

1 Internet Printing Protocol WG
2 INTERNET-DRAFT
3 <draft-ietf-ipp-notify-get-07.txt>
4 Updates: RFC 2911
5 [Target category: standards track]
6 Expires: December 27, 2002
7

R. Herriot
consultant
T. Hastings
Xerox Corp.
June 27, 2002

8 Internet Printing Protocol (IPP):
9 **The ‘ippget’ Delivery Method for Event Notifications**

10 Copyright (C) The Internet Society (2002). All Rights Reserved.

11
12
13 **Status of this Memo:**

14 This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC
15 2026. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas,
16 and its working groups. Note that other groups may also distribute working documents as Internet-
17 Drafts.

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

21 The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.html>

22 The list of Internet-Draft Shadow Directories can be accessed as <http://www.ietf.org/shadow.html>.

23 **Abstract**

24 This document describes an extension to the Internet Printing Protocol/1.1: Model and Semantics (RFC
25 2911, RFC 2910). This document specifies the ‘ippget’ Delivery Method for use with the “Internet
26 Printing Protocol (IPP): Event Notifications and Subscriptions” specification. When IPP Notification
27 [ipp-ntfy] is supported, the Delivery Method defined in this document is the REQUIRED Delivery
28 Method for clients and Printers to support. They MAY support additional Delivery Methods.

29 The ‘ippget’ Delivery Method is a Pull Delivery Method. When an Event occurs, the Printer saves the
30 Event Notification for a period of time called the Event Life. The Notification Recipient fetches (pulls)
31 Event Notifications using the Get-Notifications operation. If the Notification Recipient has selected
32 the **Event Wait Mode** option to wait for additional Event Notifications, the Printer continues to return
33 Event Notifications to the Notification Recipient as Get-Notification responses as Events occur using
34 the connection originated by the Notification Recipient.

35 Either the Notification Recipient or the Printer can terminate **Event Wait Mode** without closing the
36 connection.

37

37

38 **Table of Contents**

39	1 Introduction.....	4
40	2 Terminology.....	4
41	3 Model and Operation	5
42	4 General Information.....	7
43	5 Get-Notifications operation	8
44	5.1 Get-Notifications Request.....	9
45	5.1.1 notify-subscription-ids (1setOf integer(1:MAX)).....	9
46	5.1.2 notify-sequence-numbers (1setOf integer(1:MAX)).....	9
47	5.1.3 notify-wait (boolean).....	10
48	5.2 Get-Notifications Response	10
49	5.2.1 notify-get-interval (integer(0:MAX)).....	13
50	5.2.2 printer-up-time (integer(1:MAX))	14
51	5.2.3 redirect-uri (uri)	14
52	6 Additional Information about Subscription Template Attributes	16
53	6.1 notify-pull-method (type2 keyword).....	17
54	7 Subscription Description Attributes.....	17
55	8 Additional Printer Description Attributes.....	17
56	8.1 ippget-event-life (integer(15:MAX)).....	17
57	9 New Values for Existing Printer Description Attributes	18
58	9.1 notify-pull-method-supported (1setOf type2 keyword).....	18
59	9.2 operations-supported (1setOf type2 enum).....	18
60	10 New Status Codes	18
61	10.1 successful-ok-events-complete (0x0007).....	19
62	10.2 redirection-other-site (0x0300).....	19
63	11 Encoding and Transport.....	19
64	12 Conformance Requirements.....	20
65	12.1 Conformance for IPP Printers.....	20
66	12.2 Conformance for IPP Clients.....	21
67	13 Normative References.....	22
68	14 Informative References	22

69	15 Security Considerations	23
70	15.1 Notification Recipient client access rights.....	23
71	15.2 Printer security threats	23
72	15.3 Notification Recipient security threats.....	24
73	15.4 Security requirements for Printers	24
74	15.5 Security requirements for clients	24
75	16 Internationalization Considerations	24
76	17 IANA Considerations.....	25
77	17.1 Additional attribute value registrations for existing attributes	25
78	17.1.1 Additional values for the “notify-pull-method-supported” Printer attribute	25
79	17.1.2 Additional values for the “operations-supported” Printer attribute	25
80	17.2 Operation Registrations	25
81	17.3 Attribute Registrations.....	26
82	17.4 Status code Registrations	26
83	18 Contributors	26
84	19 Authors’ Addresses.....	27
85	20 Description of Base IPP documents.....	28
86	21 Full Copyright Statement.....	29
87		
88	Table of Tables	
89	Table 1 – Information about the Delivery Method	7
90	Table 2 - Combinations of “notify-wait”, “status-code”, and “notify-get-interval”	13
91	Table 3 – Attributes in Event Notification Content.....	15
92	Table 4 – Additional Attributes in Event Notification Content for Job Events.....	16
93	Table 5 – Combinations of Events and Subscribed Events for “job-impressions-completed”	16
94	Table 6 – Additional Attributes in Event Notification Content for Printer Events	16
95	Table 7 – Operation-id assignments	18
96	Table 8 – The "event-notification-attributes-tag" value.....	20
97		
98		

98 1 Introduction

99 The “IPP Event Notifications and Subscriptions” document [ipp-ntfy] defines an OPTIONAL
100 extension to Internet Printing Protocol/1.1: Model and Semantics [RFC2911, RFC2910]. For a
101 description of the base IPP documents, see section 20. The [ipp-ntfy] extension defines operations that
102 a client can perform in order to create Subscription Objects in a Printer and carry out other operations
103 on them. A Subscription Object represents a Subscription abstraction. A client associates Subscription
104 Objects with a particular Job by performing the Create-Job-Subscriptions operation or by submitting a
105 Job with subscription information. A client associates Subscription Objects with the Printer by
106 performing a Create-Printer-Subscriptions operation. Four other operations are defined for
107 Subscription Objects: Get-Subscriptions-Attributes, Get-Subscriptions, Renew-Subscription, and
108 Cancel-Subscription. The Subscription Object specifies that when one of the specified Events occurs,
109 the Printer sends an asynchronous Event Notification to the specified Notification Recipient via the
110 specified Delivery Method (i.e., protocol).

111 The “IPP Event Notifications and Subscriptions” document [ipp-ntfy] specifies that each Delivery
112 Method is defined in another document. This document is one such document, and it specifies the
113 ‘ippget’ delivery method. If a client or Printer supports IPP Notification [ipp-ntfy], the client or Printer
114 MUST support the ‘ippget’ Delivery Method defined in this document. Such a client or Printer MAY
115 support additional Delivery Methods.

116 The ‘ippget’ Delivery Method is a Pull Delivery Method. When an Event occurs, the Printer saves the
117 Event Notification for a period of time called the Event Life. The Notification Recipient fetches (pulls)
118 the Event Notifications using the Get-Notifications operation. This operation causes the Printer to
119 return all Event Notifications held for the specified Subscription object(s). If the Notification
120 Recipient has selected the **Event Wait Mode** option to wait for additional Event Notifications, the
121 Printer continues to return Event Notifications to the Notification Recipient as Get-Notification
122 responses as Events occur using the transaction originated by the Notification Recipient.

123 The Notification Recipient can terminate **Event Wait Mode** (without closing the connection) by
124 supplying the “notify-wait” (boolean) attribute with a ‘false’ value in a subsequent Get-Notifications
125 request. Similarly, the Printer can terminate **Event Wait Mode** (without closing the connection) by
126 returning the “notify-get-interval” (integer) operation attribute in a Get-Notifications response which
127 tells the Notification Recipient how long to wait before trying again.

