

1 INTERNET-DRAFT 1 ISSUES are highlighted like this.
2 <draft-ietf-ipp-notify-poll-01.txt>

Robert Herriot
Xerox Corp.
Tom Hastings
Xerox Corp.
Carl-Uno Manros
Xerox Corp.
Harry Lewis
IBM, Corp.
May 11, 2000

11 Internet Printing Protocol (IPP):
12 **The 'ipp-get' Notification Polling Method**

13
14 Copyright (C) The Internet Society (2000). All Rights Reserved.

15 Status of this Memo

16 This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of
17 [rfc2026]. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas,
18 and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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

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

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

24 **Abstract**

25 The IPP notification specification [ipp-ntfy] is an OPTIONAL extension to IPP/1.0 and IPP/1.1 that
26 requires the definition of one or more delivery methods for dispatching Event Notification reports to
27 Notification Recipients. This document describes the semantics and syntax of the 'ipp-get' event
28 Notification delivery method. For this delivery method, the client uses an explicit IPP Get-Notifications
29 Printer operation in order to request (pull) Event Notifications from the IPP Printer.

30 When a Printer supports the 'ipp-get' delivery method, it holds each Event Notification for a certain length
31 of time. The amount of time is called the event-lease time. A Printer may assign the same event-lease time
32 to each Event Notification or different times. If a Notification Recipient does not want to miss Event
33 Notifications, the time between consecutive pollings of Subscription objects must be less than the event-
34 lease time of the events that occur between pollings. The Get-Notifications request indicates whether the
35 client wants to receive all pending event Notifications for \ any Subscription with a particular delivery-
36 method URL. With the Get-Notifications operation, the Printer returns all existing Event Notifications
37 along with two time intervals. One specifies the minimum time at which event-leases of future events of the
38 type returned will begin to expire and the other specifies the recommended interval for the client to wait
39 before sending the next Get-Notifications operation. The second time interval is less than the first.

40 The Printer may keep the channel open if the recommended interval is sufficiently short, but in any case the
41 client performs a new Get-Notifications operation each time it wants more Event Notifications. Since the
42 time interval between consecutive client requests is normally less than the event-lease time, consecutive
43 responses will normally contain some Event Notifications that are identical. The youngest ones in the
44 previous response will become the oldest in the next response. The client is expected to filter out these
45 duplicates, which is easy to do because of the sequence number in each Event Notification.

46 The full set of IPP documents includes:

47 Design Goals for an Internet Printing Protocol [RFC2567]

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

49 Internet Printing Protocol/1.1: Model and Semantics [ipp-mod]

50 Internet Printing Protocol/1.1: Encoding and Transport [ipp-pro]

51 Internet Printing Protocol/1.1: Implementer's Guide [ipp-iig]

52 Mapping between LPD and IPP Protocols [RFC2569]

53 Internet Printing Protocol/1.0 & 1.1: Event Notification Specification [ipp-ntfy]

54

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

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

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

68 The "Internet Printing Protocol/1.1: Encoding and Transport" document is a formal mapping of the abstract
69 operations and attributes defined in the model document onto HTTP/1.1 [RFC2616]. It defines the
70 encoding rules for a new Internet MIME media type called "application/ipp". This document also defines
71 the rules for transporting over HTTP a message body whose Content-Type is "application/ipp". This
72 document defines a new scheme named 'ipp-get' for identifying IPP printers and jobs.

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

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

80 The "Event Notification Specification" document defines OPTIONAL operations that allow a client to
81 subscribe to printing related events. Subscriptions include "Per-Job subscriptions" and "Per-Printer
82 subscriptions". Subscriptions are modeled as Subscription objects. Four other operations are defined for
83 subscription objects: get attributes, get subscriptions, renew a subscription, and cancel a subscription.

