

1 Internet Printing Protocol WG
2 INTERNET-DRAFT
3 <draft-ietf-ipp-not-spec-09.txt>
4 Updates RFC 2910 and 2911
5 [Target Category: standards track]
6 Expires: December 27, 2002
7

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

8 Internet Printing Protocol (IPP):
9 **Event Notifications and Subscriptions**

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

12 Status of this Memo

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

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

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

22 **Abstract**

23 This document describes an OPTIONAL extension to the Internet Printing Protocol/1.1: Model and
24 Semantics (RFC 2911, RFC 2910). This extension allows a client to subscribe to printing related
25 Events. Subscriptions are modeled as *Subscription Objects*. The Subscription Object specifies that
26 when one of the specified *Events* occurs, the Printer sends an asynchronous *Event Notification* to the
27 specified *Notification Recipient* via the specified Push or Pull *Delivery Method* (i.e., protocol).

28 A client associates Subscription Objects with a particular Job by performing the Create-Job-
29 Subscriptions operation or by submitting a Job with subscription information. A client associates
30 Subscription Objects with the Printer by performing a Create-Printer-Subscriptions operation. Four
31 other operations are defined for Subscription Objects: Get-Subscriptions-Attributes, Get-Subscriptions,
32 Renew-Subscription, and Cancel-Subscription.
33

33 **Table of Contents**

34	1 Introduction.....	6
35	1.1 Notification Overview	7
36	2 Models for Notification	9
37	2.1 Model for Notification (Simple Case)	9
38	2.2 Model for Notification with Cascading Printers.....	9
39	2.3 Distributed Model for Notification	10
40	2.4 Extended Notification Recipient.....	10
41	3 Terminology.....	10
42	3.1 Conformance Terminology	10
43	3.2 Other Terminology.....	11
44	4 Object Relationships	13
45	4.1 Printer and Per-Printer Subscription Objects.....	13
46	4.2 Printer, Job and Per-Job Subscription Objects.....	13
47	5 Subscription Object.....	14
48	5.1 Rules for Support of Subscription Template Attributes	14
49	5.2 Rules for Processing Subscription Template Attributes	15
50	5.3 Subscription Template Attributes	18
51	5.3.1 notify-recipient-uri (uri).....	19
52	5.3.2 notify-pull-method (type2 keyword).....	20
53	5.3.3 notify-events (1setOf type2 keyword).....	20
54	5.3.3.1 Standard Values for Subscribed Events.....	21
55	5.3.3.1.1 No Events.....	21
56	5.3.3.1.2 Subscribed Printer Events	21
57	5.3.3.1.3 Subscribed Job Events	22
58	5.3.3.2 Rules for Matching of Subscribed Events	23
59	5.3.3.2.1 Rules for Matching of Printer Events	24
60	5.3.3.2.2 Rules for Matching of Job Events.....	24
61	5.3.3.2.3 Special Cases for Matching Rules	24
62	5.3.4 notify-attributes (1setOf type2 keyword).....	25
63	5.3.5 notify-user-data (octetString(63))	26
64	5.3.6 notify-charset (charset)	27
65	5.3.7 notify-natural-language (naturalLanguage).....	27
66	5.3.8 notify-lease-duration (integer(0:67108863)).....	28
67	5.3.9 notify-time-interval (integer(0:MAX))	28
68	5.4 Subscription Description Attributes.....	29
69	5.4.1 notify-subscription-id (integer (1:MAX)).....	30
70	5.4.2 notify-sequence-number (integer (0:MAX)).....	30
71	5.4.3 notify-lease-expiration-time (integer(0:MAX)).....	31
72	5.4.4 notify-printer-up-time (integer(1:MAX)).....	31

73	5.4.5 notify-printer-uri (uri)	32
74	5.4.6 notify-job-id (integer(1:MAX))	32
75	5.4.7 notify-subscriber-user-name (name(MAX))	32
76	6 Printer Description Attributes Related to Notification	33
77	6.1 printer-state-change-time (integer(1:MAX))	33
78	6.2 printer-state-change-date-time (dateTime)	33
79	7 New Values for Existing Printer Description Attributes	34
80	7.1 operations-supported (1setOf type2 enum).....	34
81	8 Attributes Only in Event Notifications	34
82	8.1 notify-subscribed-event (type2 keyword)	34
83	8.2 notify-text (text(MAX)).....	35
84	9 Event Notification Content	35
85	9.1 Content of Machine Consumable Event Notifications	37
86	9.1.1 Event Notification Content Common to All Events	38
87	9.1.2 Additional Event Notification Content for Job Events.....	39
88	9.1.3 Additional Event Notification Content for Printer Events.....	39
89	9.2 Content of Human Consumable Event Notification	39
90	9.2.1 Event Notification Content Common to All Events	40
91	9.2.2 Additional Event Notification Content for Job Events.....	41
92	9.2.3 Additional Event Notification Content for Printer Events.....	42
93	10 Delivery Methods.....	43
94	11 Operations for Notification	44
95	11.1 Subscription Creation Operations	44
96	11.1.1 Create-Job-Subscriptions Operation	45
97	11.1.1.1 Create-Job-Subscriptions Request	45
98	11.1.1.1.1 notify-job-id (integer(1:MAX))	46
99	11.1.1.2 Create-Job-Subscriptions Response.....	46
100	11.1.2 Create-Printer-Subscriptions operation.....	47
101	11.1.2.1 Create-Printer-Subscriptions Request.....	47
102	11.1.2.2 Create-Printer-Subscriptions Response	48
103	11.1.3 Job Creation Operations – Extensions for Notification.....	48
104	11.1.3.1 Job Creation Request	48
105	11.1.3.2 Job Creation Response.....	49
106	11.2 Other Operations	50
107	11.2.1 Restart-Job Operation – Extensions for Notification.....	50
108	11.2.2 Validate-Job Operation – Extensions for Notification	50
109	11.2.3 Get-Printer-Attributes – Extensions for Notification.....	51
110	11.2.4 Get-Subscription-Attributes operation.....	51
111	11.2.4.1 Get-Subscription-Attributes Request.....	52
112	11.2.4.1.1 “notify-subscription-id” (integer (1:MAX))	52

113	11.2.4.1.2 “requested-attributes” (1setOf keyword)	52
114	11.2.4.2 Get-Subscription-Attributes Response	53
115	11.2.5 Get-Subscriptions operation	54
116	11.2.5.1 Get-Subscriptions Request.....	54
117	11.2.5.1.1 “notify-job-id” (integer(1:MAX)).....	55
118	11.2.5.1.2 “limit” (integer(1:MAX)).....	55
119	11.2.5.1.3 “requested-attributes” (1setOf type2 keyword).....	55
120	11.2.5.1.4 “my-subscriptions” (boolean)	55
121	11.2.5.2 Get-Subscriptions Response	55
122	11.2.6 Renew-Subscription operation.....	56
123	11.2.6.1 Renew-Subscription Request.....	57
124	11.2.6.1.1 “notify-subscription-id” (integer (1:MAX))	57
125	11.2.6.1.2 “notify-lease-duration” (integer(0:MAX)).....	57
126	11.2.6.2 Renew-Subscription Response	58
127	11.2.6.2.1 “notify-lease-duration” (integer(0:MAX)).....	58
128	11.2.7 Cancel-Subscription operation.....	58
129	11.2.7.1 Cancel-Subscription Request.....	59
130	11.2.7.1.1 “notify-subscription-id” (integer (1:MAX))	59
131	11.2.7.2 Cancel-Subscription Response	60
132	12 Status Codes.....	60
133	12.1 successful-ok-ignored-subscriptions (0x0003)	60
134	12.2 client-error-ignored-all-subscriptions (0x0414).....	61
135	13 Status Codes in Subscription Attributes Groups.....	61
136	13.1 client-error-uri-scheme-not-supported (0x040C).....	61
137	13.2 client-error-attributes-or-values-not-supported (0x040B)	61
138	13.3 client-error-too-many-subscriptions (0x0415).....	61
139	13.4 successful-ok-too-many-events (0x0005)	62
140	13.5 successful-ok-ignored-or-substituted-attributes (0x0001).....	62
141	14 Encodings of Additional Attribute Tags	62
142	15 Conformance Requirements.....	62
143	15.1 Conformance requirements for clients.....	62
144	15.2 Conformance requirements for Printers.....	62
145	16 Normative References.....	63
146	17 Informative References	64
147	18 Security Considerations	65
148	18.1 Client access rights	65
149	18.2 Printer security threats	66
150	18.3 Notification Recipient security threats.....	67

151	19 IANA Considerations.....	67
152	19.1 Attribute Registrations.....	67
153	19.2 Additional Enum Attribute Value Registrations for the “operations-supported” Printer Attribute.....	68
154	19.3 Operation Registrations.....	69
155	19.4 Status code Registrations.....	69
156	19.5 Attribute Group tag Registrations.....	69
157	19.6 Registration of Events.....	70
158	19.7 Registration of Event Notification Delivery Methods.....	70
159	19.7.1 Requirements for Registration of Event Notification Delivery Methods.....	70
160	19.7.1.1 Required Characteristics.....	70
161	19.7.1.2 Naming Requirements.....	71
162	19.7.1.3 Functionality Requirements.....	71
163	19.7.1.4 Usage and Implementation Requirements.....	71
164	19.7.1.5 Publication Requirements.....	72
165	19.7.2 Registration Procedure.....	72
166	19.7.2.1 Present the proposal to the Community.....	72
167	19.7.2.2 Delivery Method Reviewer.....	72
168	19.7.2.3 IANA Registration.....	72
169	19.7.3 Delivery Method Document Registrations.....	73
170	19.7.4 Registration Template.....	73
171	20 Internationalization Considerations.....	74
172	21 Contributors.....	74
173	22 Author’s Addresses.....	75
174	A. Appendix - Model for Notification with Cascading Printers.....	76
175	B. Appendix - Distributed Model for Notification.....	77
176	C. Appendix - Extended Notification Recipient.....	78
177	D. Appendix - Details about Conformance Terminology.....	79
178	E. Appendix - Object Model for Notification.....	79
179	E.1 Appendix - Object relationships.....	80
180	E.2 Printer Object and Per-Printer Subscription Objects.....	80
181	E.3 Job Object and Per-Job Subscription Objects.....	81
182	F. Appendix - Per-Job versus Per-Printer Subscription Objects.....	81
183	G. Appendix - Description of the base IPP documents.....	81
184	H. Appendix - Full Copyright Statement.....	82

185
186 **Tables**

187	Table 1 – Subscription Template Attributes	19
188	Table 2 – Subscription Description Attributes	30
189	Table 3 – Printer Description Attributes Associated with Notification.....	33
190	Table 4 – Operation-id assignments	34
191	Table 5 – Attributes in Event Notification Content.....	38
192	Table 6 – Additional Event Notification Content for Job Events.....	39
193	Table 7 – Combinations of Events and Subscribed Events for “job-impressions-completed”.....	39
194	Table 8 – Additional Event Notification Content for Printer Events.....	39
195	Table 9 – Printer Name in Event Notification Content.....	41
196	Table 10 – Event Name in Event Notification Content	41
197	Table 11 – Event Time in Event Notification Content	41
198	Table 12 – Job Name in Event Notification Content.....	42
199	Table 13 – Job State in Event Notification Content	42
200	Table 14 – Printer State in Event Notification Content.....	43
201	Table 15 – Information about the Delivery Method	43
202	Table 16 – Printer Conformance Requirements for Operations	63

203

204 Figures

205	Figure 1 – Model for Notification.....	9
206	Figure 2 – Model for Notification with Cascading Printers.....	77
207	Figure 3 – Opaque Use of a Notification Service Transparent to the Client	78
208	Figure 4 – Use of an Extended Notification Recipient transparent to the Printer	79
209	Figure 5 – Object Model for Notification	80

210

211 1 Introduction

212 This IPP notification specification is an OPTIONAL extension to Internet Printing Protocol/1.1: Model
 213 and Semantics [RFC2911, RFC2910]. See Appendix G for a description of the base IPP documents.
 214 This document in combination with the following documents is intended to meet the most important
 215 notification requirements described in [ipp-not-req]:

- 216 Internet Printing Protocol (IPP): “Job Progress Attributes” [ipp-prog]
- 217 Internet Printing Protocol (IPP): “The ‘ippget’ Delivery Method for Event Notifications” [ipp-
 218 get-method]

219

220 This specification REQUIRES that clients and Printers support the ‘ippget’ Pull Delivery Method [ipp-
 221 get-method]. Conforming client and Printer implementations MAY support additional Push or Pull
 222 Delivery Methods as well. Note: this document does not define any Delivery Methods itself, but it
 223 does define the rules for conformance for Delivery Method Documents and their registration with
 224 IANA (see section 19.7.3).