128 2 Terminology

129 This section defines the following terms that are used throughout this document:

130 This document uses the same terminology as [RFC2911], such as “client”, “Printer”, “Job”, “attribute”,
131 “attribute value”, “keyword”, “operation”, “request”, “response”, and “support”.

132 Capitalized terms, such as **MUST**, **MUST NOT**, **REQUIRED**, **SHOULD**, **SHOULD NOT**, **MAY**,
133 **NEED NOT**, and **OPTIONAL**, have special meaning relating to conformance as defined in RFC 2119
134 [RFC2119] and [RFC2911] section 12.1. If an implementation supports the extension defined in this

135 document, then these terms apply; otherwise, they do not. These terms define conformance to *this*
136 *document only*; they do not affect conformance to other documents, unless explicitly stated otherwise.

137 **Event Life:** The length of time in seconds after an Event occurs during which the Printer will return
138 that Event in a Event Notification in a Get-Notifications response. After the Event Life expires,
139 the Printer will no longer return an Event Notification for that Event in a Get-Notifications
140 response.

141 **Event Notification Attributes Group:** The attributes group in a response that contains attributes that
142 are part of an Event Notification.

143 **Event Wait Mode:** The mode requested by a Notification Recipient client in its Get-Notifications
144 Request and granted by a Printer to keep the connection open where the Printer sends
145 subsequent Event Notifications to the Notification Recipient as they occur as additional Get-
146 Notification Responses.

147 Other capitalized terms, such as Notification Recipient, Event, Event Notification, Compound Event
148 Notification, Printer, etc., are defined in [ipp-ntfy], have the same meanings, and are not
149 reproduced here. However, for convenience the following key terms are reproduced here:

150 **Event** – some occurrence (either expected or unexpected) within the printing system of a change of
151 state, condition, or configuration of a Job or Printer object. An Event occurs only at one instant
152 in time and does not span the time the physical Event takes place. For example, jam-occurred
153 and jam-cleared are two distinct, instantaneous Events, even though the jam may last for a
154 while.

155 **Event Notification** – the information about an Event that the Printer sends when an Event occurs.

156 3 Model and Operation

157 In a Subscription Creation Operation, when the “notify-pull-method” attribute is present and has the
158 ‘ippget’ keyword value, the client is requesting that the Printer use the ‘ippget’ Pull Delivery Method
159 for the Event Notifications associated with the new Subscription Object.

160 When an Event occurs, the Printer **MUST** generate an Event Notification and **MUST** assign it the
161 Event Life. The Printer **MUST** hold an Event Notification for its assigned Event Life.

162 When a Notification Recipient wants to receive Event Notifications for a Subscription object, it
163 performs the Get-Notifications operation supplying the Subscription object’s subscription-id, which
164 causes the Printer to return all un-expired Event Notifications held for that Subscription object. If the
165 Notification Recipient has selected the **Event Wait Mode** option to wait for additional Event
166 Notifications, the response to the Get-Notifications request continues indefinitely as the Printer
167 continues to send Event Notifications in the response as Events occur for that Subscription object.

168 When the Notification Recipient requests Event Notifications for per-Job Subscription Objects, the
169 Notification Recipient typically performs the Get-Notifications operation within a second of

170 performing the Subscription Creation operation. Because the Printer MUST save Event Notifications
171 for at least 15 seconds (see section 8.1), the Notification Recipient is unlikely to miss any Event
172 Notifications that occur between the Subscription Creation and the Get-Notifications operation.

173 The 'ippget' Delivery Method is designed primarily for (1) a client that wants to get Events (from the
174 job's per-Job Subscription object) for a job that it has submitted and (2) for a privileged client that
175 wants to get all job or printer Events from a per-Printer Subscription object.

176 **4 General Information**

177 If a Printer supports this Delivery Method, the following are its characteristics.

178 **Table 1 – Information about the Delivery Method**

Document Method Conformance Requirement	Delivery Method Realization
1. What is the URL scheme name for the Push Delivery Method or the keyword method name for the Pull Delivery Method?	'ippget' keyword method name
2. Is the Delivery Method REQUIRED, RECOMMENDED or OPTIONAL for an IPP Printer to support?	REQUIRED
3. What transport and delivery protocols does the Printer use to deliver the Event Notification Content, i.e., what is the entire network stack?	IPP with one new operation.
4. Can several Event Notifications be combined into a Compound Event Notification?	Yes.
5. Is the Delivery Method initiated by the Notification Recipient (pull), or by the Printer (push)?	This Delivery Method is a pull method with aspects of a push method, though the Printer does not initiate the connection.
6. Is the Event Notification content Machine Consumable or Human Consumable?	Machine Consumable
7. What section in this document answers the following question? For a Machine Consumable Event Notification, what is the representation and encoding of values defined in section 9.1 of [ipp-ntfy] and the conformance requirements thereof? For a Human Consumable Event Notification, what is the representation and encoding of pieces of information defined in section 9.2 of [ipp-ntfy] and the conformance requirements thereof?	Section 5
8. What are the latency and reliability of the transport and delivery protocol?	Same as IPP and the underlying HTTP transport
9. What are the security aspects of the transport and delivery protocol, e.g., how it is handled in firewalls?	Same as IPP and the underlying HTTP transport and in the same direction, so no new firewall considerations.
10. What are the content length restrictions?	None
11. What are the additional values or pieces of information that a Printer sends in an Event Notification content and the conformance requirements thereof?	None
12. What are the additional Subscription Template and/or Subscription Description attributes and the conformance requirements thereof?	None

13. What are the additional Printer Description attributes and the conformance requirements thereof?	"ipp-event-life" (integer (15: MAX))
--	--------------------------------------

179

180 5 Get-Notifications operation

181 This operation is issued by a client acting in the role of a Notification Recipient requesting the Printer
182 to return all Event Notifications held for the identified Subscription object(s).

183 A Printer **MUST** support this operation.

184 When a Printer performs this operation, it **MUST** return all and only those Event Notifications:

- 185 1. Whose associated Subscription Object's "notify-subscription-id" Subscription Description
186 attribute equals one of the values of the "notify-subscription-ids" (1setOf integer(1:MAX))
187 operation attribute AND
- 188 2. Whose associated Subscription Object's contains the "notify-pull-method" attribute and it has
189 the 'ippget' keyword value AND
- 190 3. Whose "notify-sequence-number" is equal to or greater than the corresponding value of the
191 "notify-sequence-numbers (1setOf integer(1:MAX)) operation attribute, if supplied AND
- 192 4. Whose Event Life has not yet expired AND
- 193 5. Where the Notification Recipient client has read-access rights to the identified Subscription
194 Object (see *Access Rights* paragraph below).

195 The Notification Recipient client can request **Event Wait Mode** by supplying the "notify-wait"
196 operation attribute with a 'true' value.

197 The Notification Recipient client can terminate **Event Wait Mode** (without closing the connection) by
198 supplying the "notify-wait" attribute with a 'false' value in a subsequent Get-Notifications request.
199 Similarly, the Printer can terminate **Event Wait Mode** (without closing the connection) by returning
200 the "notify-get-interval" operation attribute in a Get-Notifications response which tells the Notification
201 Recipient how long to wait before trying again.

202 The Printer **MUST** accept the request in any state (see [RFC2911] "printer-state" and "printer-state-
203 reasons" attributes) and **MUST** remain in the same state with the same "printer-state-reasons" values.

