

1 INTERNET-DRAFT
2 [draft-ietf-ipp-implementers-guide-00981029.txt](#)

T. Hastings
Xerox Corporation
C. Manros
Xerox Corporation

~~November 16~~ ~~October 29~~, 1998

Internet Printing Protocol/1.0: Implementer's Guide

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

11 Status of this Memo

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

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

18 To learn the current status of any Internet-Draft, please check the "Iid-abstracts.txt" listing contained in the
19 Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), nic.nordu.net (Europe), munnari.oz.au (Pacific
20 Rim), ftp.ietf.org (US East Coast), or ftp.isi.edu (US West Coast).

21

22 Abstract

23 This document is one of a set of documents, which together describe all aspects of a new Internet Printing
24 Protocol (IPP). IPP is an application level protocol that can be used for distributed printing using Internet
25 tools and technologies. This document contains information that supplements the IPP Model and
26 Semantics [IPP-MOD] and the IPP Transport and Encoding [IPP-PRO] documents. It is intended to help
27 implementers understand IPP/1.0 and some of the considerations that may assist them in the design of their
28 client and/or IPP object implementations. For example, a typical order of processing requests is given,
29 including error checking. Motivation for some of the specification decisions is also included.~~The protocol
30 is heavily influenced by the printing model introduced in the Document Printing Application (DPA)
31 [ISO10175] standard. Although DPA specifies both end-user and administrative features, IPP version 1.0
32 (IPP/1.0) focuses only on end-user functionality~~

33 The full set of IPP documents includes:

- 34 Design Goals for an Internet Printing Protocol [IPP-REQ] ~~(informational)~~
- 35 Rationale for the Structure and Model and Protocol for the Internet Printing Protocol [IPP-RAT]
- 36 ~~(informational)~~
- 37 Internet Printing Protocol/1.0: Model and Semantics [IPP-MOD]
- 38 Internet Printing Protocol/1.0: Encoding and Transport [IPP-PRO]
- 39 ~~Internet Printing Protocol/1.0: Implementer's Guide [IPP-IG] (informational)~~
- 40 Mapping between LPD and IPP Protocols [IPP LPD] ~~(informational)~~

41 The ~~design goals~~-document, "Design Goals for an Internet Printing Protocol", takes a broad look at
42 distributed printing functionality, and it enumerates real-life scenarios that help to clarify the features that
43 need to be included in a printing protocol for the Internet. It identifies requirements for three types of
44 users: end users, operators, and administrators. The design goals document calls out a subset of end user
45 requirements that are satisfied in IPP/1.0. Operator and administrator requirements are out of scope for
46 version 1.0.

47 The ~~rationale~~-document, "Rationale for the Structure and Model and Protocol for the Internet Printing
48 Protocol", describes IPP from a high level view, defines a roadmap for the various documents that form the
49 suite of IPP specifications, and gives background and rationale for the IETF working group's major
50 decisions.

51 The ~~model and semantics~~-document, "Internet Printing Protocol/1.0: Model and Semantics", describes a
52 simplified model with abstract objects, their attributes, and their operations. The model introduces a Printer
53 and a Job. The Job supports multiple documents per Job. The model document also addresses how
54 security, internationalization, and directory issues are addressed.

55 The ~~protocol specification document~~, "Internet Printing Protocol/1.0: Encoding and Transport", is a formal
56 mapping of the abstract operations and attributes defined in the model document onto HTTP/1.1. ~~The~~
57 ~~protocol specification~~ It also defines the encoding rules for a new Internet media type called
58 "application/ipp".

59 ~~The implementer's guide, "Internet Printing Protocol/1.0: Implementer's Guide", gives advice to~~
60 ~~implementers of IPP clients and IPP objects related to the model and protocol documents.~~

61 The ~~LPD mapping~~-document, "Mapping between LPD and IPP Protocols", gives some advice to
62 implementers of gateways between IPP and LPD (Line Printer Daemon) implementations.

63 ~~Notice~~

64 ~~The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications,~~
65 ~~or other proprietary rights which may cover technology that may be required to practice this standard.~~
66 ~~Please address the information to the IETF Executive Director.~~

TABLE OF CONTENTS

67			
68	1	Introduction	5
69	1.1	Conformance language	5
70	1.2	Other terminology	5
71	2	Model and Semantics	5
72	2.1	Summary of Operation Attributes.....	5
73	2.2	Suggested Operation Processing Steps for IPP Objects.....	9
74	2.2.1	Suggested Operation Processing Steps for all Operations	9
75	2.2.1.1	Validate version number	9
76	2.2.1.2	Validate operation identifier	10
77	2.2.1.3	Validate the request identifier	10
78	2.2.1.4	Validate attribute group and attribute presence and order	10
79	2.2.1.5	Validate the values of the REQUIRED Operation attributes.....	17
80	2.2.1.6	Validate the values of the OPTIONAL Operation attributes	20
81	2.2.2	Suggested Additional Processing Steps for Operations that Create/Validate Jobs and Add	
82		Documents.....	22
83	2.2.2.1	Default "ipp-attribute-fidelity" if not supplied.....	22
84	2.2.2.2	Check that the Printer object is accepting jobs	22
85	2.2.2.3	Validate the values of the Job Template attributes	23
86	2.2.2.4	Check for conflicting Job Template attributes values.....	27
87	2.2.2.5	Decide whether to REJECT the request.....	27
88	2.2.2.6	For the Validate-Job operation, RETURN one of the success status codes	28
89	2.2.2.7	Create the Job object with attributes to support.....	28
90	2.2.2.8	Return one of the success status codes.....	29
91	2.2.2.9	Accept appended Document Content.....	30
92	2.2.2.10	Scheduling and Starting to Process the Job	30
93	2.2.2.11	Completing the Job	30
94	2.2.2.12	Destroying the Job after completion	30
95	2.2.2.13	Interaction with "ipp-attribute-fidelity"	30
96	2.3	Status codes returned by operation	31
97	2.3.1	Printer Operations	31
98	2.3.1.1	Print-Job	31
99	2.3.1.2	Print-URI.....	32
100	2.3.1.3	Validate-Job	33
101	2.3.1.4	Create-Job	33
102	2.3.1.5	Get-Printer-Attributes	33
103	2.3.1.6	Get-Jobs	33
104	2.3.2	Job Operations.....	34
105	2.3.2.1	Send-Document.....	34
106	2.3.2.2	Send-URI	35
107	2.3.2.3	Cancel-Job.....	35
108	2.3.2.4	Get-Job-Attributes.....	36
109	2.4	Validate-Job	36

110	2.5	Case Sensitivity in URIs	36
111	2.6	Natural Language Override (NLO).....	37
112	2.7	The "queued-job-count" Printer Description attribute	38
113	2.7.1	Why is "queued-job-count" RECOMMENDED?.....	38
114	2.7.2	Is "queued-job-count" a good measure of how busy a printer is?	38
115	2.8	Sending empty attribute groups	38
116	2.9	Returning unsupported attributes in Get-Xxxx responses	39
117	2.10	Returning job-state in Print-Job response	39
118	2.11	Multi-valued attributes	39
119	3	Encoding and Transport	40
120	3.1	General Headers	41
121	3.2	Request Headers.....	42
122	3.3	Response Headers	43
123	3.4	Entity Headers.....	43
124	3.5	Optional support for HTTP/1.0.....	44
125	3.6	HTTP/1.1 Chunking.....	44
126	4	References	45
127	4.1	Authors' Address	46
128	5	Appendix C: Full Copyright Statement.....	46
129			

130

131 1 Introduction

132 This document contains information that supplements the IPP Model and Semantics [IPP-MOD] and the
133 IPP Transport and Encoding [IPP-PRO] documents. As such this information is not part of the formal
134 specifications. Instead information is presented to help implementers understand the specification,
135 including some of the motivation for decisions taken by the committee in developing the specification.
136 Some of the implementation considerations are intended to help implementers design their client and/or IPP
137 object implementations. If there are any contradictions between this document and [IPP-MOD] or [IPP-
138 PRO], those documents take precedence over this document.

139 1.1 Conformance language

140 Usually, this document does not contain the terminology MUST, MUST NOT, MAY, NEED NOT,
141 SHOULD, SHOULD NOT, REQUIRED, and OPTIONAL. However, when those terms do appear in this
142 document, their intent is to repeat what the [IPP-MOD] and [IPP-PRO] documents require and allow, rather
143 than specifying additional conformance requirements. These terms are defined in section 13 on
144 conformance terminology in [IPP-MOD], most of which is taken from RFC 2119 [RFC2119].

145 Implementers should read section 13 in [IPP-MOD] in order to understand these capitalized words. The
146 words MUST, MUST NOT, and REQUIRED indicate what implementations are required to support in a
147 client or IPP object in order to be conformant to [IPP-MOD] and [IPP-PRO]. MAY, NEED NOT, and
148 OPTIONAL indicate what is merely allowed as an implementer option. The verbs SHOULD and SHOULD
149 NOT indicate suggested behavior, but which is not required or disallowed, respectively, in order to
150 conform to the specification.

151 1.2 Other terminology

152 The term "sender" refers to the client that sends a request or an IPP object that returns a response. The term
153 "receiver" refers to the IPP object that receives a request and to a client that receives a response.

154 2 Model and Semantics

155 This section discusses various aspects of IPP/1.0 Model and Semantics [IPP-MOD].

156 2.1 Summary of Operation Attributes

157 Legend for the following table:

158 R indicates a REQUIRED operation or attribute for an implementation to support

159 O indicates an OPTIONAL operation or attribute for an implementation to support

160

161

Table 1. Summary of operation attributes

Operation Attributes	Printer Operations						Job Operations				
	Requests					Responses	Requests				Responses
	Print-Job, Validate-Job	Print-URI (O)	Create-Job (O)	Get-Printer-Attributes	Get-Jobs	All Operations	Send-Document (O)	Send-URI (O)	Cancel-Job	Get-Job-Attributes	All Operations
Operation parameters--REQUIRED to be supplied by the sender											
operation-id	R	R	R	R	R		R	R	R	R	
status-code						R					R
request-id	R	R	R	R	R	R	R	R	R	R	R
version-number	R	R	R	R	R	R	R	R	R	R	R
Operation attributes--REQUIRED to be supplied by the sender											
attributes-charset	R	R	R	R	R	R	R	R	R	R	R
attributes-natural-language	R	R	R	R	R	R	R	R	R	R	R
document-uri		R						R			
job-id*							R	R	R	R	
job-uri*							R	R	R	R	
last-document							R	R			
printer-uri	R	R	R	R	R		R	R	R	R	
Operation attributes--RECOMMENDED to be supplied by the sender											
job-name	R	R	R								
requesting-user-name	R	R	R	R	R		R	R	R	R	

162

Operation Attributes	Printer Operations						Job Operations				
	Requests					Responses	Requests				Responses
	Print-Job, Validate-Job	Print-URI (O)	Create-Job (O)	Get-Printer-Attributes	Get-Jobs	All Operations	Send-Document (O)	Send-URI (O)	Cancel-Job	Get-Job-Attributes	All Operations
Operation attributes—OPTIONAL to be supplied by the sender											
status-message						O					O
compression	O	O					O	O			
document-format	R	R		O			R	R			
document-name	O	O					O	O			
document-natural-language	O	O					O	O			
ipp-attribute-fidelity	R	R	R								
job-impressions	O	O	O								
job-k-octets	O	O	O								
job-media-sheets	O	O	O								
limit					R						
message									O		
my-jobs					R						
requested-attributes				R	R					R	
which-jobs					R						

* "job-id" is REQUIRED only if used together with "printer-uri" to identify the target job; otherwise, "job-uri" is REQUIRED.

