

1 Internet Printing Protocol WG
2 INTERNET-DRAFT
3 <draft-ietf-ipp-notify-get-076.txt>
4 Updates: RFC 2911
5 [Target category: standards track]
6 Expires: December 27May 19, 2002

R. ~~obert~~ Herriot (~~editor~~)
consultant
Thomas N. Hastings
Xerox Corp.
June 27, 2002November 19, 2001
Carl Kugler
Harry Lewis
IBM, Corp.

Internet Printing Protocol (IPP):
The 'ippget' Delivery Method for Event Notifications

Copyright (C) The Internet Society (2002~~4~~). All Rights Reserved.

Status of this Memo:

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

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

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

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

Abstract

This document describes an extension to the Internet Printing Protocol [1.1: Model and Semantics/1.0 \(IPP\)](#) [[RFC2566](#), [RFC2565](#)] and [IPP/1.1](#) [[RFC 2911](#), [RFC 2910](#)]. This document specifies the 'ippget' Delivery Method for use with the "[Internet Printing Protocol \(IPP\): Event Notifications and Subscriptions](#)" specification [[ipp-ntfy](#)]. When IPP Notification [[ipp-ntfy](#)] is supported, the Delivery Method defined in this document is ~~one of the~~ **RECOMMENDED-REQUIRED** Delivery Methods for [clients and](#) Printers to support. They MAY support additional Delivery Methods.

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

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

40

41 **Table of Contents**

42	1 Introduction.....	4
43	2 Terminology.....	4
44	3 Model and Operation	5
45	4 General Information.....	7
46	5 Get-Notifications operation	8
47	5.1 Get-Notifications Request.....	9
48	5.1.1 notify-subscription-ids (1setOf integer(1:MAX)).....	9
49	5.1.2 notify-sequence-numbers (1setOf integer(1:MAX)).....	9
50	5.1.3 notify-wait (boolean).....	10
51	5.2 Get-Notifications Response	10
52	5.2.1 notify-get-interval (integer(0:MAX)).....	13
53	5.2.2 printer-up-time (integer(1:MAX))	14
54	5.2.3 redirect-uri (uri)	14
55	6 Additional Information about Subscription Template Attributes	16
56	6.1 notify-pull-method (type2 keyword).....	17
57	7 Subscription Description Attributes.....	17
58	8 Additional Printer Description Attributes.....	17
59	8.1 ippget-event-life (integer(15:MAX)).....	17
60	9 New Values for Existing Printer Description Attributes	18
61	9.1 notify-pull-method-supported (1setOf type2 keyword).....	18
62	9.2 operations-supported (1setOf type2 enum).....	18
63	10 New Status Codes	18
64	10.1 successful-ok-events-complete (0x0007).....	19
65	10.2 redirection-other-site (0x0300).....	19
66	11 Encoding and Transport.....	19
67	12 Conformance Requirements.....	20
68	12.1 Conformance for IPP Printers.....	20
69	12.2 Conformance for IPP Clients.....	21
70	13 Normative References.....	22
71	14 Informative References	23

72	15 Security Considerations	23
73	15.1 Notification Recipient client access rights.....	24
74	15.2 Printer security threats	24
75	15.3 Notification Recipient security threats.....	24
76	15.4 Security requirements for Printers	25
77	15.5 Security requirements for clients	25
78	16 Internationalization Considerations	25
79	17 IANA Considerations.....	25
80	17.1 Additional attribute value registrations for existing attributes	25
81	17.1.1 Additional values for the “notify-pull-method-supported” Printer attribute	26
82	17.1.2 Additional values for the “operations-supported” Printer attribute	26
83	17.2 Operation Registrations	26
84	17.3 Attribute Registrations.....	26
85	17.4 Status code Registrations	27
86	18 Contributors	27
87	19 Authors’ Addresses.....	28
88	20 Description of Base IPP documents.....	29
89	21 Full Copyright Statement.....	30
90		
91	Table of Tables	
92	Table 1 – Information about the Delivery Method	7
93	Table 2 - Combinations of “notify-wait”, “status-code”, and “notify-get-interval”	13
94	Table 3 – Attributes in Event Notification Content.....	15
95	Table 4 – Additional Attributes in Event Notification Content for Job Events.....	16
96	Table 5 – Combinations of Events and Subscribed Events for “job-impressions-completed”	16
97	Table 6 – Additional Attributes in Event Notification Content for Printer Events	16
98	Table 7 – Operation-id assignments	18
99	Table 8 – The "event-notification-attributes-tag" value.....	20
100		
101		