204 *Access Rights*: The authenticated user (see [RFC2911] section 8.3) performing this operation **MUST** be
205 (1) the owner of each Subscription Object identified by the "notify-subscription-ids" operation attribute
206 (see section 5.1.1), (2) an operator or administrator of the Printer (see [RFC2911] Sections 1 and 8.5),
207 or (3) be otherwise authorized by the Printer's administrator-configured security policy to request
208 Event Notifications from the target Subscription Object(s). Otherwise, the IPP Printer **MUST** reject
209 the operation and return: 'client-error-forbidden', 'client-error-not-authenticated', or 'client-error-not-

210 authorized' status code as appropriate. Furthermore, the Printer's security policy MAY limit the
211 attributes returned by the Get-Notifications operation, in a manner similar to the Get-Job-Attributes
212 operation (see [RFC2911] end of section 3.3.4.2).

213 5.1 Get-Notifications Request

214 The following groups of attributes are part of the Get-Notifications Request:

215 Group 1: Operation Attributes

216 Natural Language and Character Set:

217 The "attributes-charset" and "attributes-natural-language" attributes as described in
218 [RFC2911] section 3.1.4.1.

219

220 Target:

221 The "printer-uri" (uri) operation attribute which is the target for this operation as described in
222 [RFC2911] section 3.1.5.

223

224 Requesting User Name:

225 The "requesting-user-name" (name(MAX)) attribute SHOULD be supplied by the client as
226 described in [RFC2911] section 8.3.

227

228 5.1.1 notify-subscription-ids (1setOf integer(1:MAX))

229 This attribute identifies one or more Subscription objects for which Events are requested. The
230 client MUST supply this attribute with at least one value. The Printer object MUST support
231 this attribute with multiple values.

232

233 If no Subscription Object exists with the supplied identifier or the identified Subscription
234 Object does not contain the "notify-pull-method" attribute with the 'ippget' keyword value,
235 the Printer MUST return the 'client-error-not-found' status code.

236

237 Note: The name of both the "notify-subscription-ids" and "notify-sequence-
238 numbers" end in 's', since they are multi-valued. However, there are other
239 occurrences of these attribute names without the 's' that are single valued.

240 5.1.2 notify-sequence-numbers (1setOf integer(1:MAX))

241 This attribute specifies one or more lowest Event Notification sequence number values for the
242 Subscription objects identified by the corresponding values of the "notify-subscription-ids"
243 operation attribute. The Notification Recipient SHOULD supply this attribute and the number
244 of values SHOULD be the same as the number of values of the "notify-subscriptions-ids"
245 attribute. The Printer MUST support this attribute with multiple values.

246

247 The Printer MUST NOT return Notification Events with lower sequence numbers for the
248 corresponding Subscription object. Therefore, by supplying the proper values for this
249 attribute the Notification Recipient can prevent getting the same Event Notifications from a
250 Subscription object that were returned on a previous Get-Notifications request. The
251 Notification Recipient SHOULD remember the highest "notify-sequence-number" value
252 returned for each Subscription object requested and SHOULD pass that value for each
253 requested Subscription object on the next Get-Notifications request.

254
255 If the Notification Recipient supplies fewer values for this attribute (including omitting this
256 attribute) than for the "notify-subscription-ids" operation attribute, the Printer assumes a '1'
257 value for each missing value. A value of '1' causes the Printer to return any un-expired Event
258 Notification for that Subscription object, since '1' is the lowest possible sequence number. If
259 the Notification Recipient supplies more values for this attribute than the number of values for
260 the "notify-subscription-ids" operation attribute, the Printer ignores the extra values.

261
262 Note: If a Notification Recipient performs two consecutive Get-Notifications operations with
263 the same value for "notify-sequence-number" (or omits the attribute), the time stamp of the
264 first Event Notification in the second Get-Notifications Response may be less than the time
265 stamp of the last Event Notification in the first Get-Notification Response. This happens
266 because the Printer sends all unexpired Event Notification with a sequence number equal or
267 higher according to the ordering specified in [ipp-ntfy] and some Event Notifications from the
268 first Get-Notifications operation may not have expired by the time the second Get-
269 Notifications operation occurs.

270

271 5.1.3 notify-wait (boolean)

272 This value indicates whether or not the Notification Recipient wants **Event Wait Mode**. The
273 client MAY supply this attribute. The Printer object MUST support both values of this
274 attribute.

275

276 If the client supplies the 'false' value or omits this attribute, the client is not requesting **Event**
277 **Wait Mode**. If the value is 'true', the client is requesting **Event Wait Mode**. See the
278 beginning of section 5.2 for the rules for **Event Wait Mode**.

279 5.2 Get-Notifications Response

280 The Printer has the following options for responding to a Get-Notifications Request:

- 281 1. The Printer can reject the request and return the 'server-error-busy' status code, if the Printer is
282 too busy to accept this operation at this time. In this case, the Printer MUST return the "get-
283 notify-interval" operation attribute to indicate when the client SHOULD try again.
- 284 2. If the Notification Recipient did not request **Event Wait Mode** ("notify-wait-mode" = 'false' or
285 omitted), the Printer MUST return immediately whatever Event Notifications it currently holds
286 in the requested Subscription object(s) and MUST return the "notify-get-interval" operation

287 attribute with number of seconds from now at which the Notification Recipient SHOULD
288 repeat the Get-Notifications Request to get future Event Notifications.

289 3. If the Notification Recipient requested **Event Wait Mode** ("notify-wait-mode" = 'true'), the
290 Printer MUST return immediately whatever Event Notifications it currently holds in the
291 requested Subscription object(s) and MUST continue to return Event Notifications as they
292 occur until all of the requested Subscription Objects are canceled. A Subscription Object is
293 canceled either via the Cancel-Subscription operation or by the Printer (e.g., the Subscription
294 Object is canceled when the associated Job completes and is no longer in the Job Retention or
295 Job History phase - see the "ippget-event-life (integer(15:MAX))" attribute discussion in
296 section 8.1).

297 However, the Printer MAY decide to terminate **Event Wait Mode** at any time, including in the
298 first response. In this case the Printer MUST return the "notify-get-interval" operation attribute.
299 This attribute indicates that the Printer wishes to leave **Event Wait Mode** and the number of
300 seconds in the future that the Notification Recipient SHOULD try the Get-Notifications
301 operation again. The Notification Recipient MUST accept this response and MUST disconnect.
302 If the Notification Recipient does not disconnect, the Printer SHOULD do so.

303 From the Notification Recipient's view, the response appears as an initial burst of data, which includes
304 the Operation Attributes Group and one Event Notification Attributes Group per Event Notification
305 that the Printer is holding. After the initial burst of data, if the Notification Recipient has selected the
306 **Event Wait Mode** option to wait for additional Event Notifications, the Notification Recipient
307 receives occasional Event Notification Attribute Groups. Proxy servers may delay some Event
308 Notifications or cause time-outs to occur. The client MUST be prepared to perform the Get-
309 Notifications operation again when time-outs occur.

310 Each attribute is encoded using the IPP rules for encoding attributes [RFC2910] and MAY be encoded
311 in any order. Note: the Get-Jobs response in [RFC2911] acts as a model for encoding multiple groups
312 of attributes. See section 11 for the encoding and transport rules.

313 The following groups of attributes are part of the Get-Notifications Response:

314 Group 1: Operation Attributes

315 Status Message:

316 In addition to the REQUIRED status code returned in every response, the response
317 OPTIONALLY includes a "status-message" (text(255)) and/or a "detailed-status-message"
318 (text(MAX)) operation attribute as described in [RFC2911] sections 13 and 3.1.6.