225 Refer to the Table of Contents for the layout of this document.

226 1.1 Notification Overview

227 This document defines operations that a client can perform in order to create *Subscription Objects* in a
228 Printer and carry out other operations on them. A Subscription Object represents a Subscription
229 abstraction. The Subscription Object specifies that when one of the specified *Events* occurs, the Printer
230 sends an asynchronous *Event Notification* to the specified *Notification Recipient* via the specified
231 *Delivery Method* (i.e., protocol).

232 When a client (called a *Subscribing Client*) performs an operation that creates a Subscription Object,
233 the operation contains one or more *Subscription Template Attributes Groups*. Each such group holds
234 information used by the Printer to initialize a newly created Subscription Object. The Printer creates
235 one Subscription Object for each Subscription Template Attributes Group in the operation. This group
236 is like the Job Template Attributes group defined in [RFC2911]. The following is an example of the
237 information included in a Subscription Template Attributes Group (see section 5 for details on the
238 Subscription Object attributes):

- 239 1. The names of Subscribed Events that are of interest to the Notification Recipient.
- 240 2. The address (URL) of one Notification Recipient for a Push Delivery Method or the method for
241 a Pull Delivery Method.
- 242 3. The Delivery Method (i.e., the protocol) which the Printer uses to send the Event Notification.
- 243 4. Some opaque data that the Printer sends to the Notification Recipient in the Event Notification.
244 The Notification Recipient might use this opaque data as a forwarding address for the Event
245 Notification.
- 246 5. The charset to use in text fields within an Event Notification
- 247 6. The natural language to use in the text fields of the Event Notification
- 248 7. The requested lease time in seconds for the Subscription Object

249 An operation that creates a Subscription Object is called a *Subscription Creation Operation*. These
250 operations include the following operations (see section 11.1 for further details):

- 251 - **Job Creation operation:** When a client performs such an operation (Print-Job, Print-URI,
252 and Create-Job), a client can include zero or more Subscription Template Attributes Groups in
253 the request. The Printer creates one Subscription Object for each Subscription Template
254 Attributes Group in the request, and the Printer associates each such Subscription Object with
255 the newly created Job. This document extends these operations' definitions in [RFC2911] by
256 adding Subscription Template Attributes Groups in the request and Subscription Attributes
257 Groups in the response.
- 258 - **Create-Job-Subscriptions operation:** A client can include one or more Subscription
259 Template Attributes Groups in the request. The Printer creates one Subscription Object for
260 each Subscription Template Attributes Group and associates each with the job that is the
261 target of this operation.

- 262 - **Create-Printer-Subscriptions operation:** A client can include one or more Subscription
263 Template Attributes Groups in the request. The Printer creates one Subscription Object for
264 each Subscription Template Attributes Group and associates each with the Printer that is the
265 target of this operation.
- 266 For each of the above operations:
- 267 - the Printer associates a Subscription Object with the Printer or a specific Job. When a
268 Subscription Object is associated with a Job Object, it is called a *Per-Job Subscription Object*.
269 When a Subscription Object is associated with a Printer Object, it is called a *Per-Printer*
270 *Subscription Object*.
- 271 - the response contains one Subscription Attributes Group for each Subscription Template
272 Attributes Group in the request and in the same order. When the Printer successfully creates a
273 Subscription Object, its corresponding Subscription Attributes Group contains the “notify-
274 subscription-id” attribute. This attribute uniquely identifies the Subscription Object and is
275 analogous to a “job-id” for a Job object. Some operations described below use the “notify-
276 subscription-id” to identify the target Subscription Object.
- 277 This document defines the following additional operations (see section 11.2 for further details):
- 278 - **Restart-Job operation:** When a client performs the Restart-Job operation [RFC2911], the
279 Printer re-uses the same Job and its Subscription Objects.
- 280 - **Validate-Job operation:** When a client performs this operation, a client can include zero or
281 more Subscription Template Attributes Groups in the request. The Printer determines if it
282 could create one Subscription Object for each Subscription Template Attributes Group in the
283 request. This document extends this operation’s definition in [RFC2911] by adding
284 Subscription Template Attributes Groups in the request and Subscription Attributes Groups in
285 the response.
- 286 - **Get-Subscription-Attributes operation:** This operation allows a client to obtain the
287 specified attributes of a target Subscription Object.
- 288 - **Get-Subscriptions operation:** This operation allows a client to obtain the specified attributes
289 of all Subscription Objects associated with the Printer or a specified Job.
- 290 - **Renew-Subscription operation:** This operation renews the lease on the target Per-Printer
291 Subscription Object before it expires. A newly created Per-Printer Subscription Object
292 receives an initial lease. It is the duty of the client to use this operation frequently enough to
293 preserve a Per-Printer Subscription Object. The Printer deletes a Per-Printer Subscription
294 Object when its lease expires. A Per-Job Subscription Object last exactly as long as its
295 associated Job Object and thus doesn’t have a lease.
- 296 - **Cancel-Subscription operation:** This operation (1) cancels the lease on the specified Per-
297 Printer Subscription Object and thereby deletes the Per-Printer Subscription Object or (2)
298 deletes the Per-Job Subscription Object.

299 When an Event occurs, the Printer finds all Subscription Objects listening for the Event (see section 9
300 for details on finding such Subscription Objects). For each such Subscription Object, the Printer:

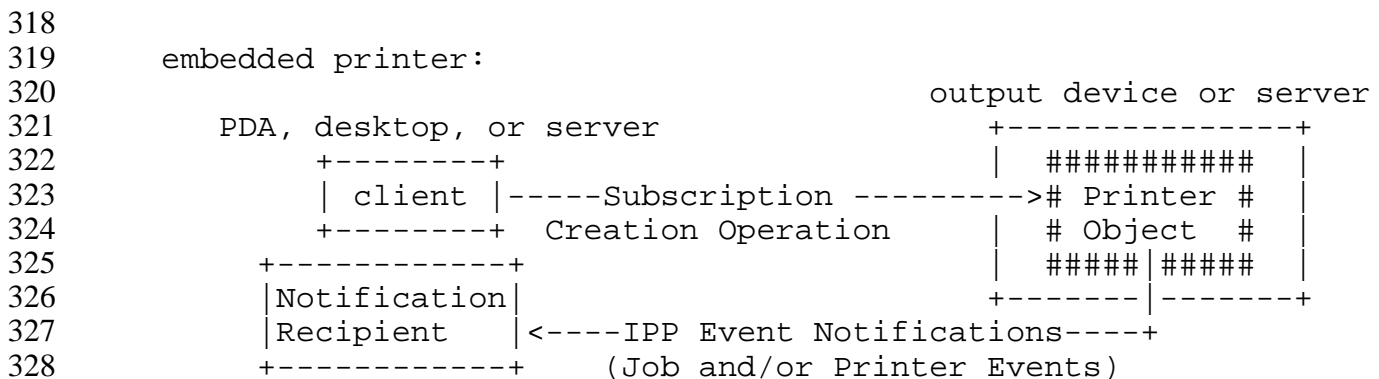
- 301 a) generates an Event Notification with information specified in section 9, AND
302 b) either:
- 303 i) If the Delivery Method is a Push Delivery Method as indicated by the presence of the
304 Subscription Object's "notify-recipient-uri" attribute, delivers the Event Notification
305 using the Delivery Method and target address identified in the Subscription Object's
306 "notify-recipient-uri" attribute, OR
 - 307 ii) If the Delivery Method is a Pull Delivery Method as indicated by the presence of the
308 Subscription Object's "notify-pull-method" attribute, saves Event Notification for a time
309 period called the Event Life defined by the Delivery Method, i.e., the Notification
310 Recipient is expected to fetch the Event Notifications.

311 2 Models for Notification

312 2.1 Model for Notification (Simple Case)

313 As part of a Subscription Creation Operation, an IPP Printer (i.e., located in an output device or a
314 server) creates one or more Subscription Objects. In a Subscription Creation Operation, the client
315 specifies the Notification Recipient to which the Printer is to deliver Event Notifications. A
316 Notification Recipient can be the Subscribing Client or a third party.

317 Figure 1 shows the Notification model for a simple Client-Printer relationship.



329 **Figure 1 – Model for Notification**

330 2.2 Model for Notification with Cascading Printers

331 With this model, there is an intervening Print server between the human user and the Printer in the
332 output device. If the Printer in the output device generates an Event, the system can be configured to
333 send Event Notification either

- 334 - directly to the Notification Recipient specified by the Subscribing Client or
335 - via the Print Server to the Notification Recipient specified by the Subscribing Client.
336 See Appendix A for more details.

337 **2.3 Distributed Model for Notification**

338 The preceding sections (2.1 and 2.2) assume that the Notification software resides in the same device
339 or Server box as the rest of the Printer software. In many implementations, the assumption is correct.
340 However, the Notification model also permits a distributed implementation.

341 For example, the software that supports both Subscription Creation Operations and sending of Event
342 Notifications could be on hardware that is separate from the output device. To make this work, there
343 must be a symbiotic relationship between the output device software and the remote Notification
344 software. Without the remote Notification software, the output device software is not a complete
345 Printer.

346 The term “Printer” in this document includes the software on the output device or server box as well as
347 Notification software that is local to or remote from the output device.

348 Appendix B describes this example in detail.

349 **2.4 Extended Notification Recipient**

350 The model allows for an extended Notification Recipient that is itself a Notification service that
351 forwards each Event Notification to another recipient. The client contacts this Notification Recipient
352 to arrange for forwarding by means outside the scope of this document. The Printer need not be aware
353 that the Notification Recipient forwards Event Notifications.

354 Appendix C describes this example in detail.

355 **3 Terminology**

356 This section defines terminology used throughout this document. Other terminology is defined in
357 [RFC2911].

358 **3.1 Conformance Terminology**

359 Capitalized terms, such as **MUST**, **MUST NOT**, **REQUIRED**, **SHOULD**, **SHOULD NOT**, **MAY**,
360 **NEED NOT**, and **OPTIONAL**, have special meaning relating to conformance as defined in RFC 2119
361 [RFC2119] and [RFC2911] section 12.1. If an implementation supports the extension defined in this
362 document, then these terms apply; otherwise, they do not. These terms define conformance to *this*

363 *document only*; they do not affect conformance to other documents, unless explicitly stated otherwise.
364 See Appendix D for complete details.

365 Note: a feature that is OPTIONAL in this document becomes REQUIRED if the Printer implements a
366 Delivery Method that REQUIRES the feature.

367 **READ-ONLY** – an adjective used in an attribute definition to indicate that an IPP Printer MUST NOT
368 allow the attribute's value to be modified.

369 3.2 Other Terminology

370 This document uses the same terminology as [RFC2911], such as “client”, “Printer”, “attribute”,
371 “attribute value”, “keyword”, “operation”, “request”, “response”, and “support”. In addition, the
372 following terms are defined for use in this document and the Delivery Method Documents:

373 **Administrator** – A human user who establishes policy for and configures the print system.

374 **Operator** – A human user who carries out the policy established by the Administrator and controls the
375 day to day running of the print system.

376 **IPP Client (or client)** – The software component (PDA, desktop, or server) that performs an IPP
377 operation directed at an IPP Printer (located in a server or output device).

378 **Job Creation operation** – One of the operations that creates a Job object: Print-Job, Print-URI and
379 Create-Job. The Restart-Job operation [RFC2911] is not considered a Job Creation operation,
380 since the Printer re-uses the existing Job object. The Validate-Job operation is not considered a
381 Job Creation operation because no Job object is created. Therefore, when a statement also applies
382 to either the Restart-Job and/or the Validate-Job operation, they are mentioned explicitly.