101 1 Introduction

102 The “IPP Event Notifications and Subscriptions” document [ipp-ntfy] defines an OPTIONAL
103 extension to Internet Printing Protocol/1.1: Model and Semantics/1.0 (IPP) [RFC2566, RFC2565] and
104 [IPP/1.1](#) [RFC2911, RFC2910]. For a description of the base IPP documents, see section 20. The [ipp-
105 ntfy] extension defines operations that a client can perform in order to create Subscription Objects in a
106 Printer and carry out other operations on them. A Subscription Object represents a Subscription
107 abstraction. A client associates Subscription Objects with a particular Job by performing the Create-
108 Job-Subscriptions operation or by submitting a Job with subscription information. A client associates
109 Subscription Objects with the Printer by performing a Create-Printer-Subscriptions operation. Four
110 other operations are defined for Subscription Objects: Get-Subscriptions-Attributes, Get-Subscriptions,
111 Renew-Subscription, and Cancel-Subscription. The Subscription Object specifies that when one of the
112 specified Events occurs, the Printer sends an asynchronous Event Notification to the specified
113 Notification Recipient via the specified Delivery Method (i.e., protocol).

114 The “IPP Event Notifications and Subscriptions” document [ipp-ntfy] specifies that each Delivery
115 Method is defined in another document. This document is one such document, and it specifies the
116 ‘ippget’ delivery method. ~~When If a client or Printer supports~~ IPP Notification [ipp-ntfy] ~~is supported,~~
117 ~~the client or Printer MUST support~~ the ‘ippget’ Delivery Method defined in this document ~~is one of the~~
118 ~~RECOMMENDED Delivery Methods for Printers to support.~~ Such a client or Printer MAY support
119 additional Delivery Methods.

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

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

132 2 Terminology

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

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

136 Capitalized terms, such as **MUST**, **MUST NOT**, **REQUIRED**, **SHOULD**, **SHOULD NOT**, **MAY**,
137 **NEED NOT**, and **OPTIONAL**, have special meaning relating to conformance as defined in RFC 2119

138 [RFC2119] and [RFC2911] section 12.1. If an implementation supports the extension defined in this
139 document, then these terms apply; otherwise, they do not. These terms define conformance to *this*
140 *document only*; they do not affect conformance to other documents, unless explicitly stated otherwise.

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

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

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

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

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

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

160 3 Model and Operation

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

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

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

172 When the Notification Recipient requests Event Notifications for per-Job Subscription Objects, the
173 Notification Recipient typically performs the Get-Notifications operation within a second of
174 performing the Subscription Creation operation. Because the Printer MUST save Event Notifications
175 for at least 15 seconds (see section 8.1), the Notification Recipient is unlikely to miss any Event
176 Notifications that occur between the Subscription Creation and the Get-Notifications operation.

177 The 'ippget' Delivery Method is designed primarily for (1) a client that wants to get Events (from the
178 job's per-Job Subscription object) for a job that it has submitted and (2) for a privileged client that
179 wants to get all job or printer Events from a per-Printer Subscription object. ~~If several groups of users
180 expect to receive jobs from other users (FAX paradigm) and each group has a different designated
181 person, say, a secretary, to receive job completion Events, the Printer should be configured to support
182 multiple URLs, one for each group. Then the designated (privileged) person can run an application
183 that gets the events for jobs submitted to that URL from the per Printer Subscription object that the
184 application creates.~~

185 **4 General Information**

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

187 **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' <u>keyword method name</u>
2. Is the Delivery Method REQUIRED, RECOMMENDED or OPTIONAL for an IPP Printer to support?	<u>REQUIRED</u> RECOMMENDED
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))
--	--------------------------------------

188

189 5 Get-Notifications operation

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

192 A Printer MUST support this operation.

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

- 194 1. Whose associated Subscription Object’s “notify-subscription-id” Subscription Description
195 attribute equals one of the values of the “notify-subscription-ids” (1setOf integer(1:MAX))
196 operation attribute AND
- 197 2. Whose associated Subscription Object’s contains the “notify-pull-method” attribute and it has
198 the ‘ippget’ keyword value AND
- 199 3. Whose “notify-sequence-number” is equal to or greater than the corresponding value of the
200 “notify-sequence-numbers (1setOf integer(1:MAX)) operation attribute, if supplied AND
- 201 4. Whose Event Life has not yet expired AND
- 202 5. Where the Notification Recipient ~~client is the owner of or~~ has read-access rights to the
203 identified Subscription Object (see Access Rights paragraph below).