319
320 The Printer can return any status codes defined in [RFC2911]. If the status code is not
321 'successful-xxx', the Printer MUST NOT return any Event Notification Attribute groups. The
322 following is a description of the important status codes:

323
324 **successful-ok:** the response contains all Event Notification associated with the specified
325 subscription-ids that had been supplied in the "notify-subscription-ids" operation

326 attribute in the request. If the requested Subscription Objects have no associated
327 Event Notification, the response MUST contain zero Event Notifications.

328 **successful-ok-events-complete:** indicate when this return is the last return for all
329 Subscription objects that match the request, whether or not there are Event
330 Notifications being returned. This condition occurs for **Event Wait Mode** with
331 Notification Recipients waiting for responses when the Subscription Object is: (1)
332 canceled with a Cancel-Subscription operation, (2) deleted when the Per-Printer
333 Subscription lease time expires, or (3) when the 'job-completed' event occurs for a
334 Per-Job Subscription. This condition also occurs for a Get-Notifications request that
335 a Notification Recipient makes after the job completes, but before the Event Life
336 expires. See section 10.1.

337 **client-error-not-found:** The Printer has no Subscription Object's whose "notify-
338 subscription-id" attribute equals any of the values of the "notify-subscription-ids"
339 operation attribute supplied or the identified Subscription Object does not contain the
340 "notify-pull-method" attribute with the 'ippget' keyword value.

341 **server-error-busy:** The Printer is too busy to accept this operation. The Printer
342 SHOULD return the "notify-get-interval" operation attribute in the Operation
343 Attributes of the response, then the Notification Recipient SHOULD wait for the
344 number of seconds specified by the "notify-get-interval" operation attribute before
345 performing this operation again. If the "notify-get-interval" Operation Attribute is
346 not present, the Notification Recipient SHOULD use the normal network back-off
347 algorithms for determining when to perform this operation again.

348 **redirection-other-site:** The Printer does not handle this operation and requests the
349 Notification Recipient to perform the operation again with the uri specified by the
350 "redirect-uri" Operation Attribute in the response. See section 10.2.

351
352 Natural Language and Character Set:

353 The "attributes-charset" and "attributes-natural-language" attributes as described in
354 [RFC2911] section 3.1.4.2.

355
356 The Printer MUST use the values of "notify-charset" and "notify-natural-language",
357 respectively, from one Subscription Object associated with the Event Notifications in this
358 response.

359
360 Normally, there is only one matched Subscription Object, or the value of the "notify-charset"
361 and "notify-natural-language" attributes is the same in all Subscription Objects. If not, the
362 Printer MUST pick one Subscription Object from which to obtain the value of these attributes.
363 The algorithm for picking the Subscription Object is implementation dependent. The choice
364 of natural language is not critical because 'text' and 'name' values can override the
365 "attributes-natural-language" operation attribute. The Printer's choice of charset is critical
366 because a bad choice may leave it unable to send some 'text' and 'name' values accurately.
367

368

5.2.1 notify-get-interval (integer(0:MAX))

369

The value of this operation attribute is the number of seconds that the Notification Recipient SHOULD wait before trying the Get-Notifications operation again. The Printer MUST return this operation attribute if: (1) it is too busy to return events, (2) the Notification Recipient client did *not* request **Event Wait Mode**, or (3) the Printer is terminating Event Wait Mode. The client MUST accept this attribute and SHOULD re-issue the Get-Notifications operation (with or without "notify-wait" = 'true') the indicated number of seconds in the future in order to get more Event Notifications. This value is intended to help the client be a good network citizen.

377

378

The value of this attribute MUST be at least as large as the value of the Printer's "ippget-event-life" Printer Description attribute (see section 8.1). The Printer MAY return a value that is larger than the value of the "ippget-event-life" Printer Description attribute provided that the Printer increases the Event Life for this Subscription object, so that Notification Recipients taking account of the larger value and polling with a longer interval will *not* miss events. Note; implementing such an algorithm requires some hidden attributes in the Subscription object that are IMPLEMENTATION DEPENDENT.

379

380

381

382

383

384

385

If the Printer wants to remain in **Event Wait Mode**, then the Printer MUST NOT return this attribute in the response.

386

387

388

389

Here is a complete table of combinations of "notify-wait", "status-code", "notify-get-interval", and Event Notification Attributes Groups for Get-Notification initial (Wait and No Wait) Responses and subsequent **Event Wait Mode** Responses (which may be staying in **Event Wait Mode** or may be requesting the Notification Recipient to leave **Event Wait Mode**):

390

391

392

393

394

Table 2 - Combinations of "notify-wait", "status-code", and "notify-get-interval"

client sends: "notify-wait"	Printer returns: "status-code"	Printer returns: "notify-get-interval"	Event Notification Attribute Groups
1. 'false'*	'successful-ok'	MUST return N	maybe
2. 'false'*	'not-found'	MUST NOT	MUST NOT
3. 'false'*	'busy'	MUST return N	MUST NOT
4. 'false'*	'events-complete'	MUST NOT	'job-completed'
5. 'true'	'successful-ok'	MUST NOT	MUST
6. 'true'	'successful-ok'	MUST return N	maybe
7. 'true'	'not-found'	MUST NOT	MUST NOT
8. 'true'	'busy'	MUST return N	MUST NOT
9. 'true'	'events-complete'	MUST NOT	'job-completed' or maybe other

395

* 'false' or client omits the "notify-wait" attribute.

396

397

Explanation:

398

399

1-4: client does *not* request **Event Wait Mode**

400

5-9: client requests **Event Wait Mode**

401

2,7: Subscription object not found, or was canceled earlier; client should NOT try again.

402

3,8: server busy, tells client to try later; client should try again in N seconds.

403

4: client polled after job completed, but before Event Life expired, and got the ‘job-

404

completed’ event, so the client shouldn’t bother trying again; client should NOT try again later.

406

5: Printer returns one or more Event Notifications and is OK to stay in **Event Wait Mode**; the client waits for more Event Notifications to be returned.

407

408

6: Printer wants to leave **Event Wait mode**. Can happen on the first response (with or without Event Notifications) or happen on a subsequent response with or without Event Notifications; the client SHOULD try again in N seconds.

409

410

9: Printer either (1) returns ‘job-completed’ event or (2) the Subscription Object was canceled by either a Cancel-Job or a Per-Printer Subscription expired without being renewed. For case (1), at least one Event Notification MUST be returned, while for case (2), it is unlikely that any Event Notifications are returned; the client should NOT try again.

411

412

413

414

415

5.2.2 printer-up-time (integer(1:MAX))

416

The value of this attribute is the Printer’s “printer-up-time” attribute at the time the Printer sends this response. The Printer MUST return this attribute. Because each Event Notification also contains the value of this attribute when the event occurred, the value of this attribute lets a Notification Recipient know when each Event Notification occurred relative to the time of this response.

417

418

419

420

421

5.2.3 redirect-uri (uri)

422

The value of this attribute is the uri that the Notification Recipient MUST use for a subsequent Get-Notifications operation. The Printer MAY support this attribute. This attribute MUST be returned in the Operation Attributes Group if and only if the Printer returns the ‘redirection-other-site’ status code (see section 10.2).

423

424

425

426

427

Group 2: Unsupported Attributes

428

See [RFC2911] section 3.1.7 for details on returning Unsupported Attributes.