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

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

388 **Compound Event Notification** – two or more Event Notifications that a Printer sends together as a
389 single entity. The Delivery Method Document specifies whether the Delivery Method supports
390 Compound Event Notifications.

391 **Job Event** – an Event caused by some change in a particular job on the Printer, e.g., 'job-completed'.

392 **Printer Event** – an Event caused by some change in the Printer that is not specific to a job, e.g.,
393 'printer-state-changed'.

- 394 **Subscribed Event** – an Event that the Subscribing Client expresses interest in by making it a value of
395 the “notify-events” attribute on a Subscription Object.
- 396 **Subscribed Job Event** – a Subscribed Event that is a Job Event.
- 397 **Subscribed Printer Event** – a Subscribed Event that is a Printer Event.
- 398 **Notification Recipient** – the entity to which the Printer sends an Event Notification.
- 399 **Delivery Method** – the mechanism by which the Printer delivers the Event Notification, e.g., via email
400 or via an Event Notification Delivery Method protocol defined for delivering IPP Event
401 Notifications.
- 402 **Delivery Method Document** – a document, separate from this document, that defines a Delivery
403 Method.
- 404 **Push Delivery Method** –The Printer sends the Event Notification shortly after an Event occurs. For
405 some Push Delivery Methods, the Notification Recipient **MUST** send a response; for others it
406 **MUST NOT** send a response.
- 407 **Pull Delivery Method** – The Printer saves Event Notifications for some event life time and expects the
408 Notification Recipient to request Event Notifications. The Printer returns the Event Notifications
409 in a response to such a request.
- 410 **Event Life** – For a Pull Delivery Method, the length of time in seconds after an Event occurs during
411 which the Printer will return that Event in response to a request for Event Notifications. After
412 the Event Life expires, the Printer will no longer return an Event Notification for that Event in
413 such a response.
- 414 **Subscription Object** – An object containing a set of attributes that indicate: the Notification Recipient,
415 the Delivery Method, the Subscribed Events that cause the Printer to send an Event Notification,
416 and the information to send in an Event Notification.
- 417 **Per-Job Subscription Object** – A Subscription Object that is associated with a single Job. The
418 Create-Job-Subscriptions operation and Job Creation operations create such an object.
- 419 **Per-Printer Subscription Object** – A Subscription Object that is associated with the Printer as a
420 whole. The Create-Printer-Subscriptions operation creates such an object.
- 421 **Subscribing Client** – The client that creates the Subscription Object.
- 422 **Subscription Creation Operation** – An operation that creates a Subscription Object: Job Creation
423 operations, Create-Job-Subscriptions operation, Create-Printer-Subscriptions operation. In the
424 context of a Job Creation operation, a Subscription Creation Operation is the part of the Job
425 Creation operation that creates a Subscription object. The Restart-Job operation [RFC2911] is not
426 considered a Subscription Creation Operation, since the Printer re-uses the Job’s existing
427 Subscription Objects, rather than creating any new Subscription Objects.

- 428 **Subscription Creation Request** – The request portion of a Subscription Creation Operation.
- 429 **Subscription Template Attributes** – Subscription Object attributes that a client can supply in a
430 Subscription Creation Operation and associated Printer Object attributes that specify supported
431 and default values for the Subscription Object attributes.
- 432 **Subscription Description Attributes** – Subscription Object attributes that a Printer supplies during a
433 Subscription Creation Operation.
- 434 **Subscription Template Attributes Group** – The attributes group in a request that contains
435 Subscription Object attributes that are Subscription Template Attributes.
- 436 **Subscription Attributes Group** – The attributes group in a response that contains Subscription Object
437 attributes.
- 438 **Human Consumable Event Notification** – localized text for human consumption only. There is no
439 standardized format and thus programs should not try to parse this text.
- 440 **Machine Consumable Event Notification** – bytes for program consumption. The bytes are formatted
441 according to the Delivery Method document.
- 442 **Printer** – the software that supports an output device or print server (see IPP/1.1 [RFC2911] which
443 uses the terms Printer and Printer object interchangeably). This document extends the IPP/1.1
444 Printer definition to include the software that implements Subscription Creation Operations and
445 the sending of Event Notifications, even if the software for such a Printer would be distributed
446 across a network (see section 2.3).
- 447 **Notification** – when not in the phrases ‘Event Notification’ and ‘Notification Recipient’ — the
448 concepts of this specification, i.e., Events, Subscription Objects, and Event Notifications.

449 **4 Object Relationships**

450 This section defines the object relationships between the Printer, Job, and Subscription Objects. It does
451 not define the implementation. For an illustration of these relationships, see Appendix E.

452 **4.1 Printer and Per-Printer Subscription Objects**

- 453 1. A Printer object can be associated with zero or more Per-Printer Subscription Objects.
- 454 2. Each Per-Printer Subscription Object is associated with exactly one Printer object.

455 **4.2 Printer, Job and Per-Job Subscription Objects**

- 456 1. A Printer object is associated with zero or more Job objects.

- 457 2. Each Job object is associated with exactly one Printer object.
- 458 3. A Job object is associated with zero or more Per-Job Subscription Objects.
- 459 4. Each Per-Job Subscription Object is associated with exactly one Job object.

460 **5 Subscription Object**

461 A Subscribing Client creates a Subscription Object with a Subscription Creation Operation in order to
462 indicate its interest in certain Events. See section 11 for a description of these operations. When an
463 Event occurs, the Subscription Object specifies to the Printer where to send Event Notifications, how
464 to send them and what to put in them. See section 9 for details on the contents of an Event
465 Notification.

466 Using the IPP Job Template attributes as a model (see [RFC2911] section 4.2), the attributes of a
467 Subscription Object are divided into two categories: Subscription Template Attributes and Subscription
468 Description Attributes.

469 Subscription Template attributes are, in turn, like the Job Template attributes, divided into

- 470 1. Subscription Object attributes that a client can supply in a Subscription Creation Request and
- 471 2. their associated Printer Object attributes that specify supported and default values for the
472 Subscription Object attributes

473 The remainder of this section specifies general rules for Subscription Template Attributes and
474 describes each attribute in a Subscription Object.

475 **5.1 Rules for Support of Subscription Template Attributes**

476 Subscription Template Attributes are fundamental to the Notification model described in this
477 specification. The client supplies these attributes in Subscription Creation Operations and the Printer
478 uses these attributes to populate a newly created Subscription Object.

479 Subscription Objects attributes that are Subscription Template Attributes conform to the following
480 rules:

- 481 1. Each attribute's name starts with the prefix string "notify-" and this document calls such
482 attributes "notify-xxx".
- 483 2. For each "notify-xxx" Subscription Object attribute defined in column 1 of Table 1 in section
484 5.3, Table 1 specifies corresponding Printer attributes: "notify-xxx-default", "notify-xxx-
485 supported", "yyy-supported" and "notify-max-xxx-supported" defined in column 2 of Table 1.
486 Note "xxx" stands for the same string in each case and "yyy" stands for some other string.

- 487 3. If a Printer supports “notify-xxx” in column 1 of Table 1, then the Printer MUST support all
488 associated attributes specified in column 2 of Table 1. For example, Table 1 shows that if the
489 Printer supports “notify-events”, it MUST support “notify-events-default”, “notify-events-
490 supported” and “notify-max-events-supported”.
- 491 4. If a Printer does not support “notify-xxx” in column 1 of Table 1, then the Printer MUST NOT
492 support any associated “notify-yyy” attributes specified in column 2 of Table 1. For example,
493 Table 1 shows that if the Printer doesn’t support “notify-events”, it MUST NOT support “notify-
494 events-default”, “notify-events-supported” and “notify-max-events-supported”. Note this rule
495 does not apply to attributes whose names do not start with the string “notify-” and are thus
496 defined in another object and used by other attributes.
- 497 5. Most “notify-xxx” attributes have a corresponding “yyy-supported” attribute that specifies the
498 supported values for “notify-xxx”. Column 2 of Table 1 specifies the name of each “yyy-
499 supported” attribute. The naming rules of IPP/1.1 (see [RFC2911]) are used when “yyy-
500 supported” is “notify-xxx-supported”.
- 501 6. Some “notify-xxx” attributes have a corresponding “notify-xxx-default” attribute that specifies
502 the value for “notify-xxx” if the client does not supply it. Column 2 of Table 1 specifies the
503 name of each “notify-xxx-default” attribute. The naming rules of IPP/1.1 (see [RFC2911]) are
504 used.

505 If a client wishes to present an end user with a list of supported values from which to choose, the client
506 SHOULD query the Printer for its supported value attributes. The client SHOULD also query the
507 default value attributes. If the client then limits selectable values to only those values that are
508 supported, the client can guarantee that the values supplied by the client in the create request all fall
509 within the set of supported values at the Printer. When querying the Printer, the client MAY enumerate
510 each attribute by name in the Get-Printer-Attributes Request, or the client MAY just supply the
511 ‘subscription-template’ group name in order to get the complete set of supported attributes (both
512 supported and default attributes – see section 11.2.3).

513 5.2 Rules for Processing Subscription Template Attributes

514 This section defines a detailed set of rules that a Printer follows when it processes Subscription
515 Template Attributes in a Subscription Creation Request. These rules are similar to the rules for
516 processing Operation attributes in [RFC2911]. That is, the Printer may or may not support an attribute
517 and a client may or may not supply the attribute. Some combinations of these cases are OK. Others
518 return warnings or errors, and perhaps a list of unsupported attributes.

519 A Printer MUST implement the following behavior for processing Subscription Template Attributes in
520 a Subscription Creation Request:

- 521 1. If a client supplies a “notify-xxx” attribute from column 1 of Table 1 and the Printer supports it and
522 its value, the Printer MUST populate the attribute on the created Subscription Object.

- 523 2. If a client supplies a “notify-xxx” attribute from column 1 of Table 1 and the Printer doesn’t
524 support it or its value, the Printer MUST NOT populate the attribute on the created Subscription
525 Object with it. The Printer MUST do one of the following:
- 526 a) If the value of the “notify-xxx” attribute is unsupported, the Printer MUST return the attribute
527 with its value in the Subscription Attributes Group of the response.
- 528 b) If “notify-xxx” is an unsupported attribute, the Printer MUST return the attribute in the
529 Subscription Attributes Group of the response with the ‘unsupported’ out-of-band value.
- 530 Note: The rules of this step are the same as for Unsupported Attributes [RFC2911] section 3.1.7.
531 except that the unsupported attributes are returned in the Subscription Attributes Group rather than
532 the Unsupported Attributes Group because Subscription Creation Operations can create more than
533 one Subscription Object).
- 534 3. If a client is REQUIRED to supply a “notify-xxx” attribute from column 1 of Table 1 and the
535 Printer doesn’t support the supplied value, the Printer MUST NOT create a Subscription Object.
536 The rules for Unsupported Attributes in step #2 still apply.
- 537 4. If a client does not supply a “notify-xxx” attribute from column 1 of Table 1 and the attribute is
538 REQUIRED for the client to supply, the Printer MUST reject the Subscription Creation Operation
539 (including Job Creation operations) without creating a Subscription Object, and MUST return in
540 the response:
- 541 c) the status code ‘client-error-bad-request’ AND
- 542 d) no Subscription Attribute Groups.
- 543 5. If a client does not supply a “notify-xxx” attribute from column 1 of Table 1 that is OPTIONAL for
544 the client to supply, and column 2 of Table 1 either:
- 545 a) specifies a “notify-xxx-default” attribute, the Printer MUST behave as if the client had supplied
546 the “notify-xxx-default” attribute (see step #1) and populate the Subscription object with the
547 value of the “notify-xxx-default” attribute as part of the Subscription Creation operation (unlike
548 Job Template attributes where the Printer does not populate the Job object with defaults – see
549 [RFC2911]) OR
- 550 b) does not specify a “notify-xxx-default” attribute, the Printer MUST populate the “notify-xxx”
551 attribute on the Subscription Object according to the definition of the “notify-xxx” attribute in a
552 section 5.3. For some attributes, the “notify-xxx” is populated with the value of some other
553 attribute, and for others, the “notify-xxx” is NOT populated on the Subscription object at all.
- 554 6. A Printer MUST create a Subscription Object for each Subscription Template Attributes group in a
555 request unless the Printer:
- 556 a) encounters some attributes in a Subscription Template Attributes Group that require the Printer
557 not to create the Subscription Object OR