204 The Notification Recipient client can request **Event Wait Mode** by supplying the “notify-wait”
205 operation attribute with a ‘true’ value.

206 The Notification Recipient client can terminate **Event Wait Mode** (without closing the connection) by
207 supplying the “notify-wait” attribute with a ‘false’ value in a subsequent Get-Notifications request.
208 Similarly, the Printer can terminate **Event Wait Mode** (without closing the connection) by returning
209 the “notify-get-interval” operation attribute in a Get-Notifications response which tells the Notification
210 Recipient how long to wait before trying again.

211 The Printer MUST accept the request in any state (see [RFC2911] “printer-state” and “printer-state-
212 reasons” attributes) and MUST remain in the same state with the same “printer-state-reasons” values.

213 *Access Rights:* ~~If the policy of the Printer is to allow all users to access all Event Notifications, then the~~
214 ~~Printer MUST accept this operation from any user. Otherwise, †~~The authenticated user (see [RFC2911]
215 section 8.3) performing this operation MUST be (1) the owner of each Subscription Object identified
216 by the “notify-subscription-ids” operation attribute (see section 5.1.1), ~~(as returned during a~~
217 ~~Subscription Creation Operation) or (2) an operator or administrator of the Printer (see [RFC2911]~~
218 Sections 1 and 8.5), or (3) be otherwise authorized by the Printer’s administrator-configured security

219 [policy to request Event Notifications from the target Subscription Object\(s\)](#). Otherwise, the IPP [object](#)
220 [Printer](#) MUST reject the operation and return: 'client-error-forbidden', 'client-error-not-authenticated',
221 or 'client-error-not-authorized' status code as appropriate. [Furthermore, the Printer's security policy](#)
222 [MAY limit the attributes returned by the Get-Notifications operation, in a manner similar to the Get-](#)
223 [Job-Attributes operation \(see \[RFC2911\] end of section 3.3.4.2\)](#).

224 5.1 Get-Notifications Request

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

226 Group 1: Operation Attributes

227 Natural Language and Character Set:

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

230

231 Target:

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

234

235 Requesting User Name:

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

238

239 5.1.1 notify-subscription-ids (1setOf integer(1:MAX))

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

243

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

247

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

251 5.1.2 notify-sequence-numbers (1setOf integer(1:MAX))

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

257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281

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

If the Notification Recipient supplies fewer values for this attribute (including omitting this attribute) than for the “notify-subscription-ids” operation attribute, the Printer assumes a ‘1’ value for each missing value. A value of ‘1’ causes the Printer to return any un-expired Event Notification for that Subscription object, since ‘1’ is the lowest possible sequence number. If the Notification Recipient supplies more values for this attribute than the number of values for the “notify-subscription-ids” operation attribute, the Printer ignores the extra values.

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

282
283
284
285
286
287
288
289

5.1.3 notify-wait (boolean)

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

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

290
291
292
293
294
295
296

5.2 Get-Notifications Response

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

1. The Printer can reject the request and return the ‘server-error-busy’ status code, if the Printer is too busy to accept this operation at this time. In this case, the Printer MUST return the “get-notify-interval” operation attribute to indicate when the client SHOULD try again.
2. If the Notification Recipient did not request **Event Wait Mode** (“notify-wait-mode” = ‘false’ or omitted), the Printer MUST return immediately whatever Event Notifications it currently holds

297 in the requested Subscription object(s) and MUST return the “notify-get-interval” operation
298 attribute with number of seconds from now at which the Notification Recipient SHOULD
299 repeat the Get-Notifications Request to get future Event Notifications.

300 3. If the Notification Recipient requested **Event Wait Mode** (“notify-wait-mode” = ‘true’), the
301 Printer MUST return immediately whatever Event Notifications it currently holds in the
302 requested Subscription object(s) and MUST continue to return Event Notifications as they
303 occur until all of the requested Subscription Objects are canceled. A Subscription Object is
304 canceled either via the Cancel-Subscription operation or by the Printer (e.g., the Subscription
305 Object is canceled when the associated Job completes and is no longer in the Job Retention or
306 Job History phase - see the “ippget-event-life (integer(15:MAX))” attribute discussion in
307 section 8.1).