429

430

431

Group 3 through N: Event Notification Attributes

432

The Printer responds with one Event Notification Attributes Group per matched Event Notification. The entire response is considered a single Compound Event Notification (see [ipp-ntfy]). The matched Event Notifications are all un-expired Event Notification associated with the matched Subscription Objects and MUST follow the “Event Notification Ordering” requirements for Event Notifications within a Compound Event Notification specified in [ipp-

433

434

435

436

437 ntfy] section 9. In other words, the Printer MUST order these Event Notification groups in
 438 ascending time stamp (and sequence number) order for a Subscription object. If Event
 439 Notifications for multiple Subscription objects are being returned, the Notification Events for
 440 the next Subscription object follow in ascending time stamp order, etc.

441
 442 Each Event Notification Group MUST contain all of attributes specified in section 9.1
 443 (“Content of Machine Consumable Event Notifications”) of [ipp-ntfy] with exceptions
 444 denoted by asterisks in the tables below.

445
 446 The tables below are copies of the tables in section 9.1 (“Content of Machine Consumable
 447 Event Notifications”) of [ipp-ntfy] except that each cell in the “Sends” column is a “MUST”.

448
 449 If more than one Event Notification is being returned and the status of each is not the same,
 450 then the Printer MUST return a “notify-status-code” attribute in each Event Notification
 451 Attributes group to indicate the differing status values.

452
 453 For an Event Notification for all Events, the Printer includes the attributes shown in Table 3.

454 **Table 3 – Attributes in Event Notification Content**

Source Value	Sends	Source Object
notify-subscription-id (integer(1:MAX))	MUST	Subscription
notify-printer-uri (uri)	MUST	Subscription
notify-subscribed-event (type2 keyword)	MUST	Event Notification
printer-up-time (integer(1:MAX)) *	MUST	Printer
printer-current-time (dateTime)	MUST **	Printer
notify-sequence-number (integer (0:MAX))	MUST	Subscription
notify-charset (charset)	MUST	Subscription
notify-natural-language (naturalLanguage)	MUST	Subscription
notify-user-data (octetString(63))	MUST ***	Subscription
notify-text (text)	MUST	Event Notification
attributes from the “notify-attributes” attribute	MUST ****	Printer
attributes from the “notify-attributes” attribute	MUST ****	Job
attributes from the “notify-attributes” attribute	MUST ****	Subscription

455
 456 * As specified in [ipp-ntfy] section 9, the value of the “printer-up-time” attribute sent in each
 457 Event Notification MUST be the time at which the Event occurred, not the time at which the
 458 Event Notification was sent.

459
 460 ** The Printer MUST send the “printer-current-time” attribute if and only if it supports the
 461 “printer-current-time” attribute on the Printer object.

462
 463 *** If the associated Subscription Object does not contain a “notify-user-data” attribute, the
 464 Printer MUST send an octet-string of length 0.

465

466

**** If the “notify-attributes” attribute is present on the Subscription Object, the Printer MUST send all attributes specified by the “notify-attributes” attribute. Note: if the Printer doesn’t support the “notify-attributes” attribute, it is not present on the associated Subscription Object.

467

468

469

470

471

For Event Notifications for Job Events, the Printer includes the additional attributes shown in Table 4.

472

473

Table 4 – Additional Attributes in Event Notification Content for Job Events

Source Value	Sends	Source Object
job-id (integer(1:MAX))	MUST	Job
job-state (type1 enum)	MUST	Job
job-state-reasons (1setOf type2 keyword)	MUST	Job
job-impressions-completed (integer(0:MAX))	MUST *	Job

474

475

* The Printer MUST send the “job-impressions-completed” attribute in an Event Notification only for the combinations of Events and Subscribed Events shown in Table 5.

476

477

478

Table 5 – Combinations of Events and Subscribed Events for “job-impressions-completed”

Job Event	Subscribed Job Event
‘job-progress’	‘job-progress’
‘job-completed’	‘job-completed’
‘job-completed’	‘job-state-changed’

479

480

481

For Event Notification for Printer Events, the Printer includes the additional attributes shown in Table 6.

482

483

Table 6 – Additional Attributes in Event Notification Content for Printer Events

Source Value	Sends	Source Object
printer-state (type1 enum)	MUST	Printer
printer-state-reasons (1setOf type2 keyword)	MUST	Printer
printer-is-accepting-jobs (boolean)	MUST	Printer

484

6 Additional Information about Subscription Template Attributes

485

The ‘ippget’ Delivery Method does not define any addition Subscription Template attributes. The

486

‘ippget’ Delivery Method has the same conformance requirements for Subscription Template attributes

487 as defined in [ipp-ntfy]. This section defines additional information about Subscription Template
488 attributes defined in [ipp-ntfy].

489 **6.1 notify-pull-method (type2 keyword)**

490 This Subscription Template attribute identifies the Pull Delivery Method to be used for the
491 Subscription Object (see [ipp-ntfy]). In order to support the 'ippget' Pull Delivery Method defined in
492 this document, the Printer MUST support this attribute with the following keyword value:

493 'ippget': indicates that the IPPGET Pull Delivery Method is to be used for this Subscription Object.

494 **7 Subscription Description Attributes**

495 The 'ippget' Delivery Method has the same conformance requirements for Subscription Description
496 attributes as defined in [ipp-ntfy]. The 'ippget' Delivery Method does not define any addition
497 Subscription Description attributes.

498 **8 Additional Printer Description Attributes**

499 This section defines additional Printer Description attributes for use with the 'ippget' Delivery Method.

500 **8.1 ippget-event-life (integer(15:MAX))**

501 This Printer Description attribute specifies the Event Life value that the Printer assigns to each Event,
502 i.e., the number of seconds after an Event occurs during which a Printer will return that Event in an
503 Event Notification in a Get-Notifications response. After the Event Life expires for the Event, the
504 Printer MAY no longer return an Event Notification for that Event in a Get-Notifications response.

505 The Printer MUST support this attribute if it supports the 'ippget' Delivery Method. The value MUST
506 be 15 or more (at least 15 seconds) and 60 (seconds) is the RECOMMENDED value to align with the
507 PWG Job Monitoring MIB [RFC2707] jmGeneralJobPersistence and jmGeneralAttributePersistence
508 objects.

509 For example, assume the following:

- 510 1. a client performs a Job Creation operation that creates a Subscription Object associated with the
511 'ippget' Delivery Method, AND
- 512 2. an Event associated with the new Job occurs immediately after the Subscription Object is
513 created, AND
- 514 3. the same client or some other client performs a Get-Notifications operation such that the client is
515 *connected* N seconds after the Job Creation operation.

516 Then, if N is less than the value of this attribute, the client(s) performing the Get-Notifications
 517 operations can expect not to miss any Event-Notifications, barring some unforeseen lack of memory
 518 space in the Printer. Note: The client MUST initiate the Get-Notifications a time that is sufficiently
 519 less than N seconds to account for network latency so that it is *connected* to the Printer before N
 520 seconds elapses.

521 If a Printer supports the ‘ippget’ Delivery Method, it MUST keep ‘completed’, ‘canceled’, or ‘aborted’
 522 Job objects in the Job Retention and/or Job History phases for at least as long as this attribute’s value.
 523 The Printer MAY retain jobs longer than this value. See [RFC2911] section 4.3.7.1 and the discussion
 524 in [ipp-ntfy] ‘job-completed’ event) that explains that a Notification Recipients can query the Job after
 525 receiving a ‘job-completed’ Event Notification in order to find out other information about the job that
 526 is ‘completed’, ‘aborted’, or ‘canceled’. However, this attribute has no effect on the Cancel-
 527 Subscription operation which deletes the Subscription object immediately, whether or not it contain the
 528 “notify-pull-method” attribute with the ‘ippget’ keyword value. Immediately thereafter, subsequent
 529 Get-Notifications Responses MUST NOT contain Event Notifications associated with the canceled
 530 Subscription object.