163

164

165 2.2 Suggested Operation Processing Steps for IPP Objects

166 This section suggests the steps and error checks that an IPP object MAY perform when processing requests
167 and returning responses. An IPP object MAY perform some or all of the error checks. However, some
168 implementations MAY choose to be more forgiving than the error checks shown here, in order to be able to
169 accept requests from non-conforming clients. Not performing all of these error checks is a so-called
170 "forgiving" implementation. On the other hand, clients that successfully submit requests to IPP objects that
171 do perform all the error checks will be more likely to be able to interoperate with other IPP object
172 implementations. Thus an implementer of an IPP object needs to decide whether to be a "forgiving" or a
173 "strict" implementation. Therefore, the error status codes returned may differ between implementations.
174 Consequentially, client SHOULD NOT expect exactly the error code processing described in this section.

175 When an IPP object receives a request, the IPP object either accepts or rejects the request. In order to
176 determine whether or not to accept or reject the request, the IPP object SHOULD execute the following
177 steps. The order of the steps may be rearranged and/or combined, including making one or multiple passes
178 over the request. ~~Therefore, the error status codes returned may differ between implementations.~~

179 A client MUST supply requests that would pass all of the error checks indicated here in order to be a
180 conforming client. Therefore, a client SHOULD supply requests that are conforming, in order to avoid
181 being rejected by some IPP object implementations.~~The next section contains the additional steps for the~~
182 ~~Print-Job, Validate-Job, Print-URI, Create-Job, Send-Document, and Send-URI operations that create jobs,~~
183 ~~adds documents, and validates jobs.~~

184 In the following, processing continues step by step until a "RETURNS the xxx status code ..." statement is
185 encountered. Error returns are indicated by the verb: "REJECTS". Since clients have difficulty getting the
186 status code before sending all of the document data in a Print-Job request, clients SHOULD use the
187 Validate-Job operation before sending large documents to be printed, in order to validate whether the IPP
188 Printer will accept the job or not.

189 It is assumed that security authentication and authorization has already taken place at a lower layer.

190 2.2.1 Suggested Operation Processing Steps for all Operations

191 This section is intended to apply to all operations. The next section contains the additional steps for the
192 Print-Job, Validate-Job, Print-URI, Create-Job, Send-Document, and Send-URI operations that create jobs,
193 adds documents, and validates jobs.

194 2.2.1.1 Validate version number

195 Every request and every response contains the "version-number" attribute. The value of this attribute is the
196 major and minor version number of the syntax and semantics that the client and IPP object is using,
197 respectively. The "version-number" attribute remains in a fixed position across all future versions so that
198 all clients and IPP object that support future versions can determine which version is being used. The IPP
199 object checks to see if the major version number supplied in the request is supported. If not, the Printer

200 object REJECTS the request and RETURNS the 'server-error-version-not-supported' status code in the
201 response. The IPP object returns in the "version-number" response attribute the major and minor version
202 for the error response. Thus the client can learn at least one major and minor version that the IPP object
203 supports. The IPP object is encouraged to return the closest version number to the one supplied by the
204 client.

205 The checking of the minor version number is implementation dependent, however if the client supplied
206 minor version is explicitly supported, the IPP object MUST respond using that identical minor version
207 number. If the requested minor version is not supported (the requested minor version is either higher or
208 lower) than a supported minor version, the IPP object SHOULD return the closest supported minor version.

209 2.2.1.2 Validate operation identifier

210 The Printer object checks to see if the "operation-id" attribute supplied by the client is supported as
211 indicated in the Printer object's "printer-operations-supported" attribute. If not, the Printer REJECTS the
212 request and returns the 'server-error-operation-not-supported' status code in the response.

213 2.2.1.3 Validate the request identifier

214 The Printer object SHOULD NOT checks to see if the "request-id" attribute supplied by the client is in
215 range. ~~If the value is not~~ between 1 and $2^{*}31 - 1$ (inclusive), but copies all 32 bits~~the Printer object~~
216 ~~REJECTS the request and returns the 'client-error-bad-request' status code in the response.~~

217 Note: The "version-number", ~~attribute,~~ "operation-id", and the "request-id" attributes-parameters are
218 ~~the same~~ fixed octet positions in the IPP/1.0 encoding. The "version-number" parameter will be the same
219 fixed octet position in all versions of the protocol. These fields are validated before proceeding with the
220 rest of the validation.

221 2.2.1.4 Validate attribute group and attribute presence and order

222 The order of the following validation steps depends on implementation.

223 2.2.1.4.1 Validate the presence and order of attribute groups

224 Client requests and IPP object responses contain attribute groups that Section 3 requires to be present and
225 in a specified order. An IPP object verifies that the attribute groups are present and in the correct order in
226 requests supplied by clients (attribute groups without an * in the following tables).

227 If an IPP object receives a request with (1) required attribute groups missing, or (2) the attributes groups are
228 out of order, or (3) the groups are repeated, the IPP object REJECTS the request and RETURNS the 'client-
229 error-bad-request' status code. For example, it is an error for the Job Template Attributes group to occur
230 before the Operation Attributes group, for the Operation Attributes group to be omitted, or for an attribute
231 group to occur more than once, except in the Get-Jobs response.

232 Since this kind of attribute group error is most likely to be an error detected by a client developer rather
233 than by a customer, the IPP object NEED NOT return an indication of which attribute group was in error in

234 either the Unsupported Attributes group or the Status Message. Also, the IPP object NEED NOT find all
235 attribute group errors before returning this error.

236 2.2.1.4.2 Ignore unknown attribute groups in the expected position

237 Future attribute groups may be added to the specification at the end of requests just before the Document
238 Content and at the end of response, except for the Get-Jobs response, where it maybe there or before the
239 first job attributes returned. If an IPP object receives an unknown attribute group in these positions, it
240 ignores the entire group, rather than returning an error, since that group may be a new group in a later
241 minor version of the protocol that can be ignored. (If the new attribute group cannot be ignored without
242 confusing the client, the major version number would have been increased in the protocol document and in
243 the request). If the unknown group occurs in a different position, the IPP object REJECTS the request and
244 RETURNS the 'client-error-bad-request' status code.

245 Clients also ignore unknown attribute groups returned in a response.

246 Note: By validating that requests are in the proper form, IPP objects force clients to use the proper form
247 which, in turn, increases the chances that customers will be able to use such clients from multiple vendors
248 with IPP objects from other vendors.

249 2.2.1.4.3 Validate the presence of a single occurrence of required Operation attributes

250 Client requests and IPP object responses contain Operation attributes that [IPP-MOD] Section 3 requires to
251 be present. Attributes within a group may be in any order, except for the ordering of target, charset, and
252 natural languages attributes. These attributes ~~must~~**MUST** be first, and ~~must~~**MUST** be supplied in the
253 following order: charset, natural language, and then target. An IPP object verifies that the attributes that
254 Section 4 requires to be supplied by the client have been supplied in the request (attributes without an * in
255 the following tables). An asterisk (*) indicates groups and Operation attributes that the client may omit in a
256 request or an IPP object may omit in a response.

257 If an IPP object receives a request with required attributes missing or repeated from a group, the IPP object
258 REJECTS the request and RETURNS the 'client-error-bad-request' status code. For example, it is an error
259 for the "attributes-charset" or "attributes-natural-language" attribute to be omitted in any operation request,
260 or for an Operation attribute to be supplied in a Job Template group or a Job Template attribute to be
261 supplied in an Operation Attribute group in a create request. It is also an error to supply the "attributes-
262 charset" attribute twice.

263 Since these kinds of attribute errors are most likely to be detected by a client developer rather than by a
264 customer, the IPP object NEED NOT return an indication of which attribute was in error in either the
265 Unsupported Attributes group or the Status Message. Also, the IPP object NEED NOT find all attribute
266 errors before returning this error.

267 The following tables list all the attributes for all the operations by attribute group in each request and each
268 response. The order of the groups is the order that the client supplies the groups as specified in [IPP-MOD]
269 Section 3. The order of the attributes within a group is arbitrary, except as noted for some of the special
270 operation attributes (charset, natural language, and target). The tables below use the following notation:

271 R indicates a REQUIRED attribute that an IPP object MUST support
272 O indicates an OPTIONAL attribute that an IPP object NEED NOT support
273 * indicates that a client MAY omit the attribute in a request and that an IPP object MAY omit
274 the attribute in a response. The absence of an * means that a client MUST supply the
275 attribute in a request and an IPP object MUST supply the attribute in a response.
276

277 Operation Requests

278 The tables below show the attributes in their proper attribute groups for operation requests:

279 Note: All operation requests contain "version-number", "operation-id",
280 and "request-id" parameters.

281

282 Print-Job Request:
283 Group 1: Operation Attributes (R)
284 attributes-charset (R)
285 attributes-natural-language (R)
286 printer-uri (R)
287 requesting-user-name (R*)
288 job-name (R*)
289 ipp-attribute-fidelity (R*)
290 document-name (R*)
291 document-format (R*)
292 document-natural-language (O*)
293 compression (O*)
294 job-k-octets (O*)
295 job-impressions (O*)
296 job-media-sheets (O*)
297 Group 2: Job Template Attributes (R*)
298 <Job Template attributes> (O*)
299 (see [IPP-MOD] Section 4.2)
300 Group 3: Document Content (R)
301 <document content>
302
303 Validate-Job Request:
304 Group 1: Operation Attributes (R)
305 attributes-charset (R)
306 attributes-natural-language (R)
307 printer-uri (R)
308 requesting-user-name (R*)
309 job-name (R*)
310 ipp-attribute-fidelity (R*)
311 document-name (R*)
312 document-format (R*)
313 document-natural-language (O*)
314 compression (O*)
315 job-k-octets (O*)
316 job-impressions (O*)
317 job-media-sheets (O*)
318 Group 2: Job Template Attributes (R*)
319 <Job Template attributes> (O*)
320 (see [IPP-MOD] Section 4.2)
321
322 Create-Job Request:
323 Group 1: Operation Attributes (R)
324 attributes-charset (R)
325 attributes-natural-language (R)
326 printer-uri (R)
327 requesting-user-name (R*)
328 job-name (R*)

329 ipp-attribute-fidelity (R*)
330 job-k-octets (O*)
331 job-impressions (O*)
332 job-media-sheets (O*)
333 Group 2: Job Template Attributes (R*)
334 <Job Template attributes> (O*) (see
335 (see [IPP-MOD] Section 4.2)
336
337 Print-URI Request:
338 Group 1: Operation Attributes (R)
339 attributes-charset (R)
340 attributes-natural-language (R)
341 printer-uri (R)
342 document-uri (R)
343 requesting-user-name (R*)
344 job-name (R*)
345 ipp-attribute-fidelity (R*)
346 document-name (R*)
347 document-format (R*)
348 document-natural-language (O*)
349 compression (O*)
350 job-k-octets (O*)
351 job-impressions (O*)
352 job-media-sheets (O*)
353 Group 2: Job Template Attributes (R*)
354 <Job Template attributes> (O*) (see
355 (see [IPP-MOD] Section 4.2)
356
357 Send-Document Request:
358 Group 1: Operation Attributes (R)
359 attributes-charset (R)
360 attributes-natural-language (R)
361 (printer-uri & job-id) | job-uri (R)
362 last-document (R)
363 requesting-user-name (R*)
364 document-name (R*)
365 document-format (R*)
366 document-natural-language (O*)
367 compression (O*)
368 Group 2: Document Content (R*)
369 <document content>
370
371 Send-URI Request:
372 Group 1: Operation Attributes (R)
373 attributes-charset (R)
374 attributes-natural-language (R)
375 (printer-uri & job-id) | job-uri (R)