308 However, the Printer MAY decide to terminate **Event Wait Mode** at any time, including in the
309 first response. In this case the Printer MUST return the “notify-get-interval” operation attribute.
310 This attribute indicates that the Printer wishes to leave **Event Wait Mode** and the number of
311 seconds in the future that the Notification Recipient SHOULD try the Get-Notifications
312 operation again. The Notification Recipient MUST accept this response and MUST disconnect.
313 If the Notification Recipient does not disconnect, the Printer SHOULD do so.

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

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

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

325 Group 1: Operation Attributes

326 Status Message:

327 In addition to the REQUIRED status code returned in every response, the response
328 OPTIONALLY includes a “status-message” (text(255)) and/or a “detailed-status-message”
329 (text(MAX)) operation attribute as described in [RFC2911] sections 13 and 3.1.6.
330

331 The Printer can return any status codes defined in [RFC2911]. If the status code is not
332 ‘successful-xxx’, the Printer MUST NOT return any Event Notification Attribute groups. The
333 following is a description of the important status codes:
334

335 **successful-ok:** the response contains all Event Notification associated with the specified
336 subscription-ids that had been supplied in the “notify-subscription-ids” operation

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

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

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

352 **server-error-busy:** The Printer is too busy to accept this operation. The Printer
353 SHOULD return the "notify-get-interval" operation attribute in the Operation
354 Attributes of the response, then the Notification Recipient SHOULD wait for the
355 number of seconds specified by the "notify-get-interval" operation attribute before
356 performing this operation again. If the "notify-get-interval" Operation Attribute is
357 not present, the Notification Recipient SHOULD use the normal network back-off
358 algorithms for determining when to perform this operation again.

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

362
363 Natural Language and Character Set:

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

366
367 The Printer MUST use the values of "notify-charset" and "notify-natural-language",
368 respectively, from one Subscription Object associated with the Event Notifications in this
369 response.

370
371 Normally, there is only one matched Subscription Object, or the value of the "notify-charset"
372 and "notify-natural-language" attributes is the same in all Subscription Objects. If not, the
373 Printer MUST pick one Subscription Object from which to obtain the value of these attributes.
374 The algorithm for picking the Subscription Object is implementation dependent. The choice
375 of natural language is not critical because 'text' and 'name' values can override the
376 "attributes-natural-language" operation attribute. The Printer's choice of charset is critical
377 because a bad choice may leave it unable to send some 'text' and 'name' values accurately.
378

379

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

380

381

382

383

384

385

386

387

388

389

390

391

392

393

394

395

396

397

398

399

400

401

402

403

404

405

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.

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.

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

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**):

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

406

* ‘false’ or client omits the “notify-wait” attribute.

407

408

Explanation:

409

410

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

411

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

412

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

413

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

414

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

415

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

417

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.

418

419

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.

420

421

422

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.

423

424

425

426

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

427

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.

428

429

430

431

432

5.2.3 redirect-uri (uri)

433

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).

434

435

436

437

438

Group 2: Unsupported Attributes

439

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

440

441

442

Group 3 through N: Event Notification Attributes

443

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-

444

445

446

447

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

452
 453 Each Event Notification Group MUST contain all of attributes specified in section 9.1
 454 (“Content of Machine Consumable Event Notifications”) of [ipp-ntfy] with exceptions
 455 denoted by asterisks in the tables below.

456
 457 The tables below are copies of the tables in section 9.1 (“Content of Machine Consumable
 458 Event Notifications”) of [ipp-ntfy] except that each cell in the “Sends” column is a “MUST”.

459
 460 If more than one Event Notification is being returned and the status of each is not the same,
 461 then the Printer MUST return a “notify-status-code” attribute in each Event Notification
 462 Attributes group to indicate the differing status values.

463
 464 For an Event Notification for all Events, the Printer includes the attributes shown in Table 3.

465

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

466

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

470

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

473

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

476

477

**** 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.

478

479

480

481

482

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

483

484

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

485

486

* 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.