531 9 New Values for Existing Printer Description Attributes

532 This section defines additional values for existing Printer Description attributes defined in [ipp-ntfy].

533 9.1 notify-pull-method-supported (1setOf type2 keyword)

534 The following keyword value for the “notify-pull-method-supported” attribute is added in order to
 535 support the new Delivery Method defined in this document:

536 ‘ippget’ - The IPP Notification Pull Delivery Method defined in this document.

537 9.2 operations-supported (1setOf type2 enum)

538 Table 7 lists the “operation-id” value defined in order to support the new Get-Notifications operation
 539 defined in this document.

540 **Table 7 – Operation-id assignments**

Value	Operation Name
0x001C	Get-Notifications

541

542 10 New Status Codes

543 The following status codes are defined as extensions for this Delivery Method and are returned as the
 544 status code of the Get-Notifications operation in Group 1 or Group 3 to N.

545 10.1 successful-ok-events-complete (0x0007)

546 The Printer MUST return the 'successful-ok-events-complete' status code to indicate when this Get-
547 Notifications response is the last response for a Subscription object, whether or not there are Event
548 Notifications being returned. This condition occurs for **Event Wait Mode** with Notification
549 Recipients waiting for responses when the Subscription Object is: (1) canceled with a Cancel-
550 Subscription operation, (2) deleted when the Per-Printer Subscription lease time expires, or (3) when
551 the 'job-completed' event occurs for a Per-Job Subscription. This condition also occurs for a Get-
552 Notifications request that a Notification Recipient makes after the job completes, but before the Event
553 Life expires.

554 10.2 redirection-other-site (0x0300)

555 This status code means that the Printer doesn't perform that Get-Notifications operation and that the
556 "redirect-uri" operation attribute (see section 5.2.3) in the response contains the uri that the
557 Notification Recipient MUST use for performing the Get-Notifications operation. If the client issues
558 subsequent Get-Notifications operations, it MUST use the value of the "redirect-uri" operation attribute
559 returned by the Printer as the target of the operation.

560 11 Encoding and Transport

561 This section defines the encoding and transport considerations for this Delivery Method based on
562 [RFC2910].

563 The encoding of a Get-Notifications Response is modeled the Get-Jobs Response (see [RFC2911]). In
564 a Get-Notifications Response, each Event Notification Attributes Group MUST start with an 'event-
565 notification-attributes-tag' (see the section "Encodings of Additional Attribute Tags" in [ipp-ntfy]), and
566 end with an 'end-of-attributes-tag'. In addition, for **Event Wait Mode** the multi-part/related is used to
567 separate each multiple response (in time) to a single Get-Notifications Request.

568 The Printer returns Get-Notification Response as follows:

- 569 1. If the Notification Recipient client did not request **Event Wait Mode** ("notify-wait" = 'false' or
570 omitted), the Printer ends the response with an 'end-of-attributes-tag' (see [RFC2911] Get-Jobs
571 encoding) as with any operation response.
- 572 2. If the Notification Recipient client requests **Event Wait Mode** ("notify-wait" = 'true') and the
573 Printer wishes to honor the request, the Printer MUST return the response as an application/ipp
574 part inside a multi-part/related MIME media type. When one or more additional Events occur,
575 the Printer returns each as an additional Event Notification Group using a separate
576 application/ipp part under the multi-part/related type.
- 577 3. If the client requested **Event Wait Mode** ("notify-wait" = 'true'), but the Printer does not wish
578 to honor the request in the initial response but wants the client explicitly poll for Event
579 Notifications, the Printer MUST return the "notify-get-interval" operation attribute (see section

580 5.2.1). The Printer returns the response as an application/ipp part which MAY be inside an
 581 multi-part/related type. The client MUST accept this response and re-issue the Get-
 582 Notifications request in the future indicated by the value of the "notify-get-interval" attribute
 583 value..

584 4. If the client requested **Event Wait Mode** ("notify-wait" = 'true'), and the Printer initially
 585 honored the request, but later wishes to leave **Event Wait Mode**, the Printer MUST return the
 586 "notify-get-interval" operation attribute (see section 5.2.1). The Printer returns the response as
 587 an application/ipp part which MUST be inside an multi-part/related type.

588 Note: All of the above is without either the Printer or the Notification Recipient closing the connection.
 589 In fact, the connection SHOULD remain open for any subsequent IPP operations. However, either the
 590 Notification Recipient or the Printer can abnormally terminate by closing the connection. But, if the
 591 Printer closes the connection too soon after returning the response, the client may not receive the
 592 response.

593 The Printer MAY chunk the responses, but this has no significance to the IPP semantics.

594 Note: While HTTP/1.1 allows a proxy to collect chunked responses over a period of time and return
 595 them back as a single un-chunked response (with a Content Length instead). However, in practice no
 596 proxy wants to have an infinite buffer. Also no proxy want to hold up responses, since user would be
 597 furious.

598 This notification delivery method uses the IPP transport and encoding [RFC2910] for the Get-
 599 Notifications operation with the following extension allocated in [ipp-ntfy]:

600 **Table 8 – The "event-notification-attributes-tag" value**

Tag Value (Hex)	Meaning
0x07	"event-notification-attributes-tag"

601

602 12 Conformance Requirements

603 This section lists the conformance requirements for clients and Printers.

604 12.1 Conformance for IPP Printers

605 It is OPTIONAL for a Printer to support IPP Notifications as defined in [ipp-ntfy]. However, if a
 606 Printer supports IPP Notifications, the Printer MUST support the 'ippget' Delivery Method as defined
 607 in this document as one of its Delivery Methods. IPP Printers that conform to this specification:

608 1. MUST meet the conformance requirements defined in [ipp-ntfy] for a Pull Delivery Method;

- 609 2. MUST support the Get-Notifications operation defined in section 5, including **Event Wait**
610 **Mode**;
- 611 3. MUST support the Subscription Template object attributes as defined in section 6;
- 612 4. MUST support the Subscription Description object attributes as defined in section 7;
- 613 5. MUST support the "ippget-event-life" Printer Description attribute defined in section 8.1,
614 including retaining jobs in the Job Retention and/or Job History phases for at least as long as
615 the value specified by the Printer's "ippget-event-life";
- 616 6. MUST support the additional values for IPP/1.1 Printer Description attributes defined in section
617 9;
- 618 7. MUST support the 'successful-ok-events-complete' status code as described in section 10.1;
- 619 8. MUST support the "redirection-other-site" status code defined 10.2, if it redirects Get-
620 Notifications operations;
- 621 9. MUST listen for the IPP Get-Notifications operation requests on IANA-assigned well-known
622 port 631, unless explicitly configured by system administrators or site policies;
- 623 10. SHOULD NOT listen for IPP Get-Notifications operation requests on any other port, unless
624 explicitly configured by system administrators or site policies.
- 625 11. MUST meet the conformance requirements as stated in section 15.4.