376 last-document (R)
377 document-uri (R)
378 requesting-user-name (R*)
379 document-name (R*)
380 document-format (R*)
381 document-natural-language (O*)
382 compression (O*)
383
384 Cancel-Job Request:
385 Group 1: Operation Attributes (R)
386 attributes-charset (R)
387 attributes-natural-language (R)
388 (printer-uri & job-id) | job-uri (R)
389 requesting-user-name (R*)
390 message (O*)
391
392 Get-Printer-Attributes Request:
393 Group 1: Operation Attributes (R)
394 attributes-charset (R)
395 attributes-natural-language (R)
396 printer-uri (R)
397 requesting-user-name (R*)
398 requested-attributes (R*)
399 document-format (R*)
400
401 Get-Job-Attributes Request:
402 Group 1: Operation Attributes (R)
403 attributes-charset (R)
404 attributes-natural-language (R)
405 (printer-uri & job-id) | job-uri (R)
406 requesting-user-name (R*)
407 requested-attributes (R*)
408
409 Get-Jobs Request:
410 Group 1: Operation Attributes (R)
411 attributes-charset (R)
412 attributes-natural-language (R)
413 printer-uri (R)
414 requesting-user-name (R*)
415 limit (R*)
416 requested-attributes (R*)
417 which-jobs (R*)
418 my-jobs (R*)
419

420 Operation Responses

421 The tables below show the response attributes in their proper attribute groups for responses.

422 Note: All operation responses contain "version-number", "status-code",
423 and "request-id" parameters.

424

425 Print-Job Response:

426 Print-URI Response:

427 Create-Job Response:

428 Send-Document Response:

429 Send-URI Response:

430 Group 1: Operation Attributes (R)

431 attributes-charset (R)

432 attributes-natural-language (R)

433 status-message (O*)

434 Group 2: Unsupported Attributes (R*) (see Note 3)

435 <unsupported attributes> (R*)

436 Group 3: Job Object Attributes (R*) (see Note 2)

437 job-uri (R)

438 job-id (R)

439 job-state (R)

440 job-state-reasons (O*)

441 job-state-message (O*)

442 number-of-intervening-jobs (O*)

443

444 Validate-Job Response:

445 Cancel-Job Response:

446 Group 1: Operation Attributes (R)

447 attributes-charset (R)

448 attributes-natural-language (R)

449 status-message (O*)

450 Group 2: Unsupported Attributes (R*) (see Note 3)

451 <unsupported attributes> (R*)

452

453 Note 2 - the Job Object Attributes and Printer Object Attributes are returned only if the IPP object returns
454 one of the success status codes.

455

456 Note 3 - the Unsupported Attributes Group is present only if the client included some Operation and/or Job
457 Template attributes or values that the Printer doesn't support whether a success or an error return.

458
459 Get-Printer-Attributes Response:
460 Group 1: Operation Attributes (R)
461 attributes-charset (R)
462 attributes-natural-language (R)
463 status-message (O*)
464 Group 2: Unsupported Attributes (R*) (see Note 4)
465 <unsupported attributes> (R*)
466 Group 3: Printer Object Attributes(R*) (see Note 2)
467 <requested attributes> (R*)
468

469 Note 4 - the Unsupported Attributes Group is present only if the client included some Operation attributes
470 that the Printer doesn't support whether a success or an error return.

471
472 Get-Job-Attributes Response:
473 Group 1: Operation Attributes (R)
474 attributes-charset (R)
475 attributes-natural-language (R)
476 status-message (O*)
477 Group 2: Unsupported Attributes (R*) (see Note 4)
478 <unsupported attributes> (R*)
479 Group 3: Job Object Attributes(R*) (see Note 2)
480 <requested attributes> (R*)
481

482 Get-Jobs Response:
483 Group 1: Operation Attributes (R)
484 attributes-charset (R)
485 attributes-natural-language (R)
486 status-message (O*)
487 Group 2: Unsupported Attributes (R*) (see Note 4)
488 <unsupported attributes> (R*)
489 Group 3: Job Object Attributes(R*) (see Note 2, 5)
490 <requested attributes> (R*)
491

492 Note 5: for the Get-Jobs operation the response contains a separate Job Object Attributes group 3 to N
493 containing requested-attributes for each job object in the response.

494 2.2.1.5 Validate the values of the REQUIRED Operation attributes

495 An IPP object validates the values supplied by the client of the REQUIRED Operation attribute that the IPP
496 object MUST support. The next section specifies the validation of the values of the OPTIONAL Operation
497 attributes that IPP objects MAY support.

498 The IPP object performs the following syntactic validation checks of each Operation attribute value:

- 499 a) that the length of each Operation attribute value is correct for the attribute syntax tag supplied
500 by the client according to [\[IPP-MOD\]](#) Section 4.1,
- 501 b) that the attribute syntax tag is correct for that Operation attribute according to [\[IPP-MOD\]](#)
502 Section 3,
- 503 c) that the value is in the range specified for that Operation attribute according to [\[IPP-MOD\]](#)
504 Section 3,
- 505 d) that multiple values are supplied by the client only for operation attributes that are multi-valued,
506 i.e., that are 1setOf X according to [\[IPP-MOD\]](#) Section 3.

507 If any of these checks fail, the IPP object REJECTS the request and RETURNS the 'client-error-bad-
508 request' or the 'client-error-request-value-too-long' status code. Since such an error is most likely to be an
509 error detected by a client developer, rather than by an end-user, the IPP object NEED NOT return an
510 indication of which attribute had the error in either the Unsupported Attributes Group or the Status
511 Message. The description for each of these syntactic checks is explicitly expressed in the first IF statement
512 in the following table.

513 In addition, the IPP object checks each Operation attribute value against some Printer object attribute or
514 some hard-coded value if there is no "xxx-supported" Printer object attribute defined. If its value is not
515 among those supported or is not in the range supported, then the IPP object REJECTS the request and
516 RETURNS the error status code indicated in the table by the second IF statement. If the value of the
517 Printer object's "xxx-supported" attribute is 'no-value' (because the system administrator hasn't configured a
518 value), the check always fails.

519

520 -----
attributes-charset (charset)

521 IF NOT any single non-empty 'charset' value less than or equal to 63 octets, REJECT/RETURN 'client-
522 error-request-value-too-long'.

523 IF NOT in the Printer object's "charset-supported" attribute, REJECT/RETURN "client-error-charset-
524 not-supported".
525

526 attributes-natural-language(naturalLanguage)

527 IF NOT any single non-empty 'naturalLanguage' value less than or equal to 63 octets,
528 REJECT/RETURN 'client-error-request-value-too-long'.

529 ACCEPT the request even if not a member of the set in the Printer object's "generated-natural-
530 language-supported" attribute.
531

532 requesting-user-name

533 IF NOT any single 'name' value less than or equal to 255 octets, REJECT/RETURN 'client-error-
534 request-value-too-long'.

535 IF the IPP object can obtain a better authenticated name, use it instead.
536

537 job-name(name)
538 IF NOT any single 'name' value less than or equal to 255 octets, REJECT/RETURN 'client-error-
539 request-value-too-long'.
540 IF NOT supplied by the client, the Printer object creates a name from the document-name or document-
541 uri.
542

543 document-name (name)
544 IF NOT any single 'name' value less than or equal to 255 octets, REJECT/RETURN 'client-error-
545 request-value-too-long'.
546

547 ipp-attribute-fidelity (boolean)
548 IF NOT either a single 'true' or 'false' 'boolean' value equal to 1 octet, REJECT/RETURN 'client-error-
549 bad-request'.
550 IF NOT supplied by the client, the IPP object assumes the value 'false'.
551

552 document-format (mimeType)
553 IF NOT any single non-empty 'mimeType' value less than or equal to 255 octets,
554 REJECT/RETURN 'client-error-request-value-too-long'.
555 IF NOT in the Printer object's "document-format-supported" attribute, REJECT/RETURN 'client-error-
556 document-format-not-supported'.
557 IF NOT supplied by the client, the IPP object assumes the value of the Printer object's "document-
558 format-default" attribute.
559

560 document-uri (uri)
561 IF NOT any single non-empty 'uri' value less than or equal to 1023 octets, REJECT/RETURN 'client-
562 error-request-value-too-long'.
563 IF the URI syntax is not valid, REJECT/RETURN 'client-error-bad-request'.
564 IF scheme is NOT in the Printer object's "reference-uri-schemes-supported" attribute,
565 REJECT/RETURN 'client-error-uri-scheme-not-supported'.
566

567 last-document (boolean)
568 IF NOT either a single 'true' or 'false' 'boolean' value equal to 1 octet, REJECT/RETURN 'client-error-
569 bad-request'.
570

571 job-id (integer(1:MAX))
572 IF NOT any single 'integer' value equal to 4 octets AND in the range 1 to MAX, REJECT/RETURN
573 'client-error-bad-request'.
574 IF NOT a job-id of an existing Job object, REJECT/RETURN 'client-error-not-found' or 'client-error-
575 gone' status code, if keep track of recently deleted jobs.
576

577 requested-attributes (1setOf keyword)

578 IF NOT any number of 'keyword' values less than or equal to 255 octets, REJECT/RETURN 'client-
579 error-request-value-too-long'.

580 Ignore unsupported values which are the keyword names of unsupported attributes. Don't bother to
581 copy such requested (unsupported) attributes to the Unsupported Attribute response group since the
582 response will not return them.
583

584 which-jobs (type2 keyword)

585 IF NOT a single 'keyword' value less than or equal to 255 octets, REJECT/RETURN 'client-error-
586 request-value-too-long'.

587 IF NEITHER 'completed' NOR 'not-completed', copy the attribute and the unsupported value to the
588 Unsupported Attributes response group and REJECT/RETURN 'client-error-attributes-or-values-
589 not-supported'.

590 Note: a Printer still supports the 'completed' value even if it keeps no completed/canceled/aborted jobs:
591 by returning no jobs when so queried.

592 IF NOT supplied by the client, the IPP object assumes the 'not-completed' value.
593

594 my-jobs (boolean)

595 IF NOT either a single 'true' or 'false' 'boolean' value equal to 1 octet, REJECT/RETURN 'client-error-
596 bad-request'.

597 IF NOT supplied by the client, the IPP object assumes the 'false' value.
598

599 limit (integer(1:MAX))

600 IF NOT any single 'integer' value equal to 4 octets AND in the range 1 to MAX, REJECT/RETURN
601 'client-error-bad-request'.

602 IF NOT supplied by the client, the IPP object returns all jobs, no matter how many.
603

604 -----

605

606 2.2.1.6 Validate the values of the OPTIONAL Operation attributes

607 OPTIONAL Operation attributes are those that an IPP object MAY or MAY NOT support. An IPP object
608 validates the values of the OPTIONAL attributes supplied by the client. The IPP object performs the same
609 syntactic validation checks for each OPTIONAL attribute value as in Section 2.2.1.5. As in Section
610 2.2.1.5, if any fail, the IPP object REJECTS the request and RETURNS the 'client-error-bad-request' or the
611 'client-error-request-value-too-long' status code.

612 In addition, the IPP object checks each Operation attribute value against some Printer attribute or some
613 hard-coded value if there is no "xxx-supported" Printer attribute defined. If its value is not among those
614 supported or is not in the range supported, then the IPP object REJECTS the request and RETURNS the
615 error status code indicated in the table. If the value of the Printer object's "xxx-supported" attribute is 'no-
616 value' (because the system administrator hasn't configured a value), the check always fails.

617 If the IPP object doesn't recognize/support an attribute, the IPP object treats the attribute as an unknown or
618 unsupported attribute (see the last row in the table below).

619 -----

620 document-natural-language (naturalLanguage)

621 IF NOT any single non-empty 'naturalLanguage' value less than or equal to 63 octets,

622 REJECT/RETURN 'client-error-request-value-too-long'.

623 IF NOT a value that the Printer object supports in document formats, (no corresponding "xxx-

624 supported" Printer attribute), REJECT/RETURN 'client-error-natural-language-not-supported'.

625

626 compression (type3 keyword)

627 IF NOT any single 'keyword' values less than or equal to 255 octets, REJECT/RETURN 'client-error-
628 request-value-too-long'.

629 IF NOT in the Printer object's "compression-supported" attribute, copy the attribute and the

630 unsupported value to the Unsupported Attributes response group and REJECT/RETURN 'client-

631 error-attributes-or-values-not-supported'.