487

488

489

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’

490

491

492

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

493

494

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

495

6 Additional Information about Subscription Template Attributes

496

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

497

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

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

500 6.1 notify-pull-method (type2 keyword)

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

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

505 7 Subscription Description Attributes

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

509 8 Additional Printer Description Attributes

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

511 8.1 ippget-event-life (integer(15:MAX))

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

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

520 For example, assume the following:

- 521 1. a client performs a Job Creation operation that creates a Subscription Object associated with the
522 'ippget' Delivery Method, AND
- 523 2. an Event associated with the new Job occurs immediately after the Subscription Object is
524 created, AND
- 525 3. the same client or some other client performs a Get-Notifications operation such that the client is
526 *connected* *N* seconds after the Job Creation operation.

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

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

542 9 New Values for Existing Printer Description Attributes

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

544 9.1 notify-pull-method-supported (1setOf type2 keyword)

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

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

548 9.2 operations-supported (1setOf type2 enum)

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

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

Value	Operation Name
0x001C	Get-Notifications

552

553 10 New Status Codes

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

556 10.1 successful-ok-events-complete (0x0007)

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

565 10.2 redirection-other-site (0x0300)

566 This status code means that the Printer doesn't perform that Get-Notifications operation and that the
567 "redirect-uri" operation attribute ([see section 5.2.3](#)) in the response contains the uri that the
568 Notification Recipient MUST use for performing the Get-Notifications operation. If the client issues
569 subsequent Get-Notifications operations, it MUST use the value of the "redirect-uri" operation attribute
570 returned by the Printer as the target of the operation.

571 11 Encoding and Transport

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

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

579 The Printer returns Get-Notification Response as follows:

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

591 5.2.1). The Printer returns the response as an application/ipp part which MAY be inside an
 592 multi-part/related type. The client MUST accept this response and re-issue the Get-
 593 Notifications request in the future indicated by the value of the "notify-get-interval" attribute
 594 value..

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

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

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

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

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

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

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

612

613 12 Conformance Requirements

614 This section lists the conformance requirements for clients and Printers. The 'ippget' Delivery Method
 615 is RECOMMEND for Printers to support.

616 12.1 Conformance for IPP Printers

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

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

- 621 2. MUST support the Get-Notifications operation defined in section 5, including **Event Wait**
622 **Mode**;
- 623 3. MUST support the Subscription Template object attributes as defined in section 6;
- 624 4. MUST support the Subscription Description object attributes as defined in section 7;
- 625 5. MUST support the "ippget-event-life" Printer Description attribute defined in section 8.1,
626 including retaining jobs in the Job Retention and/or Job History phases for at least as long as
627 the value specified by the Printer's "ippget-event-life";
- 628 6. MUST support the additional values for IPP/1.1 Printer Description attributes defined in section
629 9;
- 630 7. MUST support the 'successful-ok-events-complete' status code as described in section 10.1;
- 631 8. MUST support the "redirection-other-site" status code defined 10.2, if it redirects Get-
632 Notifications operations;
- 633 9. MUST listen for the IPP Get-Notifications operation requests on IANA-assigned well-known
634 port 631, unless explicitly configured by system administrators or site policies;
- 635 10. SHOULD NOT listen for IPP Get-Notifications operation requests on any other port, unless
636 explicitly configured by system administrators or site policies.
- 637 11. MUST meet the conformance requirements as stated in section 15.4.

638 12.2 Conformance for IPP Clients

- 639 It is OPTIONAL for an IPP Client to support IPP Notifications as defined in [ipp-ntfy]. However, if a
640 client supports IPP Notifications, the client MUST support the 'ippget' Delivery Method as defined in
641 this document as one of its Delivery Methods. IPP Clients that conform to this specification:
- 642 1. MUST create Subscription Objects containing the "notify-pull-method" attribute (as opposed to
643 the "notify-recipient-uri" attribute) using the 'ippget' keyword value (see section 17.1.1);
- 644 2. MUST send IPP Get-Notifications operation requests (see section 5.1) via the port specified in
645 the associated 'ipp' URL (if present) or otherwise via IANA assigned well-known port 631;
- 646 3. MUST convert the associated 'ipp' URLs for use in IPP Get-Notifications operation to their
647 corresponding 'http' URL forms for use in the HTTP layer according to the rules in section 5
648 "IPP URL Scheme" in [RFC2910].
- 649 4. MUST meet the conformance requirements as stated in section 15.5.