626 12.2 Conformance for IPP Clients

627 It is OPTIONAL for an IPP Client to support IPP Notifications as defined in [ipp-ntfy]. However, if a
628 client supports IPP Notifications, the client MUST support the 'ippget' Delivery Method as defined in
629 this document as one of its Delivery Methods. IPP Clients that conform to this specification:

- 630 1. MUST create Subscription Objects containing the "notify-pull-method" attribute (as opposed to
631 the "notify-recipient-uri" attribute) using the 'ippget' keyword value (see section 17.1.1);
- 632 2. MUST send IPP Get-Notifications operation requests (see section 5.1) via the port specified in
633 the associated 'ipp' URL (if present) or otherwise via IANA assigned well-known port 631;
- 634 3. MUST convert the associated 'ipp' URLs for use in IPP Get-Notifications operation to their
635 corresponding 'http' URL forms for use in the HTTP layer according to the rules in section 5
636 "IPP URL Scheme" in [RFC2910].
- 637 4. MUST meet the conformance requirements as stated in section 15.5.

638 13 Normative References

- 639 [ipp-ntfy]
640 Herriot, R., and T. Hastings, "Internet Printing Protocol/1.1: IPP Event Notifications and
641 Subscriptions", <draft-ietf-ipp-not-spec-09.txt>, June 27, 2002.
- 642 [RFC2119]
643 S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119 , March 1997
- 644 [RFC2910]
645 Herriot, R., Butler, S., Moore, P., and R. Tuner, "Internet Printing Protocol/1.1: Encoding and
646 Transport", RFC 2910, September 2000.
- 647 [RFC2911]
648 deBry, R., Hastings, T., Herriot, R., Isaacson, S., and P. Powell, "Internet Printing Protocol/1.1:
649 Model and Semantics", RFC 2911, September 2000.

650 14 Informative References

- 651 [RFC2565]
652 Herriot, R., Butler, S., Moore, P., and R. Turner, "Internet Printing Protocol/1.0: Encoding and
653 Transport", RFC 2565, April 1999.
- 654 [RFC2566]
655 R. deBry, T. Hastings, R. Herriot, S. Isaacson, and P. Powell, "Internet Printing Protocol/1.0:
656 Model and Semantics", RFC 2566, April 1999.
- 657 [RFC2567]
658 Wright, D., "Design Goals for an Internet Printing Protocol", RFC 2567, April 1999.
- 659 [RFC2568]
660 Zilles, S., "Rationale for the Structure and Model and Protocol for the Internet Printing Protocol",
661 RFC 2568, April 1999.
- 662 [RFC2569]
663 Herriot, R., Hastings, T., Jacobs, N., Martin, J., "Mapping between LPD and IPP Protocols", RFC
664 2569, April 1999.
- 665 [RFC2616]
666 R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, "Hypertext
667 Transfer Protocol - HTTP/1.1", RFC 2616, June 1999.
- 668 [RFC2707]
669 Bergman, R., Hastings, T., Isaacson, S., and H. Lewis, "Job Monitoring MIB - V1.0", November
670 1999.

671 [RFC3196]
672 Hastings, T., Manros, C., Zehler, P., Kugler, C., and H. Holst, "Internet Printing Protocol/1.1:
673 Implementer's Guide", RFC3196, November 2001.

674 **15 Security Considerations**

675 The IPP Model and Semantics document [RFC2911 section 8] discusses high-level security
676 requirements (Client Authentication, Server Authentication and Operation Privacy). The IPP Transport
677 and Encoding document [RFC2910 section 8] discusses the security requirements for the IPP protocol.
678 Client Authentication is the mechanism by which the client proves its identity to the server in a secure
679 manner. Server Authentication is the mechanism by which the server proves its identity to the client in
680 a secure manner. Operation Privacy is defined as a mechanism for protecting operations from
681 eavesdropping.

682 The 'ippget' Delivery Method with its Get-Notifications operations leverages the security mechanism
683 that are used in IPP/1.1 [RFC2910 and RFC2911] without adding any additional security mechanisms
684 in order to maintain the same security support as IPP/1.1.

685 The access control model for the Get-Notifications operation defined in this document is the same as
686 the access control model for the Get-Job-Attributes operation (see [RFC2911] section 3.2.6). The
687 primary difference is that a Get-Notifications operation is directed at Subscription Objects rather than
688 at Job objects, and a returned attribute group contains Event Notification attributes rather than Job
689 object attributes.

690 **15.1 Notification Recipient client access rights**

691 The Notification Recipient client **MUST** have the following access rights to the Subscription object(s)
692 targeted by the Get-Notifications operation request:

693 The authenticated user (see [RFC2911] section 8.3) performing this operation **MUST** be (1) the
694 owner of each Subscription Object identified by the "notify-subscription-ids" operation attribute
695 (see section 5.1.1), (2) an operator or administrator of the Printer (see [RFC2911] Sections 1 and
696 8.5), or (3) be otherwise authorized by the Printer's administrator-configured security policy to
697 request Event Notifications from the target Subscription Object(s). Furthermore, the Printer's
698 security policy **MAY** limit the attributes returned by the Get-Notifications operation, in a manner
699 similar to the Get-Job-Attributes operation (see [RFC2911] end of section 3.3.4.2).

700 **15.2 Printer security threats**

701 Because the Get-Notifications operation is sent in the same direction as Job Creation operations,
702 usually by the same client, this Event Notification Delivery Method poses no additional authentication,
703 authorization, privacy, firewall, or port assignment issues above those for the IPP Get-Job-Attributes
704 and Get-Printer-Attributes operations (see [RFC2911] sections 3.2.6 and 3.2.5).

705 **15.3 Notification Recipient security threats**

706 Unwanted Events Notifications (spam): Unlike Push Event Notification Delivery Methods in which
707 the IPP Printer initiates the Event Notification, with the Pull Delivery Method defined in this
708 document, the Notification Recipient is the client who initiates the Get-Notifications operation (see
709 section 5). Therefore, there is no chance of "spam" notifications with this method.

710 Note: when a client stays connected to a Printer using the Event Wait Mode (see section 5.1.3) in order
711 to receive Event Notifications as they occur, such a client can close down the IPP connection at any
712 time, and so can avoid future unwanted Event Notifications at any time.

713 It is true that client has control about whether to ask for Event Notifications. However, if the client
714 subscribes to an event, and does a Get-Notifications request, the client gets all events for the
715 Subscription Object in the sequence number range (see section 5.1.2), not just the ones the client
716 wants. If a client subscribes to a Per-Printer Subscription job event, such as 'job-completed', and
717 someone then starts and cancels thousands of jobs, the client would have to receive these events in
718 addition to the ones the client is interested in. A client can protect itself better by subscribing to his
719 own jobs using a Per-Job Subscription, rather than creating a Per-Printer subscription whose Job events
720 apply to all jobs.

721 **15.4 Security requirements for Printers**

722 For the Get-Notifications operation defined in this document, the same Printer conformance
723 requirements apply for supporting and using Client Authentication, Server Authentication and
724 Operation Privacy as stated in [RFC2910] section 8 for all IPP operations.

725 **15.5 Security requirements for clients**

726 For the Get-Notifications operation defined in this document, the same client conformance
727 requirements apply for supporting and using Client Authentication, Server Authentication and
728 Operation Privacy as stated in [RFC2910] section 8 for all IPP operations.

729 **16 Internationalization Considerations**

730 The IPP Printer MUST localize the "notify-text" attribute as specified in section 14 of [ipp-ntfy].

731 In addition, when the client receives the Get-Notifications response, it is expected to localize the
732 attributes that have the 'keyword' attribute syntax according to the charset and natural language
733 requested in the Get-Notifications request.

734 17 IANA Considerations

735 This section contains the exact information for IANA to add to the IPP Registries according to the
736 procedures defined in RFC 2911 [RFC2911] section 6.