632

633 job-k-octets (integer(0:MAX))

634 IF NOT any single 'integer' value equal to 4 octets,

635 REJECT/RETURN 'client-error-bad-request'.

636 IF NOT in the range of the Printer object's "job-k-octets-supported" attribute, copy the attribute and the

637 unsupported value to the Unsupported Attributes response group and REJECT/RETURN 'client-

638 error-attributes-or-values-not-supported'.

639

640 job-impressions (integer(0:MAX))

641 IF NOT any single 'integer' value equal to 4 octets,

642 REJECT/RETURN 'client-error-bad-request'.

643 IF NOT in the range of the Printer object's "job-impressions-supported" attribute, copy the attribute and

644 the unsupported value to the Unsupported Attributes response group and REJECT/RETURN 'client-

645 error-attributes-or-values-not-supported'.

646

647 job-media-sheets (integer(0:MAX))

648 IF NOT any single 'integer' value equal to 4 octets,

649 REJECT/RETURN 'client-error-bad-request'.

650 IF NOT in the range of the Printer object's "job-media-supported" attribute, copy the attribute and the

651 unsupported value to the Unsupported Attributes response group and REJECT/RETURN 'client-

652 error-attributes-or-values-not-supported'.

653

654 message (text(127))

655 IF NOT any single 'text' value less than or equal to 127 octets,

656 REJECT/RETURN 'client-error-request-value-too-long'.
657

658 unknown or unsupported attribute

659 IF the attribute syntax supplied by the client is supported but the length is not legal for that attribute
660 syntax, REJECT/RETURN 'client-error-request-value-too-long'.

661 ELSE copy the attribute and value to the Unsupported Attributes response group and change the
662 attribute value to the "out-of-band" 'unsupported' value, but otherwise ignore the attribute.
663

664 Note: Future Operation attributes may be added to the protocol specification that may occur anywhere
665 in the specified group. When the operation is otherwise successful, the IPP object returns the
666 'successful-ok-ignored-or-substituted-attributes' status code. Ignoring unsupported Operation attributes
667 in all operations is analogous to the handling of unsupported Job Template attributes in the create and
668 Validate-Job operations when the client supplies the "ipp-attribute-fidelity" Operation attribute with the
669 'false' value. This last rule is so that we can add OPTIONAL Operation attributes to future versions of
670 IPP so that older clients can inter-work with new IPP objects and newer clients can inter-work with
671 older IPP objects. (If the new attribute cannot be ignored without performing unexpectedly, the major
672 version number would have been increased in the protocol document and in the request). This rule for
673 Operation attributes is independent of the value of the "ipp-attribute-fidelity" attribute. For example, if
674 an IPP object doesn't support the OPTIONAL "job-k-octets" attribute', the IPP object treats "job-k-
675 octets" as an unknown attribute and only checks the length for the 'integer' attribute syntax supplied by
676 the client. If it is not four octets, the IPP object REJECTS the request and RETURNS the 'client-error-
677 bad-request' status code, else the IPP object copies the attribute to the Unsupported Attribute response
678 group, setting the value to the "out-of-band" 'unsupported' value, but otherwise ignores the attribute.

679 2.2.2 Suggested Additional Processing Steps for Operations that Create/Validate Jobs and Add
680 Documents

681 This section in combination with the previous section recommends the processing steps for the Print-Job,
682 Validate-Job, Print-URI, Create-Job, Send-Document, and Send-URI operations that IPP objects SHOULD
683 use. These are the operations that create jobs, validate a Print-Job request, and add documents to a job.

684 2.2.2.1 Default "ipp-attribute-fidelity" if not supplied

685 The Printer object checks to see if the client supplied an "ipp-attribute-fidelity" Operation attribute. If the
686 attribute is not supplied by the client, the IPP object assumes that the value is 'false'.

687 2.2.2.2 Check that the Printer object is accepting jobs

688 If the value of the Printer object's "printer-is-accepting-jobs" is 'false', the Printer object REJECTS the
689 request and RETURNS the 'server-error-not-accepting-jobs' status code.

690 2.2.2.3 Validate the values of the Job Template attributes

691 An IPP object validates the values of all Job Template attribute supplied by the client. The IPP object
692 performs the analogous syntactic validation checks of each Job Template attribute value that it performs for
693 Operation attributes (see Section 2.2.1.5.):

694 a) that the length of each value is correct for the attribute syntax tag supplied by the client
695 according to [IPP-MOD] Section 4.1.

696 b) that the attribute syntax tag is correct for that attribute according to [IPP-MOD] Sections 4.2 to
697 4.4.

698 c) that multiple values are supplied only for multi-valued attributes, i.e., that are 1setOf X
699 according to [IPP-MOD] Sections 4.2 to 4.4.

700 As in Section 2.2.1.5, if any of these syntactic checks fail, the IPP object REJECTS the request and
701 RETURNS the 'client-error-bad-request' or 'client-error-request-value-too-long' status code as appropriate,
702 independent of the value of the "ipp-attribute-fidelity". Since such an error is most likely to be an error
703 detected by a client developer, rather than by an end-user, the IPP object NEED NOT return an indication
704 of which attribute had the error in either the Unsupported Attributes Group or the Status Message. The
705 description for each of these syntactic checks is explicitly expressed in the first IF statement in the
706 following table.

707 In addition, the IPP object loops through all the client-supplied Job Template attributes, checking to see if
708 the supplied attribute value(s) are supported or in the range supported, i.e., the value of the "xxx" attribute
709 in the request is (1) a member of the set of values or is in the range of values of the Printer's "xxx-
710 supported" attribute. If the value of the Printer object's "xxx-supported" attribute is 'no-value' (because the
711 system administrator hasn't configured a value), the check always fails. If the check fails, the IPP object
712 copies the attribute to the Unsupported Attributes response group with its unsupported value. If the
713 attribute contains more than one value, each value is checked and each unsupported value is separately
714 copied, while supported values are not copied. If an IPP object doesn't recognize/support a Job Template
715 attribute, i.e., there is no corresponding Printer object "xxx-supported" attribute, the IPP object treats the
716 attribute as an unknown or unsupported attribute (see the last row in the table below).

717 If some Job Template attributes are supported for some document formats and not for others or the values
718 are different for different document formats, the IPP object SHOULD take that into account in this
719 validation using the value of the "document-format" supplied by the client (or defaulted to the value of the
720 Printer's "document-format-default" attribute, if not supplied by the client). For example, if "number-up" is
721 supported for the 'text/plain' document format, but not for the 'application/postscript' document format, the
722 check SHOULD (though it NEED NOT) depend on the value of the "document-format" operation attribute.
723 See "document-format" in [IPP-MOD] section 3.2.1.1 and 3.2.5.1.

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

727 -----

- 728 job-priority (integer(1:100))
- 729 IF NOT **any** single 'integer' value with a length equal to 4 octets, REJECT/RETURN 'client-error-bad-
730 request'.
- 731 IF NOT supplied by the client, use the value of the Printer object's "job-priority-default" attribute at job
732 submission time.
- 733 IF NOT in the range 1 to 100, inclusive, copy the attribute and the unsupported value to the
734 Unsupported Attributes response group.
- 735 Map the value to the nearest supported value in the range 1:100 as specified by the number of discrete
736 values indicated by the value of the Printer's "job-priority-supported" attribute. See the formula in
737 [IPP-MOD] Section 4.2.1.
738
- 739 job-hold-until (type3 keyword | name)
- 740 IF NOT **any** single 'keyword' or 'name' value with a length less than or equal to 255 octets,
741 REJECT/RETURN 'client-error-request-value-too-long'.
- 742 IF NOT supplied by the client, use the value of the Printer object's "job-hold-until" attribute at job
743 submission time.
- 744 IF NOT in the Printer object's "job-hold-until-supported" attribute, copy the attribute and the
745 unsupported value to the Unsupported Attributes response group.
746
- 747 job-sheets (type3 keyword | name)
- 748 IF NOT **any** single 'keyword' or 'name' value with a length less than or equal to 255 octets,
749 REJECT/RETURN 'client-error-request-value-too-long'.
- 750 IF NOT in the Printer object's "job-sheets-supported" attribute, copy the attribute and the unsupported
751 value to the Unsupported Attributes response group.
752
- 753 multiple-document-handling (type2 keyword)
- 754 IF NOT **any** single 'keyword' value with a length less than or equal to 255 octets, REJECT/RETURN
755 'client-error-request-value-too-long'.
- 756 IF NOT in the Printer object's "multiple-document-handling-supported" attribute, copy the attribute and
757 the unsupported value to the Unsupported Attributes response group.
758
- 759 copies (integer(1:MAX))
- 760 IF NOT **any** single 'integer' value with a length equal to 4 octets,
761 REJECT/RETURN 'client-error-bad-request'.
- 762 IF NOT in range of the Printer object's "copies-supported" attribute
763 copy the attribute and the unsupported value to the Unsupported Attributes response group.
764
- 765 finishings (1setOf type2 enum)
- 766 IF NOT **any** 'enum' value(s) each with a length equal to 4 octets, REJECT/RETURN 'client-error-bad-
767 request'.

768 IF NOT in the Printer object's "finishings-supported" attribute, copy the attribute and the unsupported
769 value(s), but not any supported values, to the Unsupported Attributes response group.
770

771 page-ranges (1setOf rangeOfInteger(1:MAX))

772 IF NOT **any** 'rangeOfInteger' value(s) each with a length equal to 8 octets, REJECT/RETURN 'client-
773 error-bad-request'.
774 IF first value is greater than second value in any range, the ranges are not in ascending order, or ranges
775 overlap, REJECT/RETURN 'client-error-bad-request'.
776 IF the value of the Printer object's "page-ranges-supported" attribute is 'false', copy the attribute to the
777 Unsupported Attributes response group and set the value to the "out-of-band" 'unsupported' value.
778

779 sides (type2 keyword)

780 IF NOT **any** single 'keyword' value with a length less than or equal to 255 octets, REJECT/RETURN
781 'client-error-request-value-too-long'.
782 IF NOT in the Printer object's "sides-supported" attribute, copy the attribute and the unsupported value
783 to the Unsupported Attributes response group.
784

785 number-up (integer(1:MAX))

786 IF NOT **any** single 'integer' value with a length equal to 4 octets,
787 REJECT/RETURN 'client-error-bad-request'.
788 IF NOT a value or in the range of one of the values of the Printer object's "number-up-supported"
789 attribute, copy the attribute and value to the Unsupported Attribute response group.
790

791 orientation-requested (type2 enum)

792 IF NOT **any** single 'enum' value with a length equal to 4 octets,
793 REJECT/RETURN 'client-error-bad-request'.
794 IF NOT in the Printer object's "orientation-requested-supported" attribute, copy the attribute and the
795 unsupported value to the Unsupported Attributes response group.
796

797 media (type3 keyword | name)

798 IF NOT **any** single 'keyword' or 'name' value with a length less than or equal to 255 octets,
799 REJECT/RETURN 'client-error-request-value-too-long'.
800 IF NOT in the Printer object's "media-supported" attribute, copy the attribute and the unsupported value
801 to the Unsupported Attributes response group.
802

803 printer-resolution (resolution)

804 IF NOT **any** single 'resolution' value with a length equal to 9 octets,
805 REJECT/RETURN 'client-error-bad-request'.
806 IF NOT in the Printer object's "multiple-document-handling-supported" attribute, copy the attribute and
807 the unsupported value to the Unsupported Attributes response group.
808

809 print-quality (type2 enum)

810 IF NOT ~~any~~ single 'enum' value with a length equal to 4 octets,

811 REJECT/RETURN 'client-error-bad-request'.

812 IF NOT in the Printer object's "print-quality-supported" attribute, copy the attribute and the
813 unsupported value to the Unsupported Attributes response group.

814

815 unknown or unsupported attribute (i.e., there is no corresponding Printer object "xxx-supported" attribute)

816 IF the attribute syntax supplied by the client is supported but the length is not legal for that attribute
817 syntax,

818 REJECT/RETURN 'client-error-bad-request' if the length of the attribute syntax is fixed or 'client-error-
819 request-value-too-long' if the length of the attribute syntax is variable.