- 558 b) would create a Per-Job Subscription Object when it doesn't have space for another Per-Job
559 Subscription Object OR
- 560 c) would create a Per-Printer Subscription Object when it doesn't have space for another Per-
561 Printer Subscription Object.
- 562 7. A response **MUST** contain one Subscription Attributes Group for each Subscription Template
563 Attributes Group in the request (and in the same order) whether the Printer creates a Subscription
564 Object from the Subscription Template Attributes Group or not. However, the attributes in each
565 Subscription Attributes Group can be in any order.
- 566 8. The Printer **MUST** populate each Subscription Attributes Group of the response such that each
567 contains:
- 568 a) the "notify-subscription-id" attribute (see section 5.4.1), if and only if the Printer creates a
569 Subscription Object.
- 570 b) the "notify-lease-duration" attribute (see section 5.3.8), if and only if the Printer creates a Per-
571 Printer Subscription Object. The value of this attribute is the value of the Subscription Object's
572 "notify-lease-duration" attribute. This value **MAY** be different from the client-supplied value
573 (see section 5.3.8). If a client supplies this attribute in the creation of a Per-Job Subscription
574 Object, it **MUST** appear in this group with the out-of-band value 'unsupported' to indicate that
575 the Printer doesn't support it in this context.
- 576 c) all of the unsupported Subscription Template Attributes from step #2. Note, they are not
577 returned in the Unsupported Attributes Group in order to separate the unsupported attributes for
578 each Subscription Object.
- 579 d) the "notify-status-code" attribute if the Printer does not create the Subscription Object or if
580 there are unsupported attributes from step #2. The possible values of the "notify-status-code"
581 attribute are shown below (see section 13 for more details). The Printer returns the first value
582 in the list below that describes the status.
- 583 'client-error-uri-scheme-not-supported': the Subscription Object was not created because
584 the scheme of the "notify-recipient-uri" attribute is not supported. See section 13.1 for
585 more details about this status code. See step #3 in this section for the case that causes
586 this error, and the resulting step #6a) that causes the Printer not to create the
587 Subscription Object.
- 588 'client-error-attributes-or-values-not-supported': the Subscription Object was not created
589 because the method of the "notify-pull-method" attribute is not supported. See section
590 13.1 for more details about this status code. See step #3 in this section for the case that
591 causes this error, and the resulting step #6a) that causes the Printer not to create the
592 Subscription Object.
- 593 'client-error-too-many-subscriptions': the Subscription Object was not created because the
594 Printer has no space for additional Subscription Objects. The client **SHOULD** try again

595 later. See section 13.3 for more details about this status code. See steps #6b) and #6c)
596 in this section for the cases that causes this error.

597 ‘successful-ok-too-many-events’: the Subscription Object was created without the “notify-
598 events” values included in this Subscription Attributes Group because the “notify-
599 events” attribute contains too many values. See section 13.4 for more details about this
600 status code. See step #2 in this section and section 5.3.3 for the cases that cause this
601 status code.

602 ‘successful-ok-ignored-or-substituted-attributes’: the Subscription Object was created but
603 some supplied Subscription Template Attributes are unsupported. These unsupported
604 attributes are also in the Subscription Attributes Group. See section 13.5 for more
605 details about this status code. See step #2 in this section for the cases that cause this
606 status code.

607 9. The Printer MUST validate all Subscription Template Attributes and MUST return all unsupported
608 attributes and values in the corresponding Subscription Attributes Group of the response (see step
609 #2) unless it determines that it could not create additional Subscription Objects because of
610 condition #6b) or condition #6c). Then, the Printer NEED NOT validate these additional
611 Subscription Template Attributes and the client MUST NOT expect to find unsupported attributes
612 from step #2 in such additional Subscription Attribute Groups.

613 5.3 Subscription Template Attributes

614 This section contains the Subscription Template Attributes defined for the Subscription and Printer
615 objects.

616 Table 1 below shows the Subscription Template Attributes and has two columns:

- 617 - **Attribute in Subscription Object:** the name and attribute syntax of each Subscription Object
618 Attribute that is a Subscription Template Attribute
- 619 - **Default and Supported Printer Attributes:** the default attribute and supported Printer
620 attributes that are associated with the attribute in column 1.

621 The “notify-recipient-uri” attribute is for use with Push Delivery Methods. The “notify-pull-method”
622 attribute is for use with Pull Delivery Methods.

623 For Push Delivery Methods, a Printer MUST support all attributes in Table 1 below except for “notify-
624 pull-method” and “notify-attributes” (and “notify-pull-method-supported” and “notify-attributes-
625 supported”). For Pull Delivery Methods, a Printer MUST support all attributes in Table 1 below
626 except for “notify-recipient-uri” and “notify-attributes” (and “notify-schemes-supported” and “notify-
627 attributes-supported”). If a Printer supports both Push and Pull Delivery Methods, then it MUST
628 support both “notify-recipient-uri” and “notify-pull-method” attributes.

629 For Pull Delivery Methods, a client MUST supply “notify-recipient-uri” and MAY omit any of the rest
630 of the attributes in column 1 of Table 1 in a Subscription Creation Request. For Push Delivery

631 Methods, a client MUST supply “notify-pull-method” and MAY omit any of the rest of the attributes in
 632 column 1 of Table 1 in a Subscription Creation Request. A client MUST NOT supply both “notify-
 633 recipient-uri” and “notify-pull-method” attributes in the same Subscription Creation Request.

634 Note: The Default and Supported Printer attributes listed in column 2 of Table 1 do not have separate
 635 sections in this specification defining their semantics. Instead, the section for the corresponding
 636 Subscription Object attribute (column 1 of Table 1) contains the semantics of these Printer attributes.
 637 This approach follows the precedence of the Job Template attributes in section 4.2 of [RFC2911]
 638 where the corresponding “xxx-default” and “xxx-supported” Printer attributes are defined in the same
 639 section as the “xxx” Job attribute.

640 **Table 1 – Subscription Template Attributes**

Attribute in Subscription Object	Default and Supported Printer Attributes
notify-recipient-uri (uri) *	notify-schemes-supported (1setOf uriScheme)
notify-pull-method (type2 keyword) **	notify-pull-method-supported (1setOf type2 keyword)
notify-events (1setOf type2 keyword)	notify-events-default (1setOf type2 keyword) notify-events-supported (1setOf type2 keyword) notify-max-events-supported (integer(2:MAX))
notify-attributes (1setOf type2 keyword)	notify-attributes-supported (1setOf type2 keyword)
notify-user-data (octetString(63))	
notify-charset (charset)	charset-supported (1setOf charset)
notify-natural-language (naturalLanguage)	generated-natural-language-supported (1setOf naturalLanguage)
notify-lease-duration (integer(0:MAX))	notify-lease-duration-default (integer(0:67108863)) notify-lease-duration-supported (1setOf (integer(0: 67108863) rangeOfInteger(0:67108863)))
notify-time-interval (integer(0:MAX))	

641 * “notify-recipient-uri” is for Push Delivery Methods only.

642 ** “notify-pull-method” is for Pull Delivery Methods only.

643 5.3.1 notify-recipient-uri (uri)

644 This attribute’s value is a URL, which is a special case of a URI. Its value consists of a scheme and an
 645 address. The address specifies the Notification Recipient and the scheme specifies the Push Delivery
 646 Method for each Event Notification associated with this Subscription Object.

647 If a Printer supports any Push Delivery Methods, a Printer MUST support this attribute and return the
 648 value as supplied by the client (no case conversion or other canonicalization) in any operation response
 649 that includes this attribute.

650 For a Push Delivery Method, a client MUST supply this attribute in a Subscription Creation Operation.
 651 Thus there is no need for a default Printer attribute.

652 The URI scheme of the value of this attribute on a Subscription object MUST be a value of the “notify-
653 schemes-supported (1setOf uriScheme)” Printer attribute. Note: According to [RFC2396] the “:”
654 terminates the scheme and so is not part of the scheme. Therefore, values of the “notify-schemes-
655 supported” Printer attribute do not include the “:” character.

656 If the client supplies an unsupported scheme in the value of this attribute, then the Printer MUST NOT
657 create the Subscription Object and MUST return the “notify-status-code” attribute with the ‘client-
658 error-uri-scheme-not-supported’ value in the Subscription Attributes Group in the response.

659 The Printer MUST treat the address part of this attribute as opaque.

660 **5.3.2 notify-pull-method (type2 keyword)**

661 This attribute’s value is a type2 keyword indicating which Pull Delivery Method is to be used.

662 Since a Printer MUST support the ‘ippget’ Pull Delivery Method [ipp-get-method] (see section 15), a
663 Printer MUST support this attribute and return the value as supplied by the client in any operation
664 response that includes this attribute.

665 For a Pull Delivery Method, a client MUST supply this attribute in a Subscription Creation Operation.
666 Thus there is no need for a default Printer attribute.

667 The keyword value of this attribute on a Subscription object MUST be a value of the “notify-pull-
668 method-supported (1setOf type2 keyword)” Printer attribute.

669 If the client supplies an unsupported method in the value of this attribute, then the Printer MUST NOT
670 create the Subscription Object and MUST return the “notify-status-code” attribute with the ‘client-
671 error-attributes-or-values-not-supported’ value in the Subscription Attributes Group in the response.

672 **5.3.3 notify-events (1setOf type2 keyword)**

673 This attribute contains a set of Subscribed Events. When an Event occurs and it “matches” a value of
674 this attribute, the Printer sends an Event Notification using information in the Subscription Object.
675 The details of “matching” are described subsection 5.3.3.2.

676 A Printer MUST support this attribute.

677 A client MAY supply this attribute in a Subscription Creation Operation. If the client does not supply
678 this attribute in Subscription Creation Operation, the Printer MUST populate this attribute on the
679 Subscription Object with its “notify-events-default” attribute value.

680 Each keyword value of this attribute on a Subscription Object MUST be a value of the “notify-events-
681 supported (1setOf type2 keyword)” Printer attribute.

682 The number of values of this attribute MUST NOT exceed the value of the “notify-max-events-
683 supported” attribute. A Printer MUST support at least 2 values per Subscription Object. If the number

684 of values supplied by a client in a Subscription Creation Operation exceeds the value of this attribute,
685 the Printer MUST treat extra values as unsupported values and MUST use the value of 'successful-ok-
686 too-many-events' for the "notify-status-code" attribute in the Subscription Attributes Group of the
687 response.

688 5.3.3.1 Standard Values for Subscribed Events

689 Each value of this attribute is a keyword and it specifies a Subscribed Event that represents certain
690 changes. Some keywords represent a subset of changes of another keyword, e.g., 'job-completed' is an
691 Event value which is a sub-value of 'job-state-change'. See section 5.3.3.2 for the case where this
692 attribute contains both a value and a sub-value.

693 The values in this section are divided into three categories: No Events, Job Events and Printer Events.

694 A Printer MUST support the Events indicated as "REQUIRED" and MAY support the Events
695 indicated as "OPTIONAL".

696 5.3.3.1.1 No Events

697 The standard and only keyword value for No Events is:

698 **'none'**: REQUIRED – no Event Notifications for any Events. As the sole value of "notify-events-
699 supported", this value means that the Printer does not support the sending of Event Notifications.
700 As the sole value of "notify-events-default", this value means that a client MUST specify the
701 "notify-events" attribute in order for a Subscription Creation Operation to succeed. If the Printer
702 receives this value as the sole value of a Subscription Creation Operation, it does not create a
703 Subscription Object. If a Printer receives this value with other values of a Subscription Creation
704 Operation, the Printer MUST treat this value as an unsupported value.