737 *Note to RFC Editors: Replace RFC NNNN below with the RFC number for this document, so that it*
738 *accurately reflects the content of the information for the IANA Registry.*

739 17.1 Additional attribute value registrations for existing attributes

740 This section lists additional attribute value registrations for use with existing attributes defined in other
741 documents.

742 17.1.1 Additional values for the "notify-pull-method-supported" Printer attribute

743 The following table lists the keyword value defined in this document as an additional keyword value
744 for use with the "notify-pull-method-supported" Printer attribute defined in [ipp-ntfy]. This is to be
745 registered according to the procedures in RFC 2911 [RFC2911] section 6.1.

746	keyword Attribute Values:	Ref.	Section:
747	ippget	RFC NNNN	9.1

748
749 The resulting keyword method attribute value registrations will be published in the
750 <ftp://ftp.iana.org/in-notes/iana/assignments/ipp/attribute-values/notify-pull-method-supported/>
751 area.
752

753 17.1.2 Additional values for the "operations-supported" Printer attribute

754 The following table lists the enum attribute value defined in this document as an additional type2 enum
755 value for use with the "operations-supported" Printer attribute defined in [RFC2911]. This is to be
756 registered according to the procedures in RFC 2911 [RFC2911] section 6.1.

757	type2 enum Attribute Values:	Value	Ref.	Section:
758	Get-Notifications	0x001C	RFC NNNN	9.2

759
760 The resulting enum attribute value registration will be published in the
761 <ftp://ftp.iana.org/in-notes/iana/assignments/ipp/attribute-values/operations-supported/>
762 area.
763

764 17.2 Operation Registrations

765 The following table lists the operation defined in this document. This is to be registered according to
766 the procedures in RFC 2911 [RFC2911] section 6.4.

767	Operations:	Ref.	Section:
768	Get-Notifications operation	RFC NNNN	5
769			

770 The resulting operation registration will be published in the
 771 <ftp://ftp.iana.org/in-notes/iana/assignments/ipp/operations/>
 772 area.
 773

774 17.3 Attribute Registrations

775 The following table lists the attribute defined in this document. This is to be registered according to
 776 the procedures in RFC 2911 [RFC2911] section 6.2.

777	Printer Description attributes:	Ref.	Section:
778	ippget-event-life (integer(15:MAX))	RFC NNNN	8.1
779			

780 The resulting attribute registration will be published in the
 781 <ftp://ftp.iana.org/in-notes/iana/assignments/ipp/attributes/>
 782 area.
 783

784 17.4 Status code Registrations

785 The following table lists the status code defined in this document. This is to be registered according to
 786 the procedures in RFC 2911 [RFC2911] section 6.6.

787	Status codes:	Ref.	Section:
788	successful-ok-events-complete (0x0007)	RFC NNNN	10.1
789	redirection-other-site (0x0300)	RFC NNNN	10.2
790			

791 The resulting status code registration will be published in the
 792 <ftp://ftp.iana.org/in-notes/iana/assignments/ipp/status-codes/>
 793 area.

794 18 Contributors

795 Carl Kugler and Harry Lewis contributed the basic idea of in-band "smart polling" coupled with
 796 multiple responses for a single operation on the same connection, one response for each event as it
 797 occurs. Without their continual persuasion, we would not have arrived at this Delivery Method
 798 specification and would not have been able to agree on a single REQUIRED Delivery Method for IPP.

799 Carl Kugler
 800 IBM
 801 P.O. Box 1900
 802 Boulder, CO 80301-9191
 803
 804 Phone:

805 Fax:
806 e-mail: kugler@us.ibm.com
807
808 Harry Lewis
809 IBM
810 P.O. Box 1900
811 Boulder, CO 80301-9191
812
813 Phone: 303-924-5337
814 FAX:
815 e-mail: harryl@us.ibm.com
816

817 **19 Authors' Addresses**

818
819 Robert Herriot
820 706 Colorado Ave.
821 Palo Alto, CA 94303
822
823 Phone: 650-327-4466
824 Fax: 650-327-4466
825 email: bob@herriot.com
826
827 Thomas N. Hastings
828 Xerox Corporation
829 737 Hawaii St. ESAE 231
830 El Segundo CA 90245
831
832 Phone: 310-333-6413
833 Fax: 310-333-5514
834 email: hastings@cp10.es.xerox.com
835

836

837 IPP Web Page: <http://www.pwg.org/ipp/>838 IPP Mailing List: ipp@pwg.org

839

840 To subscribe to the ipp mailing list, send the following email:

841 1) send it to majordomo@pwg.org

842 2) leave the subject line blank

843 3) put the following two lines in the message body:

844 subscribe ipp

845 end

846

847 Implementers of this specification document are encouraged to join the IPP Mailing List in order to
848 participate in any discussions of clarification issues and review of registration proposals for additional
849 attributes and values. In order to reduce spam the mailing list rejects mail from non-subscribers, so
850 you must subscribe to the mailing list in order to send a question or comment to the mailing list.

851 20 Description of Base IPP documents

852 The base set of IPP documents includes:

853 Design Goals for an Internet Printing Protocol [RFC2567]

854 Rationale for the Structure and Model and Protocol for the Internet Printing Protocol [RFC2568]

855 Internet Printing Protocol/1.1: Model and Semantics [RFC2911]

856 Internet Printing Protocol/1.1: Encoding and Transport [RFC2910]

857 Internet Printing Protocol/1.1: Implementer's Guide [[RFC3196]]

858 Mapping between LPD and IPP Protocols [RFC2569]

859

860 The "Design Goals for an Internet Printing Protocol" document takes a broad look at distributed
861 printing functionality, and it enumerates real-life scenarios that help to clarify the features that need to
862 be included in a printing protocol for the Internet. It identifies requirements for three types of users:
863 end users, operators, and administrators. It calls out a subset of end user requirements that are satisfied
864 in IPP/1.0. A few OPTIONAL operator operations have been added to IPP/1.1.

865 The "Rationale for the Structure and Model and Protocol for the Internet Printing Protocol" document
866 describes IPP from a high level view, defines a roadmap for the various documents that form the suite
867 of IPP specification documents, and gives background and rationale for the IETF working group's
868 major decisions.

869 The "Internet Printing Protocol/1.1: Model and Semantics" document describes a simplified model
870 with abstract objects, their attributes, and their operations that are independent of encoding and
871 transport. It introduces a Printer and a Job object. The Job object optionally supports multiple
872 documents per Job. It also addresses security, internationalization, and directory issues.

873 The "Internet Printing Protocol/1.1: Encoding and Transport" document is a formal mapping of the
874 abstract operations and attributes defined in the model document onto HTTP/1.1 [RFC2616]. It
875 defines the encoding rules for a new Internet MIME media type called "application/ipp". This

876 document also defines the rules for transporting over HTTP a message body whose Content-Type is
877 "application/ipp". This document defines the 'ipp' scheme for identifying IPP printers and jobs.

878 The "Internet Printing Protocol/1.1: Implementer's Guide" document gives insight and advice to
879 implementers of IPP clients and IPP objects. It is intended to help them understand IPP/1.1 and some
880 of the considerations that may assist them in the design of their client and/or IPP object
881 implementations. For example, a typical order of processing requests is given, including error
882 checking. Motivation for some of the specification decisions is also included.

883 The "Mapping between LPD and IPP Protocols" document gives some advice to implementers of
884 gateways between IPP and LPD (Line Printer Daemon) implementations.

885 **21 Full Copyright Statement**

886 Copyright (C) The Internet Society (2001). All Rights Reserved.

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

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

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

903 **Acknowledgement**

904
905 Funding for the RFC Editor function is currently provided by the Internet Society.