820 ELSE copy the attribute and value to the Unsupported Attributes response group and change the
821 attribute value to the "out-of-band" 'unsupported' value. Any remaining Job Template Attributes are
822 either unknown or unsupported Job Template attributes and are validated algorithmically according
823 to their attribute syntax for proper length (see below).

824 -----

825

826 If the attribute syntax is supported AND the length check fails, the IPP object REJECTS the request and
827 RETURNS the 'client-error-bad-request' if the length of the attribute syntax is fixed or the 'client-error-
828 request-value-too-long' status code if the length of the attribute syntax is variable; ~~else~~ Otherwise, the IPP
829 object copies the unsupported Job Template attribute to the Unsupported Attributes response group and
830 changes the attribute value to the "out-of-band" 'unsupported' value. The following table shows the length
831 checks for all attribute syntaxes. In the following table: "<=" means less than or equal, "=" means equal
832 to:

833	Name	Octet length check for read-write attributes
834	-----	-----
835	'textWithLanguage	<= 1023 AND 'naturalLanguage' <= 63
836	'textWithoutLanguage'	<= 1023
837	'nameWithLanguage'	<= 255 AND 'naturalLanguage' <= 63
838	'nameWithoutLanguage'	<= 255
839	'keyword'	<= 255
840	'enum'	= 4
841	'uri'	<= 1023
842	'uriScheme'	<= 63
843	'charset'	<= 63
844	'naturalLanguage'	<= 63
845	'mimeType'	<= 255
846	'octetString'	<= 1023
847	'boolean'	= 1
848	'integer'	= 4
849	'rangeOfInteger'	= 8
850	'dateTime'	= 11
851	'resolution'	= 9
852	'1setOf X'	
853		

854 2.2.2.4 Check for conflicting Job Template attributes values

855 Once all the Operation and Job Template attributes have been checked individually, the Printer object
 856 SHOULD check for any conflicting values among all the supported values supplied by the client. For
 857 example, a Printer object might be able to staple and to print on transparencies, however due to physical
 858 stapling constraints, the Printer object might not be able to staple transparencies. The IPP object copies the
 859 supported attributes and their conflicting attribute values to the Unsupported Attributes response group.
 860 The Printer object only copies over those attributes that the Printer object either ignores or substitutes in
 861 order to resolve the conflict, and it returns the original values which were supplied by the client. For
 862 example suppose the client supplies "finishings" equals 'staple' and "media" equals 'transparency', but the
 863 Printer object does not support stapling transparencies. If the Printer chooses to ignore the stapling request
 864 in order to resolve the conflict, the Printer objects returns "finishings" equal to 'staple' in the Unsupported
 865 Attributes response group. If any attributes are multi-valued, only the conflicting values of the attributes
 866 are copied.

867 Note: The decisions made to resolve the conflict (if there is a choice) is implementation dependent.

868 2.2.2.5 Decide whether to REJECT the request

869 If there were any unsupported Job Template attributes or unsupported/conflicting Job Template attribute
 870 values and the client supplied the "ipp-attribute-fidelity" attribute with the 'true' value, the Printer object
 871 REJECTS the request and return the status code:

- 872 (1) 'client-error-conflicting-attributes' status code, if there were any conflicts between attributes
- 873 supplied by the client.

874 (2) 'client-error-attributes-or-values-not-supported' status code, otherwise.

875

876 Note: Unsupported Operation attributes or values that are returned do not affect the status returned in this
877 step. If the unsupported Operation attribute was a serious error, the above already rejected the request in a
878 previous step. If control gets to this step with unsupported Operation attributes being returned, they are not
879 serious errors.

880 2.2.2.6 For the Validate-Job operation, RETURN one of the success status codes

881 If the requested operation is the Validate-Job operation, the Printer object returns:

882 (1) the "successful-ok" status code, if there are no unsupported or conflicting Job Template attributes or
883 values.

884 (2) the "successful-ok-conflicting-attributes, if there are any conflicting Job Template attribute or
885 values.

886 (3) the "successful-ok-ignored-or-substituted-attributes, if there are only unsupported Job Template
887 attributes or values.

888

889 Note: Unsupported Operation attributes or values that are returned do not affect the status returned in this
890 step. If the unsupported Operation attribute was a serious error, the above already rejected the request in a
891 previous step. If control gets to this step with unsupported Operation attributes being returned, they are not
892 serious errors.

893 2.2.2.7 Create the Job object with attributes to support

894 If "ipp-attribute-fidelity" is set to 'false' (or it was not supplied by the client), the Printer object:

895 (1) creates a Job object, assigns a unique value to the job's "job-uri" and "job-id" attributes, and
896 initializes all of the job's other supported Job Description attributes.

897 (2) removes all unsupported attributes from the Job object.

898 (3) for each unsupported value, removes either the unsupported value or substitutes the unsupported
899 attribute value with some supported value. If an attribute has no values after removing unsupported
900 values from it, the attribute is removed from the Job object (so that the normal default behavior at
901 job processing time will take place for that attribute).

902 (4) for each conflicting value, removes either the conflicting value or substitutes the conflicting
903 attribute value with some other supported value. If an attribute has no values after removing
904 conflicting values from it, the attribute is removed from the Job object (so that the normal default
905 behavior at job processing time will take place for that attribute).

906

907 If there were no attributes or values flagged as unsupported, or the value of "ipp-attribute-fidelity" was
908 'false', the Printer object is able to accept the create request and create a new Job object. If the "ipp-
909 attribute-fidelity" attribute is set to 'true', the Job Template attributes that populate the new Job object are
910 necessarily all the Job Template attributes supplied in the create request. If the "ipp-attribute-fidelity"
911 attribute is set to 'false', the Job Template attributes that populate the new Job object are all the client
912 supplied Job Template attributes that are supported or that have value substitution. Thus, some of the

913 requested Job Template attributes may not appear in the Job object because the Printer object did not
914 support those attributes. The attributes that populate the Job object are persistently stored with the Job
915 object for that Job. A Get-Job-Attributes operation on that Job object will return only those attributes that
916 are persistently stored with the Job object.

917 Note: All Job Template attributes that are persistently stored with the Job object are intended to be
918 "override values"; that is, they that take precedence over whatever other embedded instructions might be in
919 the document data itself. However, it is not possible for all Printer objects to realize the semantics of
920 "override". End users may query the Printer's "pdl-override-supported" attribute to determine if the Printer
921 either attempts or does not attempt to override document data instructions with IPP attributes.

922 There are some cases, where a Printer supports a Job Template attribute and has an associated default value
923 set for that attribute. In the case where a client does not supply the corresponding attribute, the Printer does
924 not use its default values to populate Job attributes when creating the new Job object; only Job Template
925 attributes actually in the create request are used to populate the Job object. The Printer's default values are
926 only used later at Job processing time if no other IPP attribute or instruction embedded in the document
927 data is present.

928 Note: If the default values associated with Job Template attributes that the client did not supply were to be
929 used to populate the Job object, then these values would become "override values" rather than defaults. If
930 the Printer supports the 'attempted' value of the "pdl-override-supported" attribute, then these override
931 values could replace values specified within the document data. This is not the intent of the default value
932 mechanism. A default value for an attribute is used only if the create request did not specify that attribute
933 (or it was ignored when allowed by "ipp-attribute-fidelity" being 'false') and no value was provided within
934 the content of the document data.

935 If the client does not supply a value for some Job Template attribute, and the Printer does not support that
936 attribute, as far as IPP is concerned, the result of processing that Job (with respect to the missing attribute)
937 is undefined.

938 2.2.2.8 Return one of the success status codes

939 Once the Job object has been created, the Printer object accepts the request and returns to the client:

- 940 (1) the 'successful-ok' status code, if there are no unsupported or conflicting Job Template attributes or
941 values.
- 942 (2) the 'successful-ok-conflicting-attributes' status code, if there are any conflicting Job Template
943 attribute or values.
- 944 (3) the 'successful-ok-ignored-or-substituted-attributes' status code, if there are only unsupported Job
945 Template attributes or values.

946
947 Note: Unsupported Operation attributes or values that are returned do not affect the status returned in this
948 step. If the unsupported Operation attribute was a serious error, the above already rejected the request in a
949 previous step. If control gets to this step with unsupported Operation attributes being returned, they are not
950 serious errors.

951 The Printer object also returns Job status attributes that indicate the initial state of the Job ('pending',
952 'pending-held', 'processing', etc.), etc. See Print-Job Response, [IPP-MOD] section 3.2.1.2.

953 2.2.2.9 Accept appended Document Content

954 The Printer object accepts the appended Document Content data and either starts it printing, or spools it for
955 later processing.

956 2.2.2.10 Scheduling and Starting to Process the Job

957 The Printer object uses its own configuration and implementation specific algorithms for scheduling the
958 Job in the correct processing order. Once the Printer object begins processing the Job, the Printer changes
959 the Job's state to 'processing'. If the Printer object supports PDL override (the "pdl-override-supported"
960 attribute set to 'attempted'), the implementation does its best to see that IPP attributes take precedence over
961 embedded instructions in the document data.

962 2.2.2.11 Completing the Job

963 The Printer object continues to process the Job until it can move the Job into the 'completed' state. If an
964 Cancel-Job operation is received, the implementation eventually moves the Job into the 'canceled' state. If
965 the system encounters errors during processing that do not allow it to progress the Job into a completed
966 state, the implementation halts all processing, cleans up any resources, and moves the Job into the 'aborted'
967 state.

968 2.2.2.12 Destroying the Job after completion

969 Once the Job moves to the 'completed', 'aborted', or 'canceled' state, it is an implementation decision as to
970 when to destroy the Job object and release all associated resources. Once the Job has been destroyed, the
971 Printer would return either the "client-error-not-found" or "client-error-gone" status codes for operations
972 directed at that Job.

973 Note: the Printer object SHOULD NOT re-use a "job-uri" or "job-id" value for a sufficiently long time
974 after a job has been destroyed, so that stale references kept by clients are less likely to access the wrong
975 (newer) job.

976 2.2.2.13 Interaction with "ipp-attribute-fidelity"

977 Some Printer object implementations may support "ipp-attribute-fidelity" set to 'true' and "pdl-override-
978 supported" set to 'attempted' and yet still not be able to realize exactly what the client specifies in the create
979 request. This is due to legacy decisions and assumptions that have been made about the role of job
980 instructions embedded within the document data and external job instructions that accompany the
981 document data and how to handle conflicts between such instructions. The inability to be 100% precise
982 about how a given implementation will behave is also compounded by the fact that the two special
983 attributes, "ipp-attribute-fidelity" and "pdl-override-supported", apply to the whole job rather than specific

984 values for each attribute. For example, some implementations may be able to override almost all Job
985 Template attributes except for "number-up".

986 2.3 Status codes returned by operation

987 This section lists all status codes once in the first operation (Print-Job). Then it lists the status codes that
988 are different or specialized for subsequent operations under each operation.

989 2.3.1 Printer Operations

990 2.3.1.1 Print-Job

991 The Printer object MUST return one of the following "status-code" values for the indicated reason.
992 Whether all of the document data has been accepted or not before returning the success or error response
993 depends on implementation. See Section 14 for a more complete description of each status code.

994 For the following success status codes, the Job object has been created and the "job-id", and "job-uri"
995 assigned and returned in the response:

996 successful-ok: no request attributes were substituted or ignored.

997 successful-ok-ignored-or-substituted-attributes: some supplied (1) attributes were ignored or (2)
998 unsupported attribute syntaxes or values were substituted with supported values or were ignored.
999 Unsupported attributes, attribute syntaxes, or values MUST be returned in the Unsupported
1000 Attributes group of the response.

1001 successful-ok-conflicting-attributes: some supplied attribute values conflicted with the values of other
1002 supplied attributes and were either substituted or ignored. Attributes or values which conflict with
1003 other attributes and have been substituted or ignored MUST be returned in the Unsupported
1004 Attributes group of the response as supplied by the client.
1005

1006 For the following error status codes, no job is created and no "job-id" or "job-uri" is returned:

1007 client-error-bad-request: The request syntax does not conform to the specification. The IPP object
1008 SHOULD NOT return the "status-message" operation attributes, if supported, if the "attributes-
1009 charset" in the request has not been processed.