84

85

Table of Contents

86 1 Introduction 5

87 2 Terminology 5

88 3 Model and Operation 6

89 4 Get-Notifications operation 7

90 4.1 GET-NOTIFICATIONS REQUEST 8

91 4.2 GET-NOTIFICATIONS RESPONSE 8

92 5 Extensions to Print-Job, Print-URI, Create-Job, Create-Printer-Subscription and Create-Printer-Subscription 11

93 5.1 RESPONSE 11

94

95 6 Encoding 12

96 7 IANA Considerations 12

97 8 Internationalization Considerations 12

98 9 Security Considerations 12

99 10 References 13

100 11 Authors' Addresses 13

101 12 Full Copyright Statement 14

102

103

104 **1 Introduction**

105 IPP printers that support the OPTIONAL IPP notification extension [ipp-ntfy] either a) accept, store, and
106 use notification subscriptions to generate Event Notification reports and implement one or more delivery
107 methods for notifying interested parties, or b) support a subset of these tasks and farm out the remaining
108 tasks to a Notification Delivery Service. The 'ipp-get' Event Notification delivery method specified in this
109 document defines a Get-Notifications operation that may be used in a variety of notification scenarios. Its
110 primary intended use is for clients that want to be Notification Recipients. However, the Get-Notifications
111 operation may also be used by Notification Delivery Services for subsequent distribution to the Ultimate
112 Notification Recipients.

113 When a Printer supports the 'ipp-get' delivery method, it holds each Event Notification for a certain length
114 of time. The amount of time is called the event-lease time. A Printer may assign the same event-lease time
115 to each event or different times. If a Notification Recipient does not want to miss Event Notifications, the
116 time between consecutive pollings of Subscription objects must be less than the event-lease time of the
117 Event Notifications that occur between pollings. The Get-Notifications request indicates whether the client
118 wants to receive all pending Event Notifications for any Subscription with a particular notification recipient
119 URL. With the Get-Notifications operation, the Printer returns all existing Event Notifications along with
120 two time intervals. One specifies the minimum time at which event-leases of future events of the type
121 returned will begin to expire and the other specifies the recommended interval for the client to wait before
122 sending the next Get-Notifications operation. The second time interval is less than the first.

123 The Printer may keep the channel open if the recommended interval is sufficiently short, but in any case the
124 client performs a new Get-Notifications operation each time it wants more Notifications. Since the time
125 interval between consecutive client requests is normally less than the event-lease time, consecutive
126 responses will normally contain some events that are identical. The youngest ones in the previous response
127 will become the oldest in the next response. The client is expected to filter out these duplicates, which is
128 easy to do because of the sequence number in each Notification. The reason for not removing the
129 Notifications from the Subscription object with every Get-Notifications request, is so that multiple
130 Notification Recipients can be polling the same subscription object and so the Get-Notification operation
131 satisfies the rule of idempotency. The former is useful if someone is logged in to several desktops at the
132 same time and wants to see the same events at both places. The latter is useful if the network loses the
133 response.

134 **2 Terminology**

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

136 **REQUIRED:** if an implementation supports the extensions described in this document, it **MUST**
137 support a **REQUIRED** feature.

138 **OPTIONAL:** if an implementation supports the extensions described in this document, it **MAY** support
139 an **OPTIONAL** feature.

140 Notification Recipient - See [ipp-ntfy]

141 Subscription object - See [ipp-ntfy]

142 Ultimate Notification Recipient - See [ipp-ntfy]

143 **3 Model and Operation**

144 In the IPP Notification Model [ipp-ntfy], at most one Per-Job Subscription can be supplied in the Job
145 Creation operation. In addition one Per-Job Subscription can be supplied in Create-Job-Subscription
146 operations, and one Per-Printer Subscription can be supplied in the Create-Printer operation. The client that
147 creates these Subscription objects becomes the owner of the Subscription object.

148 When creating each Subscription object, the client supplies the "notify-recipient-uri" (uri) attribute. The
149 "notify-recipient-uri" attribute specifies both a single Notification Recipient that is to receive the
150 Notifications when subsequent events occur and the URL's scheme specifies the method for Notification
151 delivery that the IPP Printer is to use. For the Notification delivery method defined in this document, the
152 scheme of the URL is 'ipp-get' and the host SHOULD be the client host's URL. In addition, the URL MAY
153 contains a path to allow for applications to have a unique URL. Because the Get-Notifications operation
154 uses the "notification-recipient-uri" to specify the events that it wants in the response, the Subscriber can
155 partition events into suitable groups by associating a different URL with each group – the URLs may have
156 the same host but different paths. If a Subscriber wants a friend to receive Event Notification via this
157 delivery method, it can use the friend's URL as the "notification-recipient-uri". When the friend performs
158 the Get-Notifications operation on the URL, it receives all pending the notifications, even those event
159 caused by subscriptions owned by others.