705 5.3.3.1.2 Subscribed Printer Events

706 The standard keyword values for Subscribed Printer Events are:

707 **'printer-state-changed'**: REQUIRED – the Printer changed state from any state to any other state.
708 Specifically, the value of the Printer's "printer-state", "printer-state-reasons" or "printer-is-
709 accepting-jobs" attributes changed.

710
711 This Subscribed Event value has the following sub-values: 'printer-restarted' and 'printer-
712 shutdown'. A client can listen for any of these sub-values if it doesn't want to listen to all printer-
713 state changes:

714 **'printer-restarted'**: OPTIONAL – when the printer is powered up .

715 **'printer-shutdown'**: OPTIONAL – when the device is being powered down .

716 **‘printer-stopped’**: REQUIRED – when the printer stops printing, i.e. the value of the
717 “printer-state” Printer attribute becomes ‘stopped’.

718 **‘printer-config-changed’**: OPTIONAL – when the configuration of a Printer has changed, i.e., the
719 value of the “printer-message-from-operator” or any “configuration” Printer attribute has changed.
720 A “configuration” Printer attribute is an attribute which can change value because of some human
721 interaction either direct or indirect, and which is not covered by one of the other Events in this
722 section. Examples of “configuration” Printer attributes are any of the Job Template attributes,
723 such as “xxx-supported”, “xxx-ready” and “xxx-default”. The client has to perform a Get-Printer-
724 Attributes to find out the new values of these changed attributes. This Event is useful for GUI
725 clients and drivers to update the available printer capabilities to the user.

726
727 This Event value has the following sub-values: ‘printer-media-changed’ and ‘printer-finishings-
728 changed’. A client can listen for any of these sub-values if it doesn’t want to listen to all printer-
729 configuration changes:

730 **‘printer-media-changed’**: OPTIONAL – when the media loaded on a printer has been
731 changed, i.e., the “media-ready” attribute has changed. This Event includes two cases:
732 an input tray that goes empty and an input tray that receives additional media of the
733 same type or of a different type. The client must check the “media-ready” Printer
734 attribute (see [RFC2911] section 4.2.11) separately to find out what changed.

735 **‘printer-finishings-changed’**: OPTIONAL – when the finisher on a printer has been
736 changed, i.e., the “finishings-ready” attribute has changed. This Event includes two
737 cases: a finisher that goes empty and a finisher that is refilled (even if it is not full). The
738 client must check the “finishings-ready” Printer attribute separately to find out what
739 changed.

740 **‘printer-queue-order-changed’**: OPTIONAL – the order of jobs in the Printer’s queue has changed,
741 so that an application that is monitoring the queue can perform a Get-Jobs operation to determine
742 the new order. This Event does not include when a job enters the queue (the ‘job-created’ Event
743 covers that) and does not include when a job leaves the queue (the ‘job-completed’ Event covers
744 that).

745 5.3.3.1.3 Subscribed Job Events

746 The standard keyword values for Subscribed Job Events are:

747 **‘job-state-changed’**: REQUIRED – the job has changed from any state to any other state.
748 Specifically, the Printer sends this Event whenever the value of the “job-state” attribute or “job-
749 state-reasons” attribute changes. When a Job is removed from the Job Retention or Job History
750 phases (see [RFC2911] section 4.3.7.1), no Event is generated.

751
752 This Event value has the following sub-values: ‘job-created’, ‘job-completed’ and ‘job-stopped’.
753 A client can listen for any of these sub-values if it doesn’t want to listen to all ‘job-state changes’.

754 **‘job-created’**: REQUIRED – the Printer has accepted a Job Creation operation, a Restart-
755 Job operation [RFC2911], or any job operation that creates a Job object from an existing
756 Job object. The Printer populates the job’s “time-at-creation” attribute value (see
757 [RFC2911] section 4.3.14.1). The Printer puts the job in the ‘pending’, ‘pending-held’
758 or ‘processing’ states.

759 **‘job-completed’**: REQUIRED – the job has reached one of the completed states, i.e., the
760 value of the job’s “job-state” attribute has changed to: ‘completed’, ‘aborted’, or
761 ‘canceled’. The Job’s “time-at-completed” and “date-time-at-completed” (if supported)
762 attributes are set (see [RFC2911] section 4.3.14). When a Job completes, a Notification
763 Recipient MAY query the Job using the Get-Job-Attributes operation. To allow such a
764 query, the Printer retains the Job in the Job Retention and/or the Job History phases (see
765 [RFC2911] section 4.3.7.1) for a suitable amount of time that depends on
766 implementation and the Delivery Methods supported. The Printer also sends this Event
767 when a Job is removed with the Purge-Job operation (see [RFC2911] section 3.2.9). In
768 this case, the Event Notification MUST report the ‘job-state’ as ‘canceled’ and the Job
769 object is no longer present for query.

770 **‘job-stopped’**: OPTIONAL – when the job stops printing, i.e. the value of the “job-state”
771 Job attribute becomes ‘processing-stopped’.

772 **‘job-config-changed’**: OPTIONAL – when the configuration of a job has changed, i.e., the value of
773 the “job-message-from-operator” or any of the “configuration” Job attributes have changed. A
774 “configuration” Job attribute is an attribute that can change value because of some human
775 interaction either direct or indirect. Examples of “configuration” Job attributes are any of the job
776 template attributes and the “job-name” attribute. The client performs a Get-Job-Attributes to find
777 out the new values of the changed attributes. This Event is useful for GUI clients and drivers to
778 update the job information to the user.

779 **‘job-progress’**: OPTIONAL – when the Printer has completed Printing a sheet. See the separate [ipp-
780 prog] specification for additional attributes that a Printer MAY send in an Event Notification
781 caused by this Event. The “notify-time-interval” attribute affects this Event by causing the Printer
782 NOT to send an Event Notification every time a ‘job-progress’ Events occurs. See section 5.3.9
783 for full details.

784 **5.3.3.2 Rules for Matching of Subscribed Events**

785 When an Event occurs, the Printer MUST find each Subscription object whose “notify-events”
786 attribute “matches” the Event. The rules for “matching” of Subscribed Events are described separately
787 for Printer Events and for Job Events. This section also describes some special cases.

788 5.3.3.2.1 Rules for Matching of Printer Events

789 Suppose that the Printer causes Printer Event E to occur. For each Per-Job or Per-Printer Subscription
790 S in the Printer, if E equals a value of this attribute in S or E is a sub-value of a value of this attribute in
791 S, the Printer MUST generate an Event Notification.

792 Consider the example. There are three Subscription Objects each with the Subscribed Printer Event
793 'printer-state-changed'. Subscription Object A is a Per-Printer Subscription Object. Subscription
794 Object B is a Per-Job Subscription Object for Job 1, and Subscription Object C is a Per-Job
795 Subscription Object for Job 2. When the Printer enters the 'stopped' state, the Printer sends an Event
796 Notification to the Notification Recipients of Subscription Objects A, B, and C because this is a Printer
797 Event. Note if Job 1 has already completed, the Printer would not send an Event Notification for its
798 Subscription Object, even if Job 1 is retained in the Job Retention and/or the Job History phases (see
799 [RFC2911] section 4.3.7.1).

800 5.3.3.2.2 Rules for Matching of Job Events

801 Suppose that Job J causes Job Event E to occur.

802 1. For each Per-Printer Subscription S in the Printer, if E equals a value of this attribute in S or E is
803 a sub-value of a value of this attribute in S, the Printer MUST generate an Event Notification.

804 2. For each Per-Job Subscription S associated with Job J, if E equals a value of this attribute in S
805 or E is a sub-value of a value of this attribute in S, the Printer MUST generate an Event
806 Notification.

807 3. For each Per-Job Subscription S that is NOT associated Job J, if E equals a value of this
808 attribute in S or E is a sub-value of a value of this attribute in, the Printer MUST NOT generate
809 an Event Notification from S.

810 Consider the example: There are three Subscription Objects listening for the Job Event 'job-
811 completed'. Subscription Object A is a Per-Printer Subscription Object. Subscription Object B is a
812 Per-Job Subscription Object for Job 1, and Subscription Object C is a Per-Job Subscription Object for
813 Job 2. In addition, Per-Printer Subscription Object D is listening for the Job Event 'job-state-changed'.
814 When Job 1 completes, the Printer sends an Event Notification to the Notification Recipient of
815 Subscription Object A (because it is Per-Printer) and Subscription Object B because it is a Per-Job
816 Subscription Object associated with the Job generating the Event. The Printer also sends an Event
817 Notification to the Notification Recipient of Subscription Object D because 'job-completed' is a sub-
818 value of 'job-state-changed' – the value that Subscription Object D is listening for. The Printer does
819 not send an Event Notification to the Notification Recipients of Subscription Object C because it is a
820 Per-Job Subscription Object associated with some Job other than the Job generating the Event.

821 5.3.3.2.3 Special Cases for Matching Rules

822 This section contains rule for special cases.

823 If an Event matches Subscribed Events in two different Subscription Objects and the Printer would
824 send two identical Event Notifications (except for the “notify-subscription-id” attribute) to the same
825 Notification Recipient using the same Delivery Method, the Printer MUST send both Event
826 Notifications. That is, the Printer MUST NOT try to consolidate seemingly identical Event
827 Notifications that occur in separate Subscription objects. Incidentally, the Printer MUST NOT reject
828 Subscription Creation Operations that would create this scenario.

829 If an Event matches two values of this “notify-events” attribute in a single Subscription object (e.g., a
830 value and its sub-value), a Printer MAY send one Event Notification for each matched value in the
831 Subscription Object or it MAY send only one Event Notification per Subscription Object. The rules in
832 sections 5.3.3.2.1 and 5.3.3.2.2 are purposefully ambiguous about the number of Event Notification
833 sent when Event E matches two or more values in a Subscription Object.

834 Consider the example: There are two Per-Printer Subscription Objects when a Job completes.
835 Subscription Object A has the Subscribed Job Event ‘job-state-changed’. Subscription Object B has
836 the Subscribed Job Events ‘job-state-changed’ and ‘job-completed’. The Printer sends an Event
837 Notification to the Notification Recipient of Subscription Object A with the value of ‘job-state-
838 changed’ for the “notify-subscribing-event” attribute. The Printer sends either one or two Event
839 Notifications to the Notification Recipient of Subscription Object B, depending on implementation. If
840 it sends two Event Notifications, one has the value of ‘job-state-changed’ for the “notify-subscribing-
841 event” attribute, and the other has the value of ‘job-completed’ for the “notify-subscribing-event”
842 attribute. If it sends one Event Notification, it has the value of either ‘job-state-changed’ or ‘job-
843 completed’ for the “notify-subscribing-event” attribute, depending on implementation. The algorithm
844 for choosing such a value is implementation dependent.

845 **5.3.4 notify-attributes (1setOf type2 keyword)**

846 This attribute contains a set of attribute names. When a Printer sends a Machine Consumable Event
847 Notification, it includes a fixed set of attributes (see section 9.1). If this attribute is present and the
848 Event Notification is Machine Consumable, the Printer also includes the attributes specified by this
849 attribute.

850 A Printer MAY support this attribute.

851 A client MAY supply this attribute in a Subscription Creation Operation. If the client does not supply
852 this attribute in Subscription Creation Operation or the Printer does not support this attribute, the
853 Subscription Object either (1) MAY contain the “notify-attributes” attribute with a ‘none’ value or (2)
854 NEED NOT contain the attribute at all. There is no “notify-attributes-default” Printer attribute.

855 Each keyword value of this attribute on a Subscription Object MUST be a value of the “notify-
856 attributes-supported (1setOf type2 keyword)” Printer attribute. The “notify-attributes-supported” MAY
857 contain any Printer attribute, Job attribute or Subscription Object attribute that the Printer supports in
858 an Event Notification. It MUST NOT contain any of the attributes in Section 9.1 that a Printer
859 automatically puts in an Event Notification; it would be redundant. If a client supplies an attribute in

860 Section 9.1, the Printer MUST treat it as an unsupported attribute value of the “notify-attributes”
861 attribute.

862 The following rules apply to each keyword value N of the “notify-attributes” attribute: If the value N
863 names:

864 a) a Subscription attribute, the Printer MUST use the attribute N in the Subscription Object that is
865 being used to generate the Event Notification.