1010 client-error-forbidden: The request is being refused for authorization or authentication reasons. The
1011 implementation security policy is to not reveal whether the failure is one of authentication or
1012 authorization.

1013 client-error-not-authenticated: Either the request requires authentication information to be supplied or
1014 the authentication information is not sufficient for authorization.

1015 client-error-not-authorized: The requester is not authorized to perform the request on the target object.
1016 client-error-not-possible: The request cannot be carried out because of the state of the system. See also
1017 'server-error-not-accepting-jobs' status code which MUST take precedence if the Printer object's
1018 "printer-accepting-jobs" attribute is 'false'.

1019 client-error-timeout: not applicable.

1020 client-error-not-found: the target object does not exist.

- 1021 client-error-gone: the target object no longer exists and no forwarding address is known.
1022 client-error-request-entity-too-large: the size of the request and/or print data exceeds the capacity of the
1023 IPP Printer to process it.
1024 client-error-request-value-too-long: the size of request variable length attribute values, such as 'text'
1025 and 'name' attribute syntaxes, exceed the maximum length specified in [IPP-MOD] for the attribute
1026 and MUST be returned in the Unsupported Attributes Group.
1027 client-error-document-format-not-supported: the document format supplied is not supported. The
1028 "document-format" attribute with the unsupported value MUST be returned in the Unsupported
1029 Attributes Group. This error SHOULD take precedence over any other 'xxx-not-supported' error,
1030 except 'client-error-charset-not-supported'.
1031 client-error-attributes-or-values-not-supported: one or more supplied attributes, attribute syntaxes, or
1032 values are not supported and the client supplied the "ipp-attributes-fidelity" operation attribute with
1033 a 'true' value. They MUST be returned in the Unsupported Attributes Group as explained below.
1034 client-error-uri-scheme-not-supported: not applicable.
1035 client-error-charset-not-supported: the charset supplied in the "attributes-charset" operation attribute is
1036 not supported. The Printer's "configured-charset" MUST be returned in the response as the value of
1037 the "attributes-charset" operation attribute and used for any 'text' and 'name' attributes returned in
1038 the error response. This error SHOULD take precedence over any other error, unless the request
1039 syntax is so bad that the client's supplied "attributes-charset" cannot be determined.
1040 client-error-conflicting-attributes: one or more supplied attribute values conflicted with each other and
1041 the client supplied the "ipp-attributes-fidelity" operation attribute with a 'true' value. They MUST
1042 be returned in the Unsupported Attributes Group as explained below.
1043 server-error-internal-error: an unexpected condition prevents the request from being fulfilled.
1044 server-error-operation-not-supported: not applicable (since Print-Job is REQUIRED).
1045 server-error-service-unavailable: the service is temporarily overloaded.
1046 server-error-version-not-supported: the version in the request is not supported. The "closest" version
1047 number supported MUST be returned in the response.
1048 server-error-device-error: a device error occurred while receiving or spooling the request or document
1049 data or the IPP Printer object can only accept one job at a time.
1050 server-error-temporary-error: a temporary error such as a buffer full write error, a memory overflow, or
1051 a disk full condition occurred while receiving the request and/or the document data.
1052 server-error-not-accepting-jobs: the Printer object's "printer-is-not-accepting-jobs" attribute is 'false'.
1053 server-error-busy: the Printer is too busy processing jobs to accept another job at this time.
1054 server-error-job-canceled: the job has been canceled by an operator or the system while the client was
1055 transmitting the document data.

1056 2.3.1.2 Print-URI

1057 All of the Print-Job status codes described in Section 3.2.1.2 Print-Job Response are applicable to Print-
1058 URI with the following specializations and differences. See Section 14 for a more complete description of
1059 each status code.

- 1060 server-error-uri-scheme-not-supported: the URI scheme supplied in the "document-uri" operation
1061 attribute is not supported and is returned in the Unsupported Attributes group.

1062 2.3.1.3 Validate-Job

1063 All of the Print-Job status codes described in Section 3.2.1.2 Print-Job Response are applicable to Validate-
1064 Job. See Section 14 for a more complete description of each status code.

1065 2.3.1.4 Create-Job

1066 All of the Print-Job status codes described in Section 3.2.1.2 Print-Job Response are applicable to Create-
1067 Job with the following specializations and differences. See Section 14 for a more complete description of
1068 each status code.

1069 server-error-operation-not-supported: the Create-Job operation is not supported.

1070 2.3.1.5 Get-Printer-Attributes

1071 All of the Print-Job status codes described in Section 3.2.1.2 Print-Job Response are applicable to the Get-
1072 Printer-Attributes operation with the following specializations and differences. See Section 14 for a more
1073 complete description of each status code.

1074 For the following success status codes, the requested attributes are returned in Group 3 in the response:

1075 successful-ok: no request attributes were substituted or ignored (same as Print-Job) and no requested
1076 attributes were unsupported.

1077 successful-ok-ignored-or-substituted-attributes: same as Print-Job, except the "requested-attributes"
1078 operation attribute MAY, but NEED NOT, be returned with the unsupported values.

1079 successful-ok-conflicting-attributes: same as Print-Job.

1080 For the error status codes, Group 3 is returned containing no attributes or is not returned at all:

1081 client-error-not-possible: Same as Print-Job, in addition the Printer object is not accepting any requests.

1082 client-error-request-entity-too-large: same as Print-job, except that no print data is involved.

1083 client-error-attributes-or-values-not-supported: not applicable, since unsupported operation attributes
1084 MUST be ignored and 'successful-ok-ignored-or-substituted-attributes' returned.

1085 client-error-conflicting-attributes: same as Print-Job, except that "ipp-attribute-fidelity" is not involved.

1086 server-error-operation-not-supported: not applicable (since Get-Printer-Attributes is REQUIRED).

1087 server-error-device-error: same as Print-Job, except that no document data is involved.

1088 server-error-temporary-error: same as Print-Job, except that no document data is involved.

1089 server-error-not-accepting-jobs: not applicable..

1090 server-error-busy: same as Print-Job, except the IPP object is too busy to accept even query requests.

1091 server-error-job-canceled: not applicable..

1092 2.3.1.6 Get-Jobs

1093 All of the Print-Job status codes described in Section 3.2.1.2 Print-Job Response are applicable to the Get-
1094 Jobs operation with the following specializations and differences. See Section 14 for a more complete
1095 description of each status code.

1096 For the following success status codes, the requested attributes are returned in Group 3 in the response:

- 1097 successful-ok: no request attributes were substituted or ignored (same as Print-Job) and no requested
 1098 attributes were unsupported.
- 1099 successful-ok-ignored-or-substituted-attributes: same as Print-Job, except the "requested-attributes"
 1100 operation attribute MAY, but NEED NOT, be returned with the unsupported values.
- 1101 successful-ok-conflicting-attributes: same as Print-Job.
- 1102 For any error status codes, Group 3 is returned containing no attributes or is not returned at all. The
 1103 following brief error status code descriptions contain unique information for use with Get-Jobs operation.
 1104 See section 14 for the other error status codes that apply uniformly to all operations:
- 1105 client-error-not-possible: Same as Print-Job, in addition the Printer object is not accepting any requests.
 1106 client-error-request-entity-too-large: same as Print-job, except that no print data is involved.
 1107 client-error-document-format-not-supported: not applicable.
 1108 client-error-attributes-or-values-not-supported: not applicable, since unsupported operation attributes
 1109 MUST be ignored and 'successful-ok-ignored-or-substituted-attributes' returned.
 1110 client-error-conflicting-attributes: same as Print-Job, except that "ipp-attribute-fidelity" is not involved.
 1111 server-error-operation-not-supported: not applicable (since Get-Jobs is REQUIRED).
 1112 server-error-device-error: same as Print-Job, except that no document data is involved.
 1113 server-error-temporary-error: same as Print-Job, except that no document data is involved.
 1114 server-error-not-accepting-jobs: not applicable.
 1115 server-error-job-canceled: not applicable.
- 1116 2.3.2 Job Operations
- 1117 2.3.2.1 Send-Document
- 1118 All of the Print-Job status codes described in Section 3.2.1.2 Print-Job Response are applicable to the Get-
 1119 Printer-Attributes operation with the following specializations and differences. See Section 14 for a more
 1120 complete description of each status code.
- 1121 For the following success status codes, the document has been added to the specified Job object and the
 1122 job's "number-of-documents" attribute has been incremented:
- 1123 successful-ok: no request attributes were substituted or ignored (same as Print-Job).
 1124 successful-ok-ignored-or-substituted-attributes: same as Print-Job.
 1125 successful-ok-conflicting-attributes: same as Print-Job.
- 1126 For the error status codes, no document has been added to the Job object and the job's "number-of-
 1127 documents" attribute has not been incremented:
- 1128 client-error-not-possible: Same as Print-Job, except that the Printer's "printer-is-accepting-jobs"
 1129 attribute is not involved, so that the client is able to finish submitting a multi-document job after this
 1130 attribute has been set to 'true'. Another condition is that the state of the job precludes Send-
 1131 Document, i.e., the job has already been closed out by the client. However, if the IPP Printer closed
 1132 out the job due to timeout, the 'client-error-timeout' error status SHOULD be returned instead.
 1133 client-error-timeout: This request was sent after the Printer closed the job, because it has not received a
 1134 Send-Document or Send-URI operation within the Printer's "multiple-operation-time-out" period ~~to~~
 1135 ~~Print Job, Create Job, or Validate Job operations.~~
 1136 client-error-request-entity-too-large: same as Print-Job.

- 1137 client-error-conflicting-attributes: same as Print-Job, except that "ipp-attributes-fidelity" operation
1138 attribute is not involved..
- 1139 server-error-operation-not-supported: the Send-Document request is not supported.
- 1140 server-error-not-accepting-jobs: not applicable.
- 1141 server-error-job-canceled: the job has been canceled by an operator or the system while the client was
1142 transmitting the data.
- 1143 2.3.2.2 Send-URI
- 1144 All of the Print-Job status code descriptions in Section 3.2.1.2 Print-Job Response with the specializations
1145 described for Send-Document are applicable to Send-URI. See Section 14 for a more complete description
1146 of each status code.
- 1147 server-error-uri-scheme-not-supported: the URI scheme supplied in the "document-uri" operation
1148 attribute is not supported and the "document-uri" attribute MUST be returned in the Unsupported
1149 Attributes group.
- 1150 2.3.2.3 Cancel-Job
- 1151 All of the Print-Job status codes described in Section 3.2.1.2 Print-Job Response are applicable to Cancel-
1152 Job with the following specializations and differences. See Section 14 for a more complete description of
1153 each status code.
- 1154 For the following success status codes, the Job object is being canceled or has been canceled:
- 1155 successful-ok: no request attributes were substituted or ignored (same as Print-Job).
- 1156 successful-ok-ignored-or-substituted-attributes: same as Print-Job.
- 1157 successful-ok-conflicting-attributes: same as Print-Job.
- 1158
- 1159 For any of the error status codes, the Job object has not been canceled or was previously canceled.
- 1160 client-error-not-possible: The request cannot be carried out because of the state of the Job object
1161 ('completed', 'canceled', or 'aborted') or the state of the system.
- 1162 client-error-not-found: the target Printer and/or Job object does not exist.
- 1163 client-error-gone: the target Printer and/or Job object no longer exists and no forwarding address is
1164 known.
- 1165 client-error-request-entity-too-large: same as Print-Job, except no document data is involved.
- 1166 client-error-document-format-not-supported: not applicable.
- 1167 client-error-attributes-or-values-not-supported: not applicable, since unsupported operation attributes
1168 and values MUST be ignored.
- 1169 client-error-conflicting-attributes: same as Print-Job, except that the Printer's "printer-is-accepting-
1170 jobs" attribute is not involved.
- 1171 server-error-operation-not-supported: not applicable (Cancel-Job is REQUIRED).
- 1172 server-error-device-error: same as Print-Job, except no document data is involved.
- 1173 server-error-temporary-error: same as Print-Job, except no document data is involved.
- 1174 server-error-not-accepting-jobs: not applicable..
- 1175 server-error-job-canceled: not applicable.

1176 2.3.2.4 Get-Job-Attributes

1177 All of the Print-Job status codes described in Section 3.2.1.2 Print-Job Response are applicable to Get-Job-
1178 Attributes with the following specializations and differences. See Section 14 for a more complete
1179 description of each status code.

1180 For the following success status codes, the requested attributes are returned in Group 3 in the response:

1181 successful-ok: no request attributes were substituted or ignored (same as Print-Job) and no requested
1182 attributes were unsupported.

1183 successful-ok-ignored-or-substituted-attributes: same as Print-Job, except the "requested-attributes"
1184 operation attribute MAY, but NEED NOT, be returned with the unsupported values.

1185 successful-ok-conflicting-attributes: same as Print-Job.

1186 For the error status codes, Group 3 is returned containing no attributes or is not returned at all.

1187 client-error-not-possible: Same as Print-Job, in addition the Printer object is not accepting any requests.

1188 client-error-document-format-not-supported: not applicable.

1189 client-error-attributes-or-values-not-supported: not applicable.

1190 client-error-uri-scheme-not-supported: not applicable.

1191 client-error-conflicting-attributes: not applicable

1192 server-error-operation-not-supported: not applicable (since Get-Job-Attributes is REQUIRED).

1193 server-error-device-error: same as Print-Job, except no document data is involved.

1194 server-error-temporary-error: sane as Print-Job, except no document data is involved..

1195 server-error-not-accepting-jobs: not applicable.

1196 server-error-job-canceled: not applicable.

1197 2.4 Validate-Job

1198 The Validate-Job operation has been designed so that its implementation may be a part of the Print-Job
1199 operation. Therefore, requiring Validate-Job is not a burden on implementers. Also it is useful for client's
1200 to be able to count on its presence in all conformance implementations, so that the client can determine
1201 before sending a long document, whether the job will be accepted by the IPP Printer or not.

1202 2.5 Case Sensitivity in URIs

1203 IPP client and server implementations must be aware of the diverse uppercase/lowercase nature of URIs.
1204 RFC 2396 defines URL schemes and Host names as case insensitive but reminds us that the rest of the
1205 URL may well demonstrate case sensitivity. When creating URL's for fields where the choice is
1206 completely arbitrary, it is probably best to select lower case. However, this cannot be guaranteed and
1207 implementations MUST NOT rely on any fields being case-sensitive or case-insensitive in the URL beyond
1208 the URL scheme and host name fields.

1209 The reason that the IPP specification does not make any restrictions on URIs, is so that implementations of
1210 IPP may use off-the-shelf components that conform to the standards that define URIs, such as RFC 2396
1211 and the HTTP/1.1 specifications [RFC2068]. See these specifications for rules of matching, comparison,
1212 and case-sensitivity.

1213 It is also recommended that that System Administrators and implementations avoid creating URLs for
1214 different printers that differ only in their case. For example, don't have Printer1 and printer1 as two
1215 different IPP Printers.

1216 The HTTP/1.1 specification [RFC2068-] contains more details on comparing URLs.

1217 2.6 Natural Language Override (NLO)

1218 The 'text' and 'name' attributes each have two forms. One has an implicit natural language, and the other
1219 has an explicit natural language. The 'textWithoutLanguage' and 'textWithLanguage' are the two 'text'
1220 forms. The 'nameWithoutLanguage' and 'nameWithLanguage' are the two 'name' forms. If a receiver (IPP
1221 object or IPP client) supports an attribute with attribute syntax 'text', it MUST support both forms in a
1222 request and a response. A sender (IPP client or IPP object) MAY send either form for any such attribute.
1223 When a sender sends a WithoutLanguage form, the implicit natural language is specified in the "attributes-
1224 natural-language" operation attribute which all senders MUST include in every request and response.

1225 When a sender sends a WithLanguage form, it MAY be different from the implicit natural language
1226 supplied by the sender or it MAY be the same. The receiver MUST treat either form equivalently.

1227 There is an implementation decision for senders, whether to always send the WithLanguage forms or use
1228 the WithoutLanguage form when the attribute's natural language is the same as the request or response.
1229 The former approach makes the sender implementation simpler. The latter approach is more efficient on
1230 the wire and allows inter-working with non-conforming receivers that fail to support the WithLanguage
1231 forms. As each approach have advantages, the choice is completely up to the implementer of the sender.

1232 Furthermore, when a client receives a 'text' or 'name' job attribute that it had previously supplied, that client
1233 MUST NOT expect to see the attribute in the same form, i.e., in the same WithoutLanguage or
1234 WithLanguage form as the client supplied when it created the job. The IPP object is free to transform the
1235 attribute from the WithLanguage form to the WithoutLanguage form and vice versa, as long as the natural
1236 language is preserved. However, in order to meet this latter requirement, it is usually simpler for the IPP
1237 object implementation to store the natural language explicitly with the attribute value, i.e., to store using an
1238 internal representation that resembles the WithLanguage form.

1239 The IPP Printer MUST copy the natural language of a job, i.e., the value of the "attributes-natural-
1240 language" operation attribute supplied by the client in the create operation, to the Job object as a Job
1241 Description attribute, so that a client is able to query it. In returning a Get-Job-Attributes response, the IPP
1242 object MAY return one of three natural language values in the response's "attributes-natural-language"
1243 operation attribute: (1) that requested by the requester, (2) the natural language of the job, or (3) the
1244 configured natural language of the IPP Printer, if the requested language is not supported by the IPP
1245 Printer.

1246 This "attributes-natural-language" Job Description attribute is useful for an IPP object implementation that
1247 prints start sheets in the language of the user who submitted the job. This same Job Description attribute is
1248 useful to a multi-lingual operator who has to communicate with different job submitters in different natural
1249 languages. This same Job Description attribute is expected to be used in the future to generate notification
1250 messages in the natural language of the job submitter.

1251 Early drafts of [IPP-MOD] contained a job-level natural language override (NLO) for the Get-Jobs
 1252 response. A job-level (NLO) is an (unrequested) Job Attribute which then specified the implicit natural
 1253 language for any other WithoutLanguage job attributes returned in the response for that job.
 1254 Interoperability testing of early implementations showed that no one was implementing the job-level NLO
 1255 in Get-Job responses. So the job-level NLO was eliminated from the Get-Jobs response. This
 1256 simplification makes all requests and responses consistent in that the implicit natural language for any
 1257 WithoutLanguage 'text' or 'name' form is always supplied in the request's or response's "attributes-natural-
 1258 language" operation attribute.

1259 2.7 The "queued-job-count" Printer Description attribute

1260 2.7.1 Why is "queued-job-count" RECOMMENDED?

1261 The reason that "queued-job-count" is RECOMMENDED, is that some clients look at that attribute alone
 1262 when summarizing the status of a list of printers, instead of doing a Get-Jobs to determine the number of
 1263 jobs in the queue. Implementations that fail to support the "queued-job-count" will cause that client to
 1264 display 0 jobs when there are actually queued jobs.

1265 We would have made it a REQUIRED Printer attribute, but some implementations had already been
 1266 completed before the issue was raised, so making it a SHOULD was a compromise.

1267 2.7.2 Is "queued-job-count" a good measure of how busy a printer is?

1268 The "queued-job-count" is not a good measure of how busy the printer is when there are held jobs. A
 1269 future registration could be to add a "held-job-count" (or an "active-job-count") Printer Description
 1270 attribute if experience shows that such an attribute (combination) is needed to quickly indicate how busy a
 1271 printer really is.

1272 2.8 Sending empty attribute groups

1273 The [IPP-MOD] and [IPP-PRO] specifications RECOMMEND that a sender not send an empty attribute
 1274 group in a request or a response. However, they REQUIRE a receiver to accept an empty attribute group as
 1275 equivalent to the omission of that group. allows empty groups to be sent by a "sender" as being entirely
 1276 equivalent to omitting the group entirely. The term "sender" means a client for a request and an IPP object
 1277 for a response. So a client ~~SHOULD~~MAY omit the Job Template Attributes group entirely, ~~or send an~~
 1278 ~~empty group~~ in a create operation that is not supplying any Job Template attributes. Similarly, an IPP
 1279 object ~~MAY SHOULD~~ return omit an empty Unsupported Attributes group ~~or MAY omit the group~~
 1280 ~~entirely~~, if there are no unsupported attributes to be returned in a response.

1281 ~~The [IPP-PRO] specification REQUIRES a receiver to be able to receive either an empty attribute group or~~
 1282 ~~an omitted attribute group and treat them equivalently. The term "receiver" means an IPP object for a~~
 1283 ~~request and a client for a response.~~

1284 ~~Issues 1.16 and 1.17 clarified the [IPP-MOD] to agree with [IPP-PRO] concerning empty attributes being~~
 1285 ~~equivalent to omitted attribute groups in requests and responses.~~

1286 2.9 Returning unsupported attributes in Get-Xxxx responses

1287 The client cannot depend on getting unsupported attributes returned in the Unsupported Attributes group of
1288 Get-Printer-Attributes, Get-Jobs, or Get-Job-Attributes responses that the client requested, but are not
1289 supported by the IPP object. However, such unsupported requested attributes will not be returned in the
1290 Job Attributes or Printer Attributes group (since they are unsupported). However, the IPP object is
1291 REQUIRED to return the 'successful-ok-ignored-or-substituted-attributes' status code, so that the client
1292 know that all that was requested has not been returned.

1293 2.10 Returning job-state in Print-Job response

1294 An IPP client submits a small job via Print-Job. By the time the IPP printer/print server is putting together
1295 a response to the operation, the job has finished printing and been removed as an object from the print
1296 system. What should the job-state be in the response?

1297 The Model suggests that the Printer return a response before it even accepts the document content ~~(see~~
1298 ~~Implementer's Guide)~~. The Job Object Attributes are returned only if the IPP object returns one of the
1299 success status codes. Then the job-state would always be "pending" or "pending-held".

1300 This issue comes up for the implementation of an IPP Printer object as a server that forwards jobs to
1301 devices that do not provide job status back to the server. If the server is reasonably certain that the job
1302 completed successfully, then it should return the job-state as 'completed'. Also the server can keep the job
1303 in its "job history" long after the job is no longer in the device. Then a user could query the server and see
1304 that the job was in the 'completed' state and completed as specified by the job's "time-at-completed" time
1305 which would be the same as the server submitted the job to the device.

1306 An alternative is for the server to respond to the client before or while sending the job to the device, instead
1307 of waiting until the server has finished sending the job to the device. In this case, the server can return the
1308 job's state as 'pending' with the 'job-outgoing' value in the job's "job-state-reasons" attribute.

1309 If the server doesn't know for sure whether the job completed successfully (or at all), it could return the
1310 (out-of-band) 'unknown' value.

1311 On the other hand, if the server is able to query the device and/or setup some sort of event notification that
1312 the device initiates when the job makes state transitions, then the server can return the current job state in
1313 the Print-Job response and in subsequent queries because the server knows what the job state is in the
1314 device (or can query the device).

1315 All of these alternatives depend on implementation of the server and the device.

1316 2.11 Multi-valued attributes

1317 What is the attribute syntax for a multi-valued attribute? Since some attributes support values in more than
1318 one data type, such as "media", "job-hold-until", and "job-sheets", IPP semantics associate the attribute
1319 syntax with each value, not with the attribute as a whole. The protocol associates the attribute syntax tag

1320 with each value. Don't be fooled, just because the attribute syntax tag comes before the attribute keyword.
1321 All attribute values after the first have a zero length attribute keyword as the indication of a subsequent
1322 value of the same attribute.

1323 3 Encoding and Transport

1324 ~~This section discusses various aspects of IPP/1.0 Encoding and Transport [IPP-PRO].~~

1325 ~~HTTP/1.1 is the transport layer for this protocol.~~

1326 ~~The operation layer has been designed with the assumption that the transport layer contains the following~~
1327 ~~information:~~

- 1328 ~~□ the URI of the target job or printer operation~~
- 1329 ~~□ the total length of the data in the operation layer, either as a single length or as a sequence of chunks~~
1330 ~~each with a length.~~

1331 ~~It is REQUIRED that a printer implementation support HTTP over the IANA assigned Well Known Port~~
1332 ~~631 (the IPP default port), though a printer implementation may support HTTP over port some other port as~~
1333 ~~well. In addition, a printer may have to support another port for privacy (See Section 5 "Security~~
1334 ~~Considerations").~~

1335 ~~Note: even though port 631 is the IPP default, port 80 remains the default for an HTTP URI. Thus a URI~~
1336 ~~for a printer using port 631 MUST contain an explicit port, e.g. "http://forest:631/pinetree".~~

1337 ~~Note: Consistent with RFC 2068 (HTTP/1.1), HTTP URI's for IPP implicitly reference port 80. If a URI~~
1338 ~~references some other port, the port number MUST be explicitly specified in the URI.~~

1339 ~~Each HTTP operation MUST use the POST method where the request URI is the object target of the~~
1340 ~~operation, and where the "Content-Type" of the message body in each request and response MUST be~~
1341 ~~"application/ipp". The message body MUST contain the operation layer and MUST have the syntax~~
1342 ~~described in section 3.2 "Syntax of Encoding". A client implementation MUST adhere to the rules for a~~
1343 ~~client described in RFC 2068 [rfc2068]. A printer (server) implementation MUST adhere the rules for an~~
1344 ~~origin server described in RFC 2068.~~

1345 The IPP layer doesn't have to deal with chunking. In the context of CGI scripts, the HTTP layer removes
1346 any chunking information in the received data.

1347 A client MUST NOT expect a response from an IPP server until after the client has sent the entire response.
1348 But a client MAY listen for an error response that an IPP server MAY send before it receives all the data.
1349 In this case a client, if chunking the data, can send a premature zero-length chunk to end the request before
1350 sending all the data. If the request is blocked for some reason, a client MAY determine the reason by
1351 opening another connection to query the server.

1352 In the following sections, there are a tables of all HTTP headers which describe their use in an IPP client or
1353 server. The following is an explanation of each column in these tables.

- 1354 • the "header" column contains the name of a header

- 1355 • the “request/client” column indicates whether a client sends the header.
 1356 • the “request/ server” column indicates whether a server supports the header when received.
 1357 • the “response/ server” column indicates whether a server sends the header.
 1358 • the “response /client” column indicates whether a client supports the header when received.
 1359 • the “values and conditions” column specifies the allowed header values and the conditions for the
 1360 header to be present in a request/response.

1361 The table for “request headers” does not have columns for responses, and the table for “response headers”
 1362 does not have columns for requests.

1363 The following is an explanation of the values in the “request/client” and “response/ server” columns.

- 1364 • **must:** the client or server **MUST** send the header,
 1365 • **must-if:** the client or server **MUST** send the header when the condition described in the “values and
 1366 conditions” column is met,
 1367 • **may:** the client or server **MAY** send the header
 1368 • **not:** the client or server **SHOULD NOT** send the header. It is not relevant to an IPP
 1369 implementation.

1370 The following is an explanation of the values in the “response/client” and “request/ server” columns.

- 1371 • **must:** the client or server **MUST** support the header,
 1372 • **may:** the client or server **MAY** support the header
 1373 • **not:** the client or server **SHOULD NOT** support the header. It is not relevant to an IPP
 1374 implementation.

1375 3.1 General Headers

1376 The following is a table for the general headers.

General-Header	Request		Response		Values and Conditions
	Client	Server	Server	Client	
Cache-Control	must	not	must	not	“no-cache” only
Connection	must-if	must	must-if	must	“close” only. Both client and server SHOULD keep a connection for the duration of a sequence of operations. The client and server MUST include this header for the last operation in such a sequence.
Date	may	may	must	may	per RFC 1123 [RFC1123] from RFC 2068 [RFC2068]
Pragma	must	not	must	not	“no-cache” only

General-Header	Request		Response		Values and Conditions
	Client	Server	Server	Client	
Transfer-Encoding	must-if	must	must-if	must	“chunked” only . Header MUST be present if Content-Length is absent.
Upgrade	not	not	not	not	
Via	not	not	not	not	

1377 3.2 Request Headers

1378 The following is a table for the request headers.

Request-Header	Client	Server	Request Values and Conditions
Accept	may	must	“application/ipp” only. This value is the default if the client omits it
Accept-Charset	not	not	Charset information is within the application/ipp entity
Accept-Encoding	may	must	empty and per RFC 2068 [RFC2068] and IANA registry for content-codings
Accept-Language	not	not	language information is within the application/ipp entity
Authorization	must-if	must	per RFC 2068. A client MUST send this header when it receives a 401 “Unauthorized” response and does not receive a “Proxy-Authenticate” header.
From	not	not	per RFC 2068. Because RFC recommends sending this header only with the user’s approval, it is not very useful
Host	must	must	per RFC 2068
If-Match	not	not	
If-Modified-Since	not	not	
If-None-Match	not	not	
If-Range	not	not	
If-Unmodified-Since	not	not	
Max-Forwards	not	not	

Request-Header	Client	Server	Request Values and Conditions
Proxy-Authorization	must-if	not	per RFC 2068. A client MUST send this header when it receives a 401 "Unauthorized" response and a "Proxy-Authenticate" header.
Range	not	not	
Referer	not	not	
User-Agent	not	not	

1379 3.3 Response Headers

1380 The following is a table for the request headers.

Response-Header	Server	Client	Response Values and Conditions
Accept-Ranges	not	not	
Age	not	not	
Location	must-if	may	per RFC 2068. When URI needs redirection.
Proxy-Authenticate	not	must	per RFC 2068
Public	may	may	per RFC 2068
Retry-After	may	may	per RFC 2068
Server	not	not	
Vary	not	not	
Warning	may	may	per RFC 2068
WWW-Authenticate	must-if	must	per RFC 2068. When a server needs to authenticate a client.

1381 3.4 Entity Headers

1382 The following is a table for the entity headers.

Entity-Header	Request		Response		Values and Conditions
	Client	Server	Server	Client	
Allow	not	not	not	not	
Content-Base	not	not	not	not	

Entity-Header	Request		Response		Values and Conditions
	Client	Server	Server	Client	
Content-Encoding	may	must	must	must	per RFC 2068 and IANA registry for content codings.
Content-Language	not	not	not	not	Application/ipp handles language
Content-Length	must-if	must	must-if	must	the length of the message-body per RFC 2068. Header MUST be present if Transfer-Encoding is absent..
Content-Location	not	not	not	not	
Content-MD5	may	may	may	may	per RFC 2068
Content-Range	not	not	not	not	
Content-Type	must	must	must	must	“application/ipp” only
ETag	not	not	not	not	
Expires	not	not	not	not	
Last-Modified	not	not	not	not	

1383 3.5 Optional support for HTTP/1.0

1384 IPP implementations consist of an HTTP layer and an IPP layer. In the following discussion, the term
 1385 "client" refers to the HTTP client layer and the term "server" refers to the HTTP server layer. The
 1386 Encoding and Transport document [IPP-PRO] requires that HTTP 1.1 **MUST** be supported by all clients
 1387 and all servers. However, a client and/or a server implementation may choose to also support HTTP 1.0.

1388 • This option means that a server may choose to communicate with a (non-conforming) client that only
 1389 supports HTTP 1.0. In such cases the server should not use any HTTP 1.1 specific parameters or
 1390 features and should respond using HTTP version number 1.0.

1391 • This option also means that a client may choose to communicate with a (non-conforming) server that
 1392 only supports HTTP 1.0. In such cases, if the server responds with an HTTP ‘unsupported version
 1393 number’ to an HTTP 1.1 request, the client should retry using HTTP version number 1.0.

1394 3.6 HTTP/1.1 Chunking

1395 Clients ~~should~~**MUST** anticipate that the HTTP/1.1 server may chunk responses and ~~should~~**MUST** accept
 1396 them in responses. However, a (non-conforming) HTTP client that is unable to accept chunked responses

1397 may attempt to request an HTTP 1.1 server not to ~~not~~ use chunking in its response to an operation by using
1398 the following HTTP header:

1399 TE: identity

1400 This mechanism should not be used by a server to disable a client from chunking a request, since chunking
1401 of document data is an important feature for clients to send long documents.

1402

1403 4 References

1404 ~~[IPP-HG]~~

1405 ~~————— Hastings, T., Manros, C., "Internet Printing Protocol/1.0: Implementer's Guide", draft-ietf-ipp-~~
1406 ~~implementors-guide-00.txt, November 1998.~~

1407 [IPP LPD]

1408 Herriot, R., Hastings, T., Jacobs, N., Martin, J., "Mapping between LPD and IPP Protocols", draft-
1409 ietf-ipp-lpd-ipp-map-04.txt, June 1998.

1410 [IPP-MOD]

1411 ————— R. deBry, T. Hastings, R. Herriot, S. Isaacson, P. Powell, "Internet Printing Protocol/1.0: Model and
1412 Semantics", draft-ietf-ipp-model-11.txt, November, 1998.

1413 [IPP-PRO]

1414 Herriot, R., Butler, S., Moore, P., Tuner, R., "Internet Printing Protocol/1.0: Encoding and
1415 Transport", draft-ietf-ipp-pro-06.txt, June, 1998.

1416 [IPP-RAT]

1417 Zilles, S., "Rationale for the Structure and Model and Protocol for the Internet Printing Protocol",
1418 draft-ietf-ipp-rat-03.txt, June, 1998.

1419 [IPP-REQ]

1420 Wright, D., "Design Goals for an Internet Printing Protocol", draft-ietf-ipp-req-02.txt, June, 1998.

1421 [RFC1123]

1422 ————— Braden, S., "Requirements for Internet Hosts - Application and Support", RFC 1123, October, 1989.

1423 [RFC2068]

1424 ————— R Fielding, et al, "Hypertext Transfer Protocol – HTTP/1.1" RFC 2068, January 1997.

1425 [rfc2119]

1426 ————— S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, March
1427 1997.

1428 [RFC2396]

1429 ————— Berners-Lee, T., Fielding, R., Masinter, L., "Uniform Resource Identifiers (URI): Generic
1430 Syntax", RFC 2396, August 1998.

1431 [\[SSL\]](#)
1432 [Netscape, The SSL Protocol, Version 3, \(Text version 3.02\), November 1996.](#)

1433

1434 4.1 Authors' Address

1435 Thomas N. Hastings
1436 Xerox Corporation
1437 701 Aviation Blvd.
1438 El Segundo, CA 90245
1439 hastings@cp10.es.xerox.com

1440

1441 Carl-Uno Manros
1442 Xerox Corporation
1443 701 Aviation Blvd.
1444 El Segundo, CA 90245
1445 manros@cp10.es.xerox.com

1446 **5 Appendix C: Full Copyright Statement**

1447 Copyright (C) The Internet Society (1998). All Rights Reserved

1448 This document and translations of it may be copied and furnished to others, and derivative works that
1449 comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and
1450 distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice
1451 and this paragraph are included on all such copies and derivative works. However, this document itself
1452 may not be modified in any way, such as by removing the copyright notice or references to the Internet
1453 Society or other Internet organizations, except as needed for the purpose of developing Internet standards
1454 in which case the procedures for copyrights defined in the Internet Standards process must be followed, or
1455 as required to translate it into languages other than English.

1456 The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its
1457 successors or assigns.

1458 This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET
1459 SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES,
1460 EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE
1461 OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED
1462 WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.