attributes without specifying "requested-attributes".
311 So printer has to reply all attributes. It consumes printer resource.

312 Suggested clarification in the IIG:

313 RECOMMEND in the IPP/1.1 Implementer's Guide that Clients should specify "requested-attributes", if
314 it wants to get just the printer status.

315 **17) ISSUE: Client display of absolute time for job attributes?**

316 What are clients doing with printers that don't support absolute time? How can client display an absolute
317 time that a job was submitted, started processing, and completed (which is what is useful for a user)?

318 Possible Solution

319 Get Uptime from printer ("printer-up-time" - time system has been up in seconds)

320 Get Job(s)

321 Calculate Display time = job tick time ("time-at-xxx" - in seconds that system has been up) – uptime
322 ("printer-up-time") + local client absolute time. The down side is that the client has to get the "printer-
323 up-time" every time with a separate Get-Printer-Attributes operation.

324 Alternatively: Add OPTIONAL job attributes: “date-time-at-creation (dateTime)”, “date-time-at-
325 processing (dateTime)”, and “date-time-at-completion (dateTime)”

326 (Microsoft)

327 ***Possible alternatives:***

328 Clarify that the "time-at-xxx" attributes can be negative if an IPP Printer is re-booted while jobs remain.

- 329 1. Add to the IPP/1.1 Model and Semantics document OPTIONAL job description attributes: “date-
330 time-at-creation (dateTime)”, “date-time-at-processing (dateTime)”, and “date-time-at-completion
331 (dateTime)”.
- 332 2. Return "printer-up-time" (in seconds) as an operation attribute in Get-Jobs and Get-Job-Attributes
333 response.
- 334 3. Make the "printer-up-time" Printer Description attribute also be a Job Description attribute. Clients
335 that request the "time-at-xxx" job attributes should also request the "printer-up-time" job attribute, so
336 that they can avoid requesting it using a separate Get-Printer-Attributes request.

337 **18) ISSUE: Return all errors on Print-Job fidelity=true**

338 If ipp-attributes-fidelity=true, MUST all attributes that are not supported, be returned, or can just the first
339 error be returned? Section 16.3 and 16.4 of the Model and Semantics document was moved to the
340 Implementer's Guide when creating the November 1998 draft from the June 1998 draft. The following
341 note was contained in section 16.4 that was moved:

342 Note: whether the request is accepted or rejected is determined by the value of the "ipp-attribute-fidelity"
343 attribute in a subsequent step, so that all Job Template attribute supplied are examined and all
344 unsupported attributes and/or values are copied to the Unsupported Attributes response group.

345 ***Suggested clarification (same clarification as Issue 27):***

346 Clarify in the IPP/1.1 Model and Semantics document that all operation attributes and all Job Template
347 attributes **MUST** be returned in the Unsupported Attributes group, unless there is a specific error status,
348 such as 'client-error-document-not-supported'.

349 **19) ISSUE: User Performing the Send-Document Operation**

350 The Send-Document and Send-URI commands need the following clarification with regard to the user
351 performing the operation. In the requesting-user-name section of Send-Document add:

352 The user performing the Send-Document operation must be the same as for the Create- Job
353 operation that created the job. The printer determines the user performing the operation from the
354 requesting-user-name or the underlying authentication mechanism as described in Section 8.3 of
355 the model document.

356 The wording in the Send-URI section would imply that the above change applies to Send-URI as well.

357 ***Suggested clarification:***

358 Add the suggested clarification to the IPP/1.1 Model and Semantics document.

359 **20) ISSUE: Non-spooling printers accept/reject additional jobs**

360 Some IPP Printer implementations reject a second Print-Job (or Create-Job) while they are processing a
361 Print-Job. Other IPP Printer implementations, such as forwarding servers and non-spooling printers,
362 accept some number of subsequent jobs, but flow control them off until the first job is finished.

363 ***Suggested clarification (same as Issues 4 and 5):***

364 Also clarify the IPP/1.1 Model and Semantics document that the following actions are conforming for
365 non-spooling IPP Printer objects: After accepting a create job operation, a non-spooling IPP Printer
366 **MAY** either:

- 367 • Reject any subsequent create job operations while it is busy transferring and/or processing an
368 accepted job request and return the 'server-error-busy (0x0507).
- 369 • Accept up to some implementation-defined subsequent create job operations and flow control
370 them to prevent buffer overflow. When the implementation-defined number of jobs is exceeded,
371 the IPP Printer **MUST** return the 'server-error-busy' status code and reject the create job request
372 as in 1 above.

373 Client **MUST NOT** close the channel when flow controlled off. Clients that are rejected with a 'server-
374 error-busy' status code **MAY** retry periodically, try another IPP Printer, and/or subscribe for a 'ready-for-
375 job' event when we have notification specified.

376

377 **21) ISSUE: Does 'none' "uri-security-supported" mean Basic/Digest?**

378 Section 4.4.2 "uri-security-supported" 'none' values says:

379 'none': There are no secure communication channel protocols in use for the given URI.

380 Should be clarified that the REQUIRED Basic and Digest are intended for the 'none' value. (Hugo Parra)

381 ***Suggested clarification:***

382 Instead, clarify that the "uri-security-supported" is only referring to the privacy part of security, not the
383 authentication part, such as HTTP Basic and Digest authentication. Add a note to both the "uri-security-
384 supported" attribute and Section 5.4 on Security Conformance Requirements in the IPP/1.1 Model and
385 Semantics that authentication conformance requirements are specific to a transport, such as HTTP Basic
386 and Digest, and are specified in the Encoding and Transport [ipp-pro] document.

387 **22) ISSUE: Status code on variable-length attributes that are 'too short'**

388 IPP defines a status code 'client-error-request-value-too-long' for a variable-length attribute that exceeds
389 the maximum length allowed by the attribute. However, it is not clear what status code to use in the
390 opposite case, i.e. the supplied attribute value is shorter than the requirement. In the current spec, this
391 problem will arise when a 0-length value is supplied in 'keyword' attributes. In this case, should the
392 request be rejected with status code 'client-error-request-value-too-long' or 'client-error-bad-request' ?

393 Furthermore, if "ipp-attribute-fidelity" is 'false', should the request be rejected at all? (Jason Chien-Hung
394 Chen)

395 ***Suggested clarification in the IIG:***

396 No special status code is needed and no special action is needed by the IPP object. Since this is a
397 keyword, its value needs to be compared with the supported values. Assuming that the printer doesn't
398 have any values in its corresponding "xxx-supported" attribute that are keywords of zero length, the
399 comparison will fail. Then the request will be accepted or rejected depending on the value of "ipp-
400 attributes-fidelity" being 'false' or 'true', respectively. No change to the [ipp-mod]. Indicate this handling
401 of too short keywords in the IIG. All other variable length attribute syntaxes have a minimum greater
402 than 0.

403 **23) ISSUE: There seems to be some misunderstanding about the**
404 **unsupported-attributes group.**

405 Some implementations return all the attributes that are in the spec that their implementation does not
406 support in the Unsupported Attributes group on a get-attributes operation, independent of the attributes
407 that were actually requested. The unsupported-attributes presumably contains all the attributes the
408 implementation knows about but does not support. I do not believe this is the proper use of the
409 unsupported-attributes group. Do we need a clarification in the specification.

410 ***Suggested clarification (related to Issues 11 and 18):***

411 Clarify IPP/1.1 Model and Semantics document that only attributes (operation, Job Template, ...)
412 supplied in the request by the client that the IPP object does not support are returned in the Unsupported
413 Attributes group.

414 **24) ISSUE What status does Get-Jobs return when no jobs?**

415 Should Get-Jobs return 'successful-ok' when there are no jobs to be returned? The client can see that the
416 Jobs group contains no jobs from the response. Returning an error may confuse the client. Some
417 implementations returned 'client-error-not-found' error code.

418 ***Suggested clarification:***

419 Clarify IPP/1.1 Model and Semantics document that the IPP Printer MUST return 'successful-ok' even
420 when there are no jobs to return. The operation is successful and the client will see that there are no
421 returned jobs.

422 **25) ISSUE - MAY an IPP object return more Operation attributes?**

423 Is it ok for an IPP object to return additional operation attributes in a response (as an extension to the
424 standard)? If so, then the client MUST ignore or do something with them. (Hugo Parra)

425 ***Suggested clarification:***

426 Clarify IPP/1.1 Model and Semantics document that the client MUST ignore or do something with
427 additional operation attributes returned than are in the IPP/1.1 Model and Semantics specification.

428 **26) ISSUE: MAY an IPP object return additional groups?**

429 It is ok for an IPP object to return additional groups of attributes in a response (as an extension to the
430 standard)? For example, returning the "job-state" and "job-state-reasons" in a Hold-Job, Release-Job,
431 and/or Cancel-Job operation. What about newly registered groups of attributes. If so, then the client
432 MUST ignore or do something with them. (Hugo Parra)

433 ***Suggested clarification:***

434 Clarify IPP/1.1 Model and Semantics document that the client MUST ignore or do something with
435 additional attribute groups returned than are in the IPP/1.1 Model and Semantics specification.

436 **27) ISSUE: Return first or all unsupported attributes in Unsupported**
437 **Group?**

438 Section 16.3 and 16.4 of the Model and Semantics document was moved to the Implementer's Guide
439 when creating the November 1998 draft from the June 1998 draft. The following note was contained in
440 section 16.4 that was moved:

441 Note: whether the request is accepted or rejected is determined by the value of the "ipp-attribute-fidelity"
442 attribute in a subsequent step, so that all Job Template attribute supplied are examined and all
443 unsupported attributes and/or values are copied to the Unsupported Attributes response group.

444 ***Suggested clarification (same clarification as Issue 18):***

445 Clarify in the IPP/1.1 Model and Semantics document that all operation attributes and all Job Template
446 attributes MUST be returned in the Unsupported Attributes group, unless there is a specific error status,
447 such as 'client-error-document-not-supported'.

448 **28) ISSUE: What if compression is supplied but not supported?**

449 The "compression" operation attribute is an OPTIONAL attribute for a Printer object to support in a
450 create operation. However, if a client supplies the "compression" attribute, but the IPP object doesn't
451 support the attribute at all, the Printer might attempt to print data it doesn't understand, because it is
452 compressed. In order to prevent this error, the "compression" operation attribute should have been
453 REQUIRED.

454 ***Possible Alternatives (related to Issues 3 and 6):***

- 455 1. Clarify that an IPP object MUST reject a request that supplies a "compression" operation attribute, if
456 the IPP object does not support the "compression" attribute at all. As with any such error, the IPP
457 object copies the "compression" attribute to the Unsupported Attribute Group setting the value to the
458 out-of-band 'unsupported' value and returns the "client-error-attributes-or-values-not-supported"
459 status code. The IPP object MAY reject the request, even if the client supplies the 'none' value, since
460 the IPP Printer does not have a corresponding "compression-supported" attribute.
- 461 2. Add a 'client-error-compression-not-supported' error status code. Require IPP Printer's to support
462 this error code if they do not support the "compression" operation attribute.
- 463 3. Change IPP/1.1 Model and Semantics conformance requirement for the "compression" and
464 "compression-supported" attributes from OPTIONAL to REQUIRED.

465 **29) ISSUE: Should "queued-job-count" be REQUIRED?**

466 The "queued-job-count" Printer Description attribute is an OPTIONAL attribute for a Printer object to
467 support. Since some clients may want a quick way to determine the load on an IPP Printer, querying the
468 "Printer's "queued-job-count" should always be possible, but an implementation might not support it.

469 ***Suggested change:***

470 Change IPP/1.1 Model and Semantics so that the "queued-job-count" changes from OPTIONAL to
471 REQUIRED.