650 **13 Normative References**651 [~~ipp-ii~~g]

652 ~~— Hastings, T., Manros, C., Kugler, K., Holst H., Zehler, P., “Internet Printing Protocol/1.1: draft-ietf-~~
653 ~~ipp-implementers-guide-v11-03.txt, work in progress, July 17, 2001~~

654 [ipp-ntfy]

655 ~~R. Herriot, R., and T. Hastings, T., Isaacson, S., Martin, J., deBry, R., Shepherd, M., Bergman, R.,~~
656 ~~“Internet Printing Protocol/1.1: IPP Event Notifications and Subscriptions”, <draft-ietf-ipp-not-~~
657 ~~spec-098.txt>, ~~November 19, 2001~~June 27, 2002.~~

658 [~~RFC2026~~]

659 ~~— S. Bradner, “The Internet Standards Process — Revision 3”, RFC 2026, October 1996.~~

660 [RFC2119]

661 S. Bradner, “Key words for use in RFCs to Indicate Requirement Levels”, RFC 2119 , March 1997

662 [~~RFC2565~~]

663 ~~— Herriot, R., Butler, S., Moore, P., and R. Turner, “Internet Printing Protocol/1.0: Encoding and~~
664 ~~Transport”, RFC 2565, April 1999.~~

665 [~~RFC2566~~]

666 ~~— R. deBry, T. Hastings, R. Herriot, S. Isaacson, and P. Powell, “Internet Printing Protocol/1.0:~~
667 ~~Model and Semantics”, RFC 2566, April 1999.~~

668 [~~RFC2567~~]

669 ~~— Wright, D., “Design Goals for an Internet Printing Protocol”, RFC 2567, April 1999.~~

670 [~~RFC2568~~]

671 ~~— Zilles, S., “Rationale for the Structure and Model and Protocol for the Internet Printing Protocol”,~~
672 ~~RFC 2568, April 1999.~~

673 [~~RFC2569~~]

674 ~~— Herriot, R., Hastings, T., Jacobs, N., Martin, J., “Mapping between LPD and IPP Protocols”, RFC~~
675 ~~2569, April 1999.~~

676 [~~RFC2616~~]

677 ~~— R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, “Hypertext~~
678 ~~Transfer Protocol — HTTP/1.1”, RFC 2616, June 1999.~~

679 [~~RFC2707~~]

680 ~~— Bergman, R., Hastings, T., Isaacson, S., and H. Lewis, “Job Monitoring MIB — V1.0”, November~~
681 ~~1999.~~

682 [RFC2910]

683 Herriot, R., Butler, S., Moore, P., and R. Tuner, ~~R.~~, “Internet Printing Protocol/1.1: Encoding and
684 Transport”, RFC 2910, September 2000.

685 [RFC2911]
686 deBry, R., Hastings, T., Herriot, R., Isaacson, S., and P. Powell, P., "Internet Printing Protocol/1.1:
687 Model and Semantics", RFC 2911, September 2000.

688 **14 Informative References**

689 [RFC2565]
690 Herriot, R., Butler, S., Moore, P., and R. Turner, "Internet Printing Protocol/1.0: Encoding and
691 Transport", RFC 2565, April 1999.

692 [RFC2566]
693 R. deBry, T. Hastings, R. Herriot, S. Isaacson, and P. Powell, "Internet Printing Protocol/1.0:
694 Model and Semantics", RFC 2566, April 1999.

695 [RFC2567]
696 Wright, D., "Design Goals for an Internet Printing Protocol", RFC 2567, April 1999.

697 [RFC2568]
698 Zilles, S., "Rationale for the Structure and Model and Protocol for the Internet Printing Protocol",
699 RFC 2568, April 1999.

700 [RFC2569]
701 Herriot, R., Hastings, T., Jacobs, N., Martin, J., "Mapping between LPD and IPP Protocols", RFC
702 2569, April 1999.

703 [RFC2616]
704 R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, "Hypertext
705 Transfer Protocol - HTTP/1.1", RFC 2616, June 1999.

706 [RFC2707]
707 Bergman, R., Hastings, T., Isaacson, S., and H. Lewis, "Job Monitoring MIB - V1.0", November
708 1999.