160 For most Notification delivery methods, a Printer sends Event Notifications to the delivery URL and the
161 Printer does not perform any authentication or authorization with the receivers of the Event Notifications.
162 For the Notification delivery method defined in this document, the client requests Event Notifications from
163 the Printer via a Get-Notifications operation, and the Printer performs the same authentication and
164 authorization as it does for the Get-Job-Attributes operation. That is, a Printer MAY allow a client to
165 perform a Get-Notifications operation on any Subscription object or it MAY restrict access as follows. Any
166 client that is authenticated (1) as an operator or administrator or (2) as the owner of the Subscription object
167 can initiate a Get-Notifications operation for that Subscription object.

168 Because a Printer has to wait for a client to request Event Notifications for the 'ipp-get' delivery method,
169 any Printer that supports the 'ipp-get' notification delivery method MUST hold each Event Notification at
170 least for the event-lease time that it advertises to clients. With this rule, a single user can login at different
171 places, say his/her office, the lab, and/or several desktops in the same room, and receive the same Event
172 Notifications from a single Subscription object. In addition, a client that gets no response, perhaps because
173 of a network failure, can perform the Get-Notifications operations two or more times in quick succession
174 and get the same results except for a few newly arrived Event Notifications and a few old Event
175 Notifications whose event-leases have expired.

176 The event-lease time assigned to Event Notifications MAY be different for each implementation.
177 Furthermore, a particular implementation MAY assign different event-lease times to each Event
178 Notification. If a Printer assigns different event-lease times to each Event Notification, the event-lease time

179 returned with Get-Notifications MUST be a value that ensures a client will not miss future Event
180 Notifications.

181 The client issues a Get-Notifications Printer operation in order to initiate the delivery of the pending
182 Notifications held by the Printer for the Subscription objects requested. In this operation, the client
183 specifies the "notification-recipient-uri" attribute and the Printer returns all pending Event Notifications
184 associated with Subscription objects whose "notification-recipient-uri" attribute matches the "notification-
185 recipient-uri" attribute specified in the operation.

186 If the client requests a persistent channel, then the Printer MAY keep the channel open. Either the client or
187 the IPP Printer can disconnect the HTTP connection.

188 **4 Get-Notifications operation**

189 This REQUIRED operation allows the client to request that pending Event Notifications be delivered as a
190 response to this request. The client MUST be the owner or have read-access rights of the Subscription
191 objects that are involved and the delivery method specified when the Subscription objects were created
192 MUST be 'ipp-get'.

193 This operation returns all pending Event Notifications specified by the "notify-recipient-uri" operation
194 attribute. To help a client know when to perform this operation again, the Printer returns both the event-
195 lease time and the suggested-ask-again time in the following operations:

196 a) Job Creation, Create-Printer-Subscription and Create-Job-Subscription operation if the scheme
197 of the "notify-recipient-uri" operation attribute is 'ipp-get'.

198 b) All Get-Notifications operation

199 The client SHOULD perform a Get-Notifications operation at about the suggested-ask-again time and if the
200 Printer receives the Get-Notifications before the event-lease time has elapsed, it MUST have all of the
201 Notifications since the previous Get-Notification operation or the Subscription object creation, whichever
202 was most recent.

203 The IPP Printer MUST accept the request in any state (see [ipp-mod] "printer-state" and "printer-state-
204 reasons" attributes) and MUST remain in the same state with the same "printer-state-reasons".

205 *Access Rights:* The authenticated user (see [ipp-mod] section 8.3) performing this operation must either be
206 the Subscription object owner (as determined when the Subscription object was created by the Job Creation
207 operation, Create-Job-Subscription, or Create-Printer-Subscription operations) or an operator or
208 administrator of the Printer object (see [ipp-mod] Sections 1 and 8.5). Otherwise, the IPP object MUST
209 reject the operation and return: 'client-error-forbidden', 'client-error-not-authenticated', or 'client-error-not-
210 authorized' as appropriate.

211 4.1 Get-Notifications Request

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

213 Group 1: Operation Attributes

214 Natural Language and Character Set:

215 The "attributes-charset" and "attributes-natural-language" attributes as described in [ipp-mod]
216 section 3.1.4.1.

217

218 Target:

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

221

222 Requesting User Name:

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

225

226 "notification-recipient-uri" (url):

227 The client MUST supply this attribute. The Printer object MUST support this attribute. The Printer
228 matches the value of this attribute byte for byte against the value of the "notification-recipient-uri"
229 in each Subscription object in the Printer. If there are no matches, the IPP Printer MUST return the
230 'client-error-not-found' status code. If there are matches, the IPP Printer MUST return all
231 accumulated Event Notifications associated with Subscription objects that contain the matched
232 "notification-recipient-uri" attribute.

233

234 Note: this attribute allows a subscribing client to pick URLs that are unique, e.g. the client's own
235 URL or a friend's URL, which in both cases is likely the URL of the person's host. An application
236 could make a URL unique for each application if it wants. It allows clients who didn't subscribe to
237 get Event Notifications without knowing job-ids or subscription-ids.

238 4.2 Get-Notifications Response

239 The Printer object returns either an immediate error response or a successful response with status code:
240 'successful-ok' when the first event occurs, i.e., when the Printer delivers the first Event Notification.

241 Group 1: Operation Attributes

242 Status Message:

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

246

247 Natural Language and Character Set:

248 The "attributes-charset" and "attributes-natural-language" attributes as described in [ipp-mod]
249 section 3.1.4.2. The Printer uses the values of "notify-charset" and "notify-natural-language",

250 respectively, from one of the Subscription objects associated with the Event Notifications in this
251 response.

252
253 Normally, the values of these attributes is the same in all Subscriptions. If they are not, the Printer
254 picks the values from one Subscription object to put in these attributes. The algorithm for picking
255 the Subscription object is implementation dependent. The choice of natural language is not critical
256 because 'text' and 'name' values can override the operation level natural-language. The Printer's
257 choice of charset is critical because a bad choice may leave it unable to send some 'text' and 'name'
258 values accurately.

259

260 "suggested-ask-again-time-interval" (integer(0:MAX)):

261 The value of this attribute is the suggested number of seconds that SHOULD elapse before the client
262 performs this operation again for these Subscription objects. A client MAY perform this operation
263 at any time, and a Printer MUST respond with all existing Notifications. A client observes this value
264 in order to be a "good network citizen". The value that a Printer returns for this attribute MUST
265 NOT exceed 80% of the "event-lease-time-interval" in order to give a client plenty of time to
266 perform another Get-Notifications operation before the event-lease of the oldest Event Notifications
267 expire.

268

269 "event-lease-time-interval" (integer(0:MAX)):

270 The value of this attribute is the minimum number of seconds until the event-lease expiration time
271 for all future Event Notifications associated with the Subscription objects generating the requested
272 Event Notifications. Thus this number is the maximum number of seconds that elapses before this
273 client SHOULD issue this operation again for these Subscription objects. A Printer MUST preserve
274 all Notifications that occur for the number of seconds specified by this attribute starting at the time
275 it is sent in a response. A client MAY perform this operation at any time, and a Printer MUST
276 respond with all existing Event Notifications. If a Printer receives this operation after this time
277 interval, it MAY have discarded some Notifications since the last response.

278

279 "printer-up-time" (integer(0:MAX)):

280 The value of this attribute is the Printer's "printer-up-time" attribute. Because each Event
281 Notification also contains the value of this attribute when the event occurred, the value of this
282 attribute lets a client know when each Event Notification occurred relative to the time of this
283 response.

284

285

286 Group 2: Unsupported Attributes

287 See [ipp-mod] section 3.1.7 for details on returning Unsupported Attributes.

288

289 If the "subscription-ids" attribute contained subscription-ids that do not exist, the Printer returns
290 them in this group as value of the "subscription-ids" attribute.

291

292 Group 3 through N: Event Notification Attributes

293 The Printer object responds with one Event Notification per Group for each supplied Event
 294 Notification. Each Event Notification MUST meet the criteria specified by the request.(see [ipp-
 295 ntfy]). Each Event Notification Group MUST start with an 'event-notification-attributes-tag', which
 296 is the tag that begins an Event Notification Group (see the section "Encodings of Additional
 297 Attribute Tags" in [ipp-ntfy]).
 298

299 This group includes the following attributes from the section on " Content of Machine Consumable
 300 Event Notifications" in [ipp-ntfy]. They are encoded using the IPP rules for encoding attributes [ipp-
 301 pro] and they may be encoded in any order. Note: the Get-Jobs response acts as a model for
 302 encoding multiple groups of attributes.
 303

304 Table 1 and Table 3 contains the following information

- 305 a) Attribute: the name of the attribute to include from the section on " Content of Machine
 306 Consumable Event Notifications" in [ipp-ntfy].
 307 b) Condition: the condition for the attribute to be present. The value
 308 i) " means that the attribute MUST be present in all Event Notifications.
 309 ii) 'conditional' means the attribute MUST be present if the Printer supports the attribute
 310 iii) 'progress' means the attribute MUST be present for 'job-progress' or 'job-completed'
 311 events only.
 312

313 For a Event Notification for job and printer events, the Printer includes the following attributes.
 314

Table 1 – REQUIRED Attributes in all IPP Event Notification Content

Attribute	Condition
subscription-request-id (integer (0:MAX))	
notify-text (text)	
notify-text-format (mimeMediaType)	
printer-uri (uri)	
trigger-event (type2 keyword)	
printer-up-time (integer(MIN:MAX))	
printer-current-time (dateTime)	conditional

315 For Event Notification for job events, the Printer includes the following additional attributes.
 316

317 **Table 2 – REQUIRED Attributes in all IPP Event Notification Content**

Attribute	Condition
-----------	-----------

Attribute	Condition
job-id (integer(1:MAX))	
job-state (type1 enum)	
job-state-reasons (1setOf type2 keyword)	
job-impressions-completed (integer(0:MAX))	progress

318

319

For Event Notification for printer events, the Printer includes the following additional attributes.

320

Table 3 – REQUIRED Attributes in all IPP Event Notification Content

Attribute	Condition
printer-state (type1 enum)	
printer-state-reasons (1setOf type2 keyword)	
printer-is-accepting-jobs (boolean)	

321

5 Extensions to Print-Job, Print-URI, Create-Job, Create-Printer-Subscription and Create-Printer-Subscription

322

323

5.1 Response

324

When Print-Job, Print-URI or Create-Job contains a "notify-recipient-uri" attribute and the scheme in its value is 'ipp-get', the response contains two additional Operation Attributes that pertain to subscriptions.

325

326

When Create-Job-Subscription or Create-Printer-Subscription operation contains a "notify-recipient-uri" value whose scheme is 'ipp-get', the response contains two additional Operation Attributes that pertain to subscriptions.

327

328

329

Group 1: Operation Attributes

330

" suggested-ask-again-time-interval" (integer(0:MAX)):

331

The value of this attribute is the suggested number of seconds that SHOULD elapse before the client SHOULD perform the Get-Notification operation for the first time with any Subscription objects returned with this job. A client MAY perform the Get-Notification operation at any time, and a Printer MUST respond with all existing Notifications. A client observes this value in order to be a "good network citizen". The value that a Printer returns for this attribute MUST NOT exceed 80% of the "event-lease-time-interval" in order to give a client plenty of time to perform another Get-Notifications operation before the event-lease of the oldest events expire.

332

333

334

335

336

337

338

339

340 "event-lease-time-interval" (integer(0:MAX)):

341 The value of this attribute is the minimum number of seconds until the event-lease expiration time
342 for all future Event Notifications associated with the Subscription objects generating the requested
343 Event Notifications. Thus this number is the maximum number of seconds that elapses before a
344 client SHOULD perform the Get-Notification operation for the first time with any Subscription
345 objects returned with this job. A Printer MUST preserve all Notifications that occur for the number
346 of seconds specified by this attribute starting at the time it is sent in a response. A client MAY
347 perform the Get-Notification operation at any time, and a Printer MUST respond with all existing
348 Event Notifications. If a Printer receives a Get-Notification operation after this time interval, it may
349 have discarded some Notifications since the last response.