866 b) a Job attribute and the Printer is generating an Event Notification from a Per-Job Subscription
867 Object S, the Printer MUST use the attribute N in the Job object associated with S.

868 c) a Job attribute and the Printer is generating an Event Notification from a Per-Printer Subscription
869 Object and the Event is:

870 • a Job Event, the Printer MUST use the attribute N in the Job object that caused the Event.

871 • a Printer Event, the Printer MUST use the attribute N in the active Job.

872 If a Printer supports this attribute and a Subscription Object contains this attribute and the Delivery
873 Method generates a Machine Consumable Event Notification, the Printer MUST include in each Event
874 Notification:

875 a) the attributes specified in section 9.1 and

876 b) each attribute named by this attribute.

877 The Printer MUST NOT use this attribute to generate a Human Consumable Event Notification.

878 **5.3.5 notify-user-data (octetString(63))**

879 This attribute contains opaque data that some Delivery Methods include in each Machine Consumable
880 Event Notification. The opaque data might contain, for example:

881 - the identity of the Subscriber

882 - a path or index to some Subscriber information

883 - a key that identifies to the Notification Recipient the ultimate recipient of the Event
884 Notification

885 - the id for a Notification Recipient that had previously registered with an Instant Messaging
886 Service

887 A Printer MUST support this attribute.

888 A client MAY supply this attribute in a Subscription Creation Operation. If the client does not supply
889 this attribute in the Subscription Creation Operation, the Subscription Object either (1) MAY contain
890 the “notify-user-data” attribute with a zero length value or (2) NEED NOT contain the attribute at all.
891 There is no “notify-user-data-default” Printer attribute.

892 There is no “notify-user-data-supported” Printer attribute. Rather, any octetString whose length does
893 not exceed 63 octets is a supported value. If the length exceeds 63 octets, the Printer MUST treat it as
894 an unsupported value.

895 **5.3.6 notify-charset (charset)**

896 This attribute specifies the charset to be used in the Event Notification content sent to the Notification
897 Recipient, whether the Event Notification content is Machine Consumable or Human Consumable.

898 A Printer MUST support this attribute.

899 A client MAY supply this attribute in a Subscription Creation Operation. If the client does not supply
900 this attribute in Subscription Creation Operation or supplies an unsupported value, the Printer MUST
901 populate this attribute in the Subscription Object with the value of the “attributes-charset” operation
902 attribute, which is a REQUIRED attribute in all IPP requests (see [RFC2911]). If the value of the
903 “attributes-charset” attribute is unsupported, the Printer MUST populate this attribute in the
904 Subscription Object with the value of the Printer’s “charset-configured” attribute. There is no “notify-
905 charset-default” Printer attribute.

906 The value of this attribute on a Subscription Object MUST be a value of the “charset-supported (1setOf
907 charset)” Printer attribute.

908 **5.3.7 notify-natural-language (naturalLanguage)**

909 This attribute specifies the natural language to be used in any human consumable text in the Event
910 Notification content sent to the Notification Recipient, whether the Event Notification content is
911 Machine Consumable or Human Consumable.

912 A Printer MUST support this attribute.

913 A client MAY supply this attribute in a Subscription Creation Operation. If the client does not supply
914 this attribute in Subscription Creation Operation or supplies an unsupported value, the Printer MUST
915 populate this attribute in the Subscription Object with the value of the “attributes-natural-language”
916 operation attribute, which is a REQUIRED attribute in all IPP requests (see [RFC2911]). If the value
917 of the “attributes-natural-language” attribute is unsupported, the Printer MUST populate this attribute
918 in the Subscription Object with the value of the Printer’s “natural-language-configured” attribute.
919 There is no “notify-natural-language-default” Printer attribute.

920 The value of this attribute on a Subscription Object MUST be a value of the “generated-natural-
921 language-supported (1setOf type2 naturalLanguage)” Printer attribute.

922 **5.3.8 notify-lease-duration (integer(0:67108863))**

923 This attribute specifies the duration of the lease (in seconds) associated with the Per-Printer
924 Subscription Object at the time the Subscription Object was created or the lease was renewed. The
925 duration of the lease is infinite if the value is 0, i.e., the lease never expires. See section 5.4.3 on
926 “notify-lease-expiration-time (integer(0:MAX))” for more details.

927 This attribute is not present on a Per-Job Subscription Object because the Subscription Object lasts
928 exactly as long as the associated Job object. See discussion of the ‘job-completed’ event in section
929 5.3.3.1.3 about retention of the Job object after completion.

930 A Printer **MUST** support this attribute.

931 For a Subscription Object Creation operation of a Per-Job Subscription Object, the client **MUST NOT**
932 supply this attribute. If the client does supply this attribute, the Printer **MUST** treat it as an
933 unsupported attribute.

934 For a Subscription Creation Operation of a Per-Printer Subscription Object or a Renew-Subscription
935 operation, a client **MAY** supply this attribute. If the client does not supply this attribute, the Printer
936 **MUST** populate this attribute with its “notify-lease-duration-default” (0:67108863) attribute value. If
937 the client supplies this attribute with an unsupported value, the Printer **MUST** populate this attribute
938 with a supported value, and this value **SHOULD** be as close as possible to the value requested by the
939 client. Note: this rule implies that a Printer doesn’t assign the value of 0 (infinite) unless the client
940 requests it.

941 After the Printer has populated this attribute with a supported value, the value represents the “granted
942 duration” of the lease in seconds and the Printer updates the value of the Subscription Object’s “notify-
943 lease-expiration-time” attribute as specified in section 5.4.3.

944 The value of this attribute on a Subscription Object **MUST** be a value of the “notify-lease-duration-
945 supported” (1setOf (integer(0:67108863) | rangeOfInteger(0:67108863))) Printer attribute.

946 A Printer **MAY** require authentication in order to return the value of 0 (the lease never expires) as one
947 of the values of “notify-lease-duration-supported”, and to allow 0 as a value of the “notify-lease-
948 duration” attribute.

949 Note: The maximum value 67,108,863 is 2 raised to the 26 power minus 1 and is about 2 years in
950 seconds. The value is considerably less than MAX so that there is virtually no chance of an overflow
951 when it is added to “printer-up-time” to produce “notify-lease-expiration-time”.

952 **5.3.9 notify-time-interval (integer(0:MAX))**

953 The ‘job-progress’ Event occurs each time that a Printer completes a sheet. Some Notification
954 Recipients do not want to receive an Event Notification every time this Event occurs. This attribute
955 allows a Subscribing Client to request how often it wants to receive Event Notifications for ‘job-

956 progress' Events. The value of this attribute MAY be any nonnegative integer (0,MAX) indicating the
957 minimum number of seconds between 'job-progress' Event Notifications.

958 The Printer MUST support this attribute if and only if the Printer supports the 'job-progress' Event.

959 A client MAY supply this attribute in a Subscription Creation Operation. If the client does not supply
960 this attribute in the Subscription Creation Operation, the Subscription Object either (1) MAY contain
961 the "notify-time-interval" attribute with a '0' value or (2) NEED NOT contain this attribute at all.
962 There is no "notify-time-interval-default" Printer attribute.

963 There is no "notify-time-interval-supported" Printer attribute.

964 If the 'job-progress' Event occurs and a Subscription Object contains the 'job-progress' Event as a
965 value of the 'notify-events' attribute, there are two cases to consider:

966 1. This attribute is not present on the Subscription Object or has the value of 0. The Printer MUST
967 generate and send an Event Notification (as is the case with other Events).

968 2. This attribute is present with a nonzero value of N:

969 a) If the Printer has not sent an Event Notification for the 'job-progress' Event for the associated
970 Subscription Object within the past N seconds, the Printer MUST send an Event Notification for
971 the Event that just occurred. Note when the Printer completes the first page of a Job, this rule
972 implies that the Printer sends an Event Notification for a Per-Job Subscription Object.

973 b) Otherwise, the Printer MUST NOT generate or send an Event Notification for the associated
974 Subscription Object. The Printer MUST NOT increase the value of the "notify-sequence-
975 number" Subscription Object attribute (i.e., the sequence of values of the "notify-sequence-
976 number" attribute counts the Event Notifications that the Printer sent and not the Events that do
977 not cause an Event Notification to be sent).

978 It is RECOMMENDED that a Subscribing Client use this attribute when it subscribes to the 'job-
979 progress' Event, and that the value be sufficiently large to limit the frequency with which the Printer
980 sends Event Notifications requests.

981 This attribute MUST NOT effect any Events other than 'job-progress'.

982 **5.4 Subscription Description Attributes**

983 Subscription Description Attributes are those attributes that a Printer adds to a Subscription Object at
984 the time of its creation.

985 A Printer MUST support all attributes in this Table 2.

986 A client MUST NOT supply the attributes in Table 2 in a Subscription Template Attributes Group of a
987 Subscription Creation Operation. There are no corresponding default or supported attributes.

988

Table 2 – Subscription Description Attributes

Subscription Object attributes:
notify-subscription-id (integer(1:MAX))
notify-sequence-number (integer(0:MAX))
notify-lease-expiration-time (integer(0:MAX))
notify-printer-up-time (integer(1:MAX))
notify-printer-uri (uri)
notify-job-id (integer(1:MAX))
notify-subscriber-user-name (name(MAX))

989

990 **5.4.1 notify-subscription-id (integer (1:MAX))**

991 This attribute identifies a Subscription Object instance with a number that is unique within the context
992 of the Printer. The Printer generates this value at the time it creates the Subscription Object.

993 A Printer **MUST** support this attribute.

994 The Printer **MAY** assign the value of this attribute sequentially as it creates Subscription Objects.
995 However, if there is no security on Subscription objects, sequential assignment exposes the system to a
996 passive traffic monitoring threat.

997 The Printer **SHOULD** avoid re-using recent values of this attribute during continuous operation of the
998 Printer as well as across power cycles. Then a Subscribing Client is unlikely to find that a stale
999 reference accesses a new Subscription Object.

1000 The 0 value is not permitted in order to allow for compatibility with “job-id” and with SNMP index
1001 values, which also cannot be 0.

1002 **5.4.2 notify-sequence-number (integer (0:MAX))**

1003 The value of this attribute indicates the number of times that the Printer has generated and attempted to
1004 send an Event Notification for this Subscription object. When an Event Notification contains this
1005 attribute, the Notification Recipient can determine whether it missed some Event Notifications (i.e.,
1006 numbers skipped) or received duplicates (i.e., same number twice).

1007 A Printer **MUST** support this attribute.

1008 When the Printer creates a Subscription Object, it **MUST** populate this attribute with a value of 0. This
1009 value indicates that the Printer has not sent any Event Notifications for this Subscription Object.

1010 Each time the Printer sends a newly generated Event Notification, it **MUST** increase the value of this
1011 attribute by 1. For some Delivery Methods, the Printer **MUST** include this attribute in each Event
1012 Notification, and the value **MUST** be the value after it is increased by 1. That is, the value of this

1013 attribute in the first Event Notification after Subscription object creation MUST be 1, the second
1014 MUST be 2, etc. If a Delivery Method is defined such that the Notification Recipient returns a
1015 response, the Printer can re-try sending an Event Notification a certain number of times with the same
1016 sequence number when the Notification Recipient fails to return a response.

1017 If a Subscription Object lasts long enough to reach the value of MAX, its next value MUST be 0, i.e., it
1018 wraps.

1019 **5.4.3 notify-lease-expiration-time (integer(0:MAX))**

1020 This attribute specifies the time in the future when the lease on the Per-Printer Subscription Object will
1021 expire, i.e. the “printer-up-time” value at which the lease will expire. If the value is 0, the lease never
1022 expires.

1023 A Printer MUST support this attribute.

1024 When the Printer creates a Per-Job Subscription Object, this attribute MUST NOT be present – the
1025 Subscription Object lasts exactly as long as the associated Job object. See also the discussion of the
1026 ‘job-completed’ event in section 5.3.3.1.3 about retention of the Job object after completion so that a
1027 Notification Recipient can query the Job object after receiving the ‘job-completed’ Event Notification.