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

712 **15 Security Considerations**

713 The IPP Model and Semantics document [RFC2911 [section 8](#)] discusses high-level security
714 requirements (Client Authentication, Server Authentication and Operation Privacy). [The IPP Transport](#)
715 [and Encoding document \[RFC2910 section 8\]](#) discusses the security requirements for the IPP protocol.
716 Client Authentication is the mechanism by which the client proves its identity to the server in a secure
717 manner. Server Authentication is the mechanism by which the server proves its identity to the client in

718 a secure manner. Operation Privacy is defined as a mechanism for protecting operations from
719 eavesdropping.

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

723 The ~~Notification~~ access control model for the Get-Notifications operation defined in this document
724 should be is similar to the same as the IPP access control model for the Get-Job-Attributes operation
725 (see [RFC2911] section 3.2.6). The primary difference is that a Get-Notifications operation is directed
726 at Subscription Objects rather than at Job objects, and a returned attribute group contains Event
727 Notification attributes rather than Job object attributes.

728 15.1 Notification Recipient client access rights

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

731 The authenticated user (see [RFC2911] section 8.3) performing this operation MUST be (1) the
732 owner of each Subscription Object identified by the "notify-subscription-ids" operation attribute
733 (see section 5.1.1), (2) an operator or administrator of the Printer (see [RFC2911] Sections 1 and
734 8.5), or (3) be otherwise authorized by the Printer's administrator-configured security policy to
735 request Event Notifications from the target Subscription Object(s). Furthermore, the Printer's
736 security policy MAY limit the attributes returned by the Get-Notifications operation, in a manner
737 similar to the Get-Job-Attributes operation (see [RFC2911] end of section 3.3.4.2).

738 15.2 Printer security threats

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

743 15.3 Notification Recipient security threats

744 Unwanted Events Notifications (spam): Unlike ~~other~~ Push Event Notification ~~d~~Delivery ~~m~~Methods in
745 which the IPP Printer initiates the Event Notification, with the Pull Delivery ~~m~~Method defined in this
746 document, the Notification Recipient is the client who initiates the Get-Notifications operation (see
747 section 5). Therefore, there is no chance of "spam" notifications with this method.

748 Note: ~~Furthermore,~~ when a client stays connected to a Printer using the Event Wait Mode (see section
749 5.1.3) in order to receive Event Notifications as they occur, such a client can close down the HTTP IPP
750 channel connection at any time, and so can avoid future unwanted Event Notifications at any time.

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

759 **15.4 Security requirements for Printers**

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

763 **15.5 Security requirements for clients**

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

767 **16 Internationalization Considerations**

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

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

772 **17 IANA Considerations**

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

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

777 **17.1 Additional attribute value registrations for existing attributes**

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

780 17.1.1 Additional values for the “notify-pull-method-supported” Printer attribute

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

784	keyword Attribute Values:	Ref.	Section:
785	ippget	RFC NNNN	9.1

786
 787 The resulting keyword method attribute value registrations will be published in the
 788 <ftp://ftp.iana.org/in-notes/iana/assignments/ipp/attribute-values/notify-pull-method-supported/>
 789 area.
 790

791 17.1.2 Additional values for the “operations-supported” Printer attribute

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

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

797
 798 The resulting enum attribute value registration will be published in the
 799 <ftp://ftp.iana.org/in-notes/iana/assignments/ipp/attribute-values/operations-supported/>
 800 area.
 801

802 17.2 Operation Registrations

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

805	Operations:	Ref.	Section:
806	Get-Notifications operation	RFC NNNN	5

807
 808 The resulting operation registration will be published in the
 809 <ftp://ftp.iana.org/in-notes/iana/assignments/ipp/operations/>
 810 area.
 811

812 17.3 Attribute Registrations

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

815	Printer Description attributes:	Ref.	Section:
816	ippget-event-life (integer(15:MAX))	RFC NNNN	8.1

817
818 The resulting attribute registration will be published in the
819 ftp://ftp.iana.org/in-notes/iana/assignments/ipp/attributes/
820 area.
821

822 **17.4 Status code Registrations**

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

825	Status codes:	Ref.	Section:
826	successful-ok-events-complete (0x0007)	RFC NNNN	10.1
827	redirection-other-site (0x0300)	RFC NNNN	10.2
828			

829 The resulting status code registration will be published in the
830 ftp://ftp.iana.org/in-notes/iana/assignments/ipp/status-codes/
831 area.

832 **18 Contributors**

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

837 Carl Kugler
838 IBM
839 P.O. Box 1900
840 Boulder, CO 80301-9191
841
842 Phone:
843 Fax:
844 e-mail: kugler@us.ibm.com
845

846 Harry Lewis
847 IBM
848 P.O. Box 1900
849 Boulder, CO 80301-9191
850
851 Phone: 303-924-5337
852 FAX:
853 e-mail: harryl@us.ibm.com
854

855 **19 Authors' Addresses**

856

857

Robert Herriot

858

706 Colorado Ave.

859

Palo Alto, CA 94303

860

861

Phone: 650-327-4466

862

Fax: 650-327-4466

863

email: bob@herriot.com

864

865

Thomas N. Hastings

866

Xerox Corporation

867

737 Hawaii St. ESAE 231

868

El Segundo CA 90245

869

870

Phone: 310-333-6413

871

Fax: 310-333-5514

872

email: hastings@cp10.es.xerox.com

873

874

~~Carl Kugler~~

875

~~IBM~~

876

~~P.O. Box 1900~~

877

~~Boulder, CO 80301-9191~~

878

879

~~Phone:~~

880

~~Fax:~~

881

~~e-mail: kugler@us.ibm.com~~

882

883

~~Harry Lewis~~

884

~~IBM~~

885

~~P.O. Box 1900~~

886

~~Boulder, CO 80301-9191~~

887

888

~~Phone: 303-924-5337~~

889

~~FAX:~~

890

~~e-mail: harryl@us.ibm.com~~

891

892

893 IPP Web Page: <http://www.pwg.org/ipp/>894 IPP Mailing List: ipp@pwg.org

895

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

897 1) send it to majordomo@pwg.org

898 2) leave the subject line blank

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

900 subscribe ipp

901 end

902

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

907 20 Description of Base IPP documents

908 The base set of IPP documents includes:

909 Design Goals for an Internet Printing Protocol [RFC2567]

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

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

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

913 Internet Printing Protocol/1.1: Implementer’s Guide [[RFC3196](#)][ipp-ig](#)]

914 Mapping between LPD and IPP Protocols [RFC2569]

915 [Internet Printing Protocol \(IPP\): IPP Event Notifications and Subscriptions](#) [[ipp-ntfy](#)]

916

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

922 The “Rationale for the Structure and Model and Protocol for the Internet Printing Protocol” document
923 describes IPP from a high level view, defines a roadmap for the various documents that form the suite
924 of IPP specification documents, and gives background and rationale for the IETF working group’s
925 major decisions.

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

930 The “Internet Printing Protocol/1.1: Encoding and Transport” document is a formal mapping of the
931 abstract operations and attributes defined in the model document onto HTTP/1.1 [RFC2616]. It

932 defines the encoding rules for a new Internet MIME media type called “application/ipp”. This
933 document also defines the rules for transporting over HTTP a message body whose Content-Type is
934 “application/ipp”. This document defines the ‘ipp’ scheme for identifying IPP printers and jobs.

935 The “Internet Printing Protocol/1.1: Implementer’s Guide” document gives insight and advice to
936 implementers of IPP clients and IPP objects. It is intended to help them understand IPP/1.1 and some
937 of the considerations that may assist them in the design of their client and/or IPP object
938 implementations. For example, a typical order of processing requests is given, including error
939 checking. Motivation for some of the specification decisions is also included.

940 The “Mapping between LPD and IPP Protocols” document gives some advice to implementers of
941 gateways between IPP and LPD (Line Printer Daemon) implementations.

942 ~~The “IPP Event Notifications and Subscriptions” document defines an extension to IPP/1.0 [RFC2566,~~
943 ~~RFC2565] and IPP/1.1 [RFC2911, RFC2910]. This extension allows a client to subscribe to printing~~
944 ~~related Events and defines the semantics for delivering asynchronous *Event Notifications* to the~~
945 ~~specified *Notification Recipient* via a specified *Delivery Method* (i.e., protocols) defined in (separate)~~
946 ~~*Delivery Method* documents.~~

947 21 Full Copyright Statement

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

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

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

959 This document and the information contained herein is provided on an “AS IS” basis and THE
960 INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL
961 WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY
962 WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY
963 RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A
964 PARTICULAR PURPOSE.

965 Acknowledgement

966
967 Funding for the RFC Editor function is currently provided by the Internet Society.