350 6 Encoding

351 The operation-id assigned for the Get-Notification operation is:

352 0x00??

353 and should be added to the next version of [ipp-mod] section 4.4.15 "operations-supported".

354 **ISSUE: what is the value?**

355 This notification delivery method uses the IPP transport and encoding [ipp-pro] for the Get-Notifications
356 operation with one extension:

357 notification-attributes-tag = %x07 ; tag of 7

358 7 IANA Considerations

359 There is nothing to register.

360 8 Internationalization Considerations

361 With the 'ipp-get' method defined in this document, the client cannot request the Human Consumable form
362 by supplying the "notify-format" operation attribute (see [ipp-ntfy]). The only supported value for this
363 delivery method is "application/ipp". Therefore, the IPP Printer does not have to perform any localization
364 with this notification delivery method. However, the client when it receives the Get-Notifications response
365 is expected to localize the attributes that have the 'keyword' attribute syntax according to the charset and
366 natural language requested in the Get-Notifications request.

367 9 Security Considerations

368 The IPP Model and Semantics document [ipp-mod] discusses high-level security requirements (Client
369 Authentication, Server Authentication and Operation Privacy). Client Authentication is the mechanism by
370 which the client proves its identity to the server in a secure manner. Server Authentication is the

371 mechanism by which the server proves its identity to the client in a secure manner. Operation Privacy is
372 defined as a mechanism for protecting operations from eavesdropping.

373 Unlike other Event Notification delivery methods in which the IPP Printer initiates the Event Notification,
374 with the method defined in this document, the Notification Recipient is the client who issues the Get-
375 Notifications operation. Therefore, there is no chance of "spam" notifications with this method.
376 Furthermore, such a client can close down the HTTP channel at any time, and so can avoid future unwanted
377 Event Notifications at any time.

378 10 References

379 [ipp-mod]

380 R. deBry, T. Hastings, R. Herriot, S. Isaacson, P. Powell, "Internet Printing Protocol/1.1: Model and
381 Semantics", <draft-ietf-ipp-model-v11-06.txt>, March 1, 2000.

382 [ipp-ntfy]

383 Isaacson, S., Martin, J., deBry, R., Hastings, R. Herriot, T., Shepherd, M., Bergman, R., "Internet
384 Printing Protocol/1.1: IPP Event Notification Specification", <draft-ietf-ipp-not-spec-03.txt>, May
385 11, 2000.

386 [ipp-pro]

387 Herriot, R., Butler, S., Moore, P., Tuner, R., "Internet Printing Protocol/1.1: Encoding and
388 Transport", draft-ietf-ipp-protocol-v11-05.txt, March 1, 2000.

389 [rfc2026]

390 S. Bradner, "The Internet Standards Process -- Revision 3", RFC 2026, October 1996.

391 [RFC2616]

392 R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, "Hypertext
393 Transfer Protocol - HTTP/1.1", RFC 2616, June 1999.

394 11 Authors' Addresses

395

396 Robert Herriot
397 Xerox Corp.
398 3400 Hill View Ave, Building 1
399 Palo Alto, CA 94304

400

401 Phone: 650-813-7696

402 Fax: 650-813-6860

403 e-mail: robert.herriot@pahv.xerox.com

404

405 Tom Hastings

406 Xerox Corporation

407 737 Hawaii St. ESAE 231
408 El Segundo, CA 90245
409
410 Phone: 310-333-6413
411 Fax: 310-333-5514
412 e-mail: hastings@cp10.es.xerox.com
413
414 Carl-Uno Manros
415 Xerox Corporation
416 701 Aviation Blvd.
417 El Segundo, CA 90245
418
419 Phone: 310-333-
420 Fax: 310-333-5514
421 e-mail: manros@cp10.es.xerox.com
422
423 Harry Lewis
424 IBM
425 P.O. Box 1900
426 Boulder, CO 80301-9191
427
428 Phone: (303) 924-5337
429 FAX:
430 e-mail: harryl@us.ibm.com
431

432 **12 Full Copyright Statement**

433 Copyright (C) The Internet Society (2000). All Rights Reserved.

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

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

444 This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET
445 SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES,
446 EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE

447 OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED
448 WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.