1028 When the Printer creates a Per-Printer Subscription Object, it populates this attribute with a value that
1029 is the sum of the values of the Printer’s “printer-up-time” attribute and the Subscription Object’s
1030 “notify-lease-duration” attribute with the following exception. If the value of the Subscription Object’s
1031 “notify-lease-duration” attribute is 0 (i.e., no expiration time), then the value of this attribute MUST be
1032 set to 0 (i.e., no expiration time).

1033 When the Printer powers up, it MUST populate this attribute in each persistent Subscription Object
1034 with a value using the algorithm in the previous paragraph.

1035 When the “printer-up-time” equals the value of this attribute, the Printer MUST delete the Subscription
1036 Object. A client can extend a lease of a Per-Printer Subscription Object with the Renew-Subscription
1037 operation (see section 11.2.6).

1038 Note: In order to compute the number of seconds remaining in a lease for a Per-Printer Subscription
1039 Object, a client can subtract the Subscription’s “notify-printer-up-time” attribute (see section 5.4.4)
1040 from the Subscription’s “notify-lease-expiration-time” attribute.

1041 **5.4.4 notify-printer-up-time (integer(1:MAX))**

1042 This attribute is an alias for the Printer’s “printer-up-time” attribute “ (see [RFC2911] section 4.4.29).
1043 In other words, when this attribute is queried with the Get-Subscriptions or Get-Subscription-Attributes
1044 operations (see sections 11.2.4 and 11.2.5), the value returned is the current value of the Printer’s
1045 “printer-up-time” attribute, rather than the time at which the Subscription Object was created.

1046 A Printer MUST support this attribute.

1047 When the Printer creates a Per-Job Subscription Object, this attribute MUST NOT be present. When
1048 the Printer creates a Per-Printer Subscription Object, this attribute MUST be present.

1049 Note: this attribute exists in a Per-Printer Subscription Object so that a client using the Get-
1050 Subscription-Attributes or Get-Subscription operations can convert the Per-Printer Subscription's
1051 "notify-lease-expiration-time" attribute to wall clock time with one request. If the value of the "notify-
1052 lease-expiration-time" attribute is not 0 (i.e., no expiration time), then the difference between the
1053 "notify-lease-expiration-time" attribute and the "notify-printer-up-time" is the remaining number of
1054 seconds on the lease from the current time.

1055 **5.4.5 notify-printer-uri (uri)**

1056 This attribute identifies the Printer object that created this Subscription Object.

1057 A Printer MUST support this attribute.

1058 During a Subscription Creation Operation, the Printer MUST populate this attribute with the value of
1059 the "printer-uri" operation attribute in the request. From the Printer URI, the client can, for example,
1060 determine what security scheme was used.

1061 **5.4.6 notify-job-id (integer(1:MAX))**

1062 This attribute specifies whether the containing Subscription Object is a Per-Job or Per-Printer
1063 Subscription Object, and for Per-Job Subscription Objects, it specifies the associated Job.

1064 A Printer MUST support this attribute.

1065 If this attribute is not present, the Subscription Object MUST be a Per-Printer Subscription. If this
1066 attribute is present, the Subscription Object MUST be a Per-Job Subscription Object and this attribute
1067 MUST identify the Job with which the Subscription Object is associated.

1068 Note: This attribute could be useful to a Notification Recipient that receives an Event Notification
1069 generated from a Per-Job Subscription Object and caused by a Printer Event. The Event Notification
1070 gives access to the Printer and the Subscription Object. The Event Notification gives access to the
1071 associated Job only via this attribute. See discussion of the 'job-completed' event in section 5.3.3.1.3
1072 about retention of the Job object after completion so that a Notification Recipient can query the Job
1073 object after receiving the 'job-completed' Event Notification.

1074 **5.4.7 notify-subscriber-user-name (name(MAX))**

1075 This attribute contains the name of the user who performed the Subscription Creation Operation.

1076 A Printer MUST support this attribute.

1077 The Printer **MUST** populates this attribute with the most authenticated printable name that it can obtain
 1078 from the authentication service over which the Subscription Creation Operation was received. The
 1079 Printer uses the same mechanism for determining the value of this attribute as it does for a Job's "job-
 1080 originating-user-name" (see [RFC2911] section 4.3.6).

1081 Note: To help with authentication, a Subscription Object may have additional private attributes about
 1082 the user, e.g., a credential of a principal. Such private attributes are implementation-dependent and not
 1083 defined in this document.

1084 6 Printer Description Attributes Related to Notification

1085 This section defines the Printer Description attributes that are related to Notification. Table 3 lists the
 1086 Printer Description attributes, indicates the Printer support required for conformance, and whether or
 1087 not the attribute is READ-ONLY (see section 3.1):

1088 **Table 3 – Printer Description Attributes Associated with Notification**

Printer object attributes:	REQUIRED	READ-ONLY
printer-state-change-time (integer(1:MAX))	No	Yes
printer-state-change-date-time (dateTime)	No	Yes

1089

1090 6.1 printer-state-change-time (integer(1:MAX))

1091 This **OPTIONAL** attribute records the most recent time at which the 'printer-state-changed' Printer
 1092 Event occurred whether or not any Subscription objects were listening for this event. This attribute
 1093 helps a client or operator to determine how long the Printer has been in its current state.

1094 A Printer **MAY** support this attribute and if so, the attribute **MUST** be READ-ONLY.

1095 On power-up, the Printer **MUST** populate this attribute with the value of its "printer-up-time" attribute,
 1096 so that it always has a value. Whenever the 'printer-state-changed' Printer Event occurs, the Printer
 1097 **MUST** update this attribute with the value of the Printer's "printer-up-time" attribute.

1098 6.2 printer-state-change-date-time (dateTime)

1099 This **OPTIONAL** attribute records the most recent time at which the 'printer-state-changed' Printer
 1100 Event occurred whether or not there were any Subscription Objects listening for this event. This
 1101 attribute helps a client or operator to determine how long the Printer has been in its current state.

1102 A Printer **MAY** support this attribute and if so, the attribute **MUST** be READ-ONLY.

1103 On power-up, the Printer MUST populate this attribute with the value of its “printer-current-time”
 1104 attribute, so that it always has a value (see [RFC2911] section 4.4.30 on “printer-current-time”).
 1105 Whenever the ‘printer-state-changed’ Printer Event occurs, the Printer MUST update this attribute with
 1106 the value of the Printer’s “printer-current-time” attribute.

1107 7 New Values for Existing Printer Description Attributes

1108 This section contains those attributes for which additional values are added.

1109 7.1 operations-supported (1setOf type2 enum)

1110 The following “operation-id” values are added in order to support the new operations defined in this
 1111 document:

1112 **Table 4 – Operation-id assignments**

Value	Operation Name
0x0016	Create-Printer-Subscriptions
0x0017	Create-Job-Subscriptions
0x0018	Get-Subscription-Attributes
0x0019	Get-Subscriptions
0x001A	Renew-Subscription
0x001B	Cancel-Subscription

1113 8 Attributes Only in Event Notifications

1114 This section contains those attributes that exist only in Event Notifications and do not exist in any
 1115 objects.

1116 8.1 notify-subscribed-event (type2 keyword)

1117 This attribute indicates the Subscribed Event that caused the Printer to send this Event Notification.
 1118 This attribute exists only in Event Notifications.

1119 This attribute MUST contain one of the values of the “notify-events” attribute in the Subscription
 1120 Object, i.e., one of the Subscribed Event values. Its value is the Subscribed Event that “matches” the
 1121 Event that caused the Printer to send this Event Notification. This Subscribed Event value may be
 1122 identical to the Event or the Event may be a sub-value of the Subscribed Event. For example, the ‘job-
 1123 completed’ Event (which is a sub-event of the ‘job-state-changed’ event) would cause the Printer to
 1124 send an Event Notification for either the ‘job-completed’ or ‘job-state-changed’ Subscribed Events and
 1125 to send the ‘job-completed’ or ‘job-state-changed’ value for this attribute, respectively,. See section
 1126 5.3.3.2 for the “matching” rules of Subscribed Events and for additional examples.

1127 The Delivery Method Document specifies whether the Printer includes the value of this attribute in an
1128 Event Notification.

1129 **8.2 notify-text (text(MAX))**

1130 This attribute contains a Human Consumable text message (see section 9.2). This message describes
1131 the Event and is encoded as plain text, i.e., 'text/plain' with the charset specified by Subscription
1132 Object's "notify-charset" attribute.

1133 The Delivery Method Document specifies whether the Printer includes this attribute in an Event
1134 Notification.

1135 **9 Event Notification Content**

1136 This section defines the Event Notification content that the Printer sends when an Event occurs.

1137 When an Event occurs, the Printer **MUST** find each Subscription object whose "notify-events"
1138 attribute "matches" the Event. See section 5.3.3.2 for details on "matching". For each matched
1139 Subscription Object, the Printer **MUST** create an Event Notification with the content and format that
1140 the Delivery Method Document specifies. The content contains the value of attributes specified by the
1141 Delivery Method Document. The Printer obtains the values immediately after the Event occurs. For
1142 example, if the "printer-state" attribute changes from 'idle' to 'processing', the Event 'printer-state-
1143 changed' occurs and the Printer puts various attributes into the Event Notification, including "printer-
1144 up-time" and "printer-state" with the values that they have immediately after the Event occurs, i.e., the
1145 value of "printer-state" is 'processing'.

1146 **Event Notification Ordering:**

1147 When a Printer sends Event Notifications, the Event Notifications from any given Subscription Object
1148 **MUST** be in time stamp order, i.e., in order of increasing "printer-up-time" attribute value in the Event
1149 Notification (see Table 5). These Event Notifications **MAY** be interleaved with those from other
1150 Subscription Objects, as long as those others are also in time stamp order. The Printer **MUST** observe
1151 these ordering requirements whether sending multiple pending Events as multiple separate Event
1152 Notifications or together in a single Compound Event Notification.

1153 If a Subscribing Client wants the Printer to send certain Event Notifications in time stamp order, the
1154 Subscribing Client uses a single Subscription Object. Even so, depending on the underlying transport,
1155 the actual order that a Notification Recipient receives separate Event Notifications may differ from the
1156 order sent by the Printer (e.g., email).

1157 Example: Consider two Per-Printer Subscription Objects: SO1 and SO2. SO1 requests 'job-state-
1158 changed' events and SO2 requests 'printer-state-changed' events. The number in parens is the time
1159 stamp. The following Event Notification sequences are the only ones that conform to the ordering
1160 requirements for the Printer to send the Event Notifications:

1161 (a) SO1: 'job-created' (1000), SO1: 'job-stopped' (1005), SO1: 'job-completed' (1009), SO2: 'printer-
1162 stopped' (1005)

1163 (b) SO1: 'job-created' (1000), SO1: 'job-stopped' (1005), SO2: 'printer-stopped' (1005), SO1: 'job-
1164 completed' (1009)

1165 (c) SO1: 'job-created' (1000), SO2: 'printer-stopped' (1005), SO1: 'job-stopped' (1005), SO1: 'job-
1166 completed' (1009)

1167 (d) SO2: 'printer-stopped (1005), SO1: 'job-created' (1000), SO1: 'job-stopped' (1005), SO1: 'job-
1168 completed' (1009)

1169 Examples (b) and (c) are interleaved; examples (a) and (d) are not interleaved and are not appropriate
1170 for some Delivery Methods.

1171 If two different Events occur simultaneously, or nearly so (e.g., "printer-up-time" has the same value
1172 for both), the Printer MUST create a separate Event Notification for each Event, even if the associated
1173 Subscription Object is the same for both Events. However, the Printer MAY combine these distinct
1174 Event Notifications into a single Compound Event Notification if the Delivery Method supports
1175 Compound Event Notifications. For example, suppose that two nearly-simultaneously Events
1176 represent two successive 'printer-state-changed' Events, one from 'idle' to 'processing' and another
1177 from 'processing' to 'stopped'. These two Events have the same name but are different instances of
1178 the Event. Then the Printer MUST create a separate Event Notification for each Event and SHOULD
1179 accurately report the "printer-state" of the first Event as 'processing' and the second Event as
1180 'stopped'.

1181 If a Subscription Object contains more than one Subscribed Event, and several Events occur in quick
1182 succession each matching a different Subscribed Event in the Subscription Object, the Printer MUST
1183 NOT generate a single Event Notification from several of these Events, but MAY combine distinct
1184 Event Notifications into a single Compound Event Notification if the Delivery Method supports
1185 Compound Event Notifications.

1186 After the Printer has created the Event Notification, the Printer delivers it via either a:

1187 Push Delivery Method: The Printer sends the Event Notification shortly after an Event occurs.
1188 For some Push Delivery Methods, the Notification Recipient MUST send a response; for others
1189 it MUST NOT send a response.

1190 Pull Delivery Method: The Printer saves Event Notifications for some Event Life and expects
1191 the Notification Recipient to request Event Notifications. The Printer returns the Event
1192 Notifications in a response to such a request.

1193 If an error that meets the following conditions occurs, the Printer MUST cancel the Subscription
1194 Object.

1195 a) the error occurs during the sending of an Event Notification generated from Subscription Object S
1196 AND

1197 b) the error would continue to occur every time the Printer sends an Event Notification generated from
1198 Subscription Object S in the future.

1199 For example, if the address of the “notify-recipient-uri” of Subscription Object A references a non-
1200 existent target and the Printer determines this fact, it **MUST** delete Subscription Object A.

1201 The next two sections describe the values that a Printer sends in the content of Machine Consumable
1202 and Human Consumable Event Notifications, respectively.

1203 The tables in the sub-sections of this section contain the following columns:

1204 a) **Source Value:** the name of the attribute that supplies the value for the Event Notification.
1205 Asterisks in this field refer to a note below the table.

1206 b) **Sends:** if the Printer supports the value (column 1) on the Source Object (column 3) the
1207 Delivery Method **MUST** specify:

1208 **MUST:** that the Printer **MUST** send the value.

1209 **SHOULD:** either that the Printer **MUST** send the value or that the value is incompatible
1210 with the Delivery Method.

1211 **MAY:** that the Printer **MUST**, **SHOULD**, **MAY**, **MUST NOT**, **SHOULD NOT**, or **NEED**
1212 **NOT** send the value. The Delivery Method specifies the level of conformance for the
1213 Printer.

1214 c) **Source Object:** the object from which the source value comes. If the object is “Event
1215 Notification”, the Printer fabricates the value when it sends the Event Notification. See section
1216 8.

1217 9.1 Content of Machine Consumable Event Notifications

1218 This section defines the attributes that a Delivery Method **MUST** mention in a Delivery Method
1219 Document when specifying the Machine Consumable Event Notification’s contents.

1220 This document does not define the order of attributes in Event Notifications. However, Delivery
1221 Method Documents **MAY** define the order of some or all of the attributes.

1222 A Delivery Method Document **MUST** specify additional attributes (if any) that a Printer
1223 implementation sends in a Machine Consumable Event Notification.

1224 Notification Recipients **MUST** be able to accept Event Notifications containing attributes they do not
1225 recognize. What a Notification Recipient does with an unrecognized attribute is implementation-
1226 dependent. Notification Recipients **MAY** attempt to display unrecognized attributes anyway or **MAY**
1227 ignore them.

1228 The next three sections define the attributes in Event Notification Contents that are:

- 1229 1. for all Events
 1230 2. for Job Events only
 1231 3. for Printer Events only

1232 9.1.1 Event Notification Content Common to All Events

1233 This section lists the attributes that a Delivery Method Document **MUST** specify for all Events.

1234 Table 5 lists potential values in each Event Notification.

1235 **Table 5 – 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(MIN:MAX))	MUST	Printer
printer-current-time (dateTime) *	MUST	Printer
notify-sequence-number (integer (0:MAX))	SHOULD	Subscription
notify-charset (charset)	SHOULD	Subscription
notify-natural-language (naturalLanguage)	SHOULD	Subscription
notify-user-data (octetString(63)) **	SHOULD	Subscription
notify-text (text)	SHOULD	Event Notification
attributes from the “notify-attributes” attribute ***	MAY	Printer
attributes from the “notify-attributes” attribute ***	MAY	Job
attributes from the “notify-attributes” attribute ***	MAY	Subscription

1236
 1237 *A Printer **MUST** send this value only if and only if it supports the Printer’s “printer-current-time”
 1238 attribute.

1239 ** If the Subscription Object does not contain a “notify-user-data” attribute and the Delivery Method
 1240 Document **REQUIRES** the Printer to send the “notify-user-data” source value in the Event
 1241 Notification, the Printer **MUST** send an octet-string of length 0.

1242 *** The last three rows represent additional attributes that a client **MAY** request via the “notify-
 1243 attributes” attribute. A Printer **MAY** support the “notify-attributes” attribute. The Delivery Method
 1244 **MUST** say that the Printer **MUST**, **SHOULD**, **MAY**, **MUST NOT**, **SHOULD NOT**, or **NEED NOT**
 1245 support the “notify-attributes” attribute and specific values of this attribute. The Delivery Method
 1246 **MAY** say that support for the “notify-attributes” is conditioned on support of the attribute by the
 1247 Printer or it **MAY** say that Printer **MUST** support the “notify-attributes” attribute if the Printer supports
 1248 the Delivery Method.

1249 **9.1.2 Additional Event Notification Content for Job Events**

1250 This section lists the additional attributes that a Delivery Method Document MUST specify for Job
1251 Events. See Table 6.

1252 **Table 6 – Additional 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

1253 * The Printer MUST send the “job-impressions-completed” attribute in an Event Notification only for
1254 the combinations of Events and Subscribed Events shown in Table 7.
1255

1256 **Table 7 – 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’

1257

1258 **9.1.3 Additional Event Notification Content for Printer Events**

1259 This section lists the additional attributes that a Delivery Method Document MUST specify for Printer
1260 Events. See Table 8.

1261 **Table 8 – Additional 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

1262

1263 **9.2 Content of Human Consumable Event Notification**

1264 This section defines the information that a Delivery Method MUST mention in a Delivery Method
1265 Document when specifying the Human Consumable Event Notifications contents or the value of the
1266 “notify-text” attribute.

1267 Such a Delivery Method MUST specify the following information and a Printer SHOULD send it:

1268 a) the Printer name (see Table 9)

1269 b) the time of the Event (see Table 11)

1270 c) for Printer Events only:

1271 i) the Event (see Table 10) and/or Printer state information (see Table 14)

1272 d) for Job Events only:

1273 i) the job identity (see Table 12)

1274 ii) the Event (see Table 10) and/or Job state information (see Table 13)

1275

1276 The subsections of this section specify the attributes that a Printer MUST use to obtain this
1277 information.

1278 A Delivery Method Document MUST specify additional information (if any) that a Printer
1279 implementation sends in a Human Consumable Event Notification or in the “notify-text” attribute.

1280 A client MUST NOT request additional attributes via the “notify-attributes” attribute because this
1281 attribute works only for Machine Consumable Event Notifications.

1282 Notification Recipients MUST NOT expect to be able to parse the Human Consumable Event
1283 Notification contents or the value of the “notify-text” attribute.

1284 The next three sections define the attributes in Event Notification Contents that are:

1285 a) for all Events

1286 b) for Job Events only

1287 c) for Printer Events only

1288

1289 **9.2.1 Event Notification Content Common to All Events**

1290 This section lists the source of the information that a Delivery Method MUST specify for all Events.

1291 There is a separate table for each piece of information. Each row in the table represents a source value
1292 for the information and the values are listed in order of preference, with the first one being the
1293 preferred one. An implementation SHOULD use the source value from the earliest row in each table.
1294 It MAY use the source value from another row instead, or it MAY combine the source values from
1295 several rows. An implementation is free to determine the best way to present this information.

1296 In all tables of this section, all rows contain a “MAY” in order to state that the Delivery Method
1297 specifies the conformance.

1298 Table 9 lists the source of the information for the Printer Name. The “printer-name” is more user-
1299 friendly unless the Notification Recipient is in a place where the Printer name is not meaningful. For
1300 example, an implementation could have the intelligence to send the value of the “printer-name”

1301 attribute to a Notification Recipient that can access the Printer via value of the “printer-name” attribute
 1302 and otherwise send the value of the “notify-printer-uri” attribute.

1303 **Table 9 – Printer Name in Event Notification Content**

Source Value	Sends	Source Object
printer-name (name(127))	MAY	Printer
notify-printer-uri (uri)	MAY	Subscription

1304

1305 Table 10 lists the source of the information for the Event name. A Printer MAY combine this
 1306 information with state information described for Jobs in Table 13 or for Printers in Table 14.

1307 **Table 10 – Event Name in Event Notification Content**

Source Value	Sends	Source Object
notify-subscribed-event (type2 keyword)	MAY	Subscription

1308

1309 Table 11 lists the source of the information for the time that the Event occurred. A Printer can send
 1310 this value only if it supports the Printer’s “printer-current-time” attribute. If a Printer does not support
 1311 the “printer-current-time” attribute, it MUST NOT send the “printer-up-time” value instead, since it is
 1312 not an allowed option for human consumable information.

1313 **Table 11 – Event Time in Event Notification Content**

Source Value	Sends	Source Object
printer-current-time (dateTime)	MAY	Printer

1314

1315 9.2.2 Additional Event Notification Content for Job Events

1316 This section lists the source of the additional information that a Delivery Method MUST specify for
 1317 Job Events.

1318 Table 12 lists the source of the information for the job name. The “job-name” is likely more
 1319 meaningful to a user than “job-id”.

1320

Table 12 – Job Name in Event Notification Content

Source Value	Sends	Source Object
job-name (name(MAX))	MAY	Job
job-id (integer(1:MAX))	MAY	Job

1321

1322 Table 13 lists the source of the information for the job state. If a Printer supports the “job-state-
 1323 message” and “job-detailed-state-message” attributes, it SHOULD use those attributes for the job state
 1324 information, otherwise, it should fabricate such information from the “job-state” and “job-state-
 1325 reasons”. For some Events, a Printer MAY combine this information with Event information.

1326

Table 13 – Job State in Event Notification Content

Source Value	Sends	Source Object
job-state-message (text(MAX))	MAY	Job
job-detailed-status-messages (1setOf text(MAX))	MAY	Job
job-state (type1 enum)	MAY	Job
job-state-reasons (1setOf type2 keyword)	MAY	Job

1327

1328 9.2.3 Additional Event Notification Content for Printer Events

1329 This section lists the source of the additional information that a Delivery Method MUST specify for
 1330 Printer Events.

1331 Table 14 lists the source of the information for the printer state. If a Printer supports the “printer-state-
 1332 message”, it SHOULD use that attribute for the job state information, otherwise it SHOULD fabricate
 1333 such information from the “printer-state” and “printer-state-reasons”. For some Events, a Printer MAY
 1334 combine this information with Event information.

1335

Table 14 – Printer State in Event Notification Content

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

1336 **10 Delivery Methods**

1337 A Delivery Method is the mechanism, i.e., protocol, by which the Printer delivers an Event Notification
 1338 to a Notification Recipient. There are several potential Delivery Methods for Event Notifications,
 1339 standardized, as well as proprietary. This specification REQUIRES that the ‘ippget’ Pull Delivery
 1340 Method [ipp-get-method] be supported. Conforming implementations MAY support additional Push
 1341 or Pull Delivery Methods as well. This document does not define any of these delivery mechanisms.
 1342 Each Delivery Method MUST be defined in a Delivery Method Document that is separate from this
 1343 document. New Delivery Methods will be created as needed using an extension to the registration
 1344 procedures defined in [RFC2911]. Such documents are registered with IANA (see section 19.7.3).

1345 The following sorts of Delivery Methods are possible:

- 1346 – The Notification Recipient polls for Event Notifications at intervals directed by the Printer
- 1347 – The Printer sends Event Notifications to the Notification Recipient using http as the transport.
- 1348 – The Printer sends an email message.

1349 This section specifies how to define a Delivery Method Document and what to put in such a document.

1350 A Delivery Method Document MUST contain an exact copy of the following paragraph, caption and
 1351 table. In addition, column 2 of the table in the Delivery Method Document MUST contain answers to
 1352 questions in column 1 for the Delivery Method. Also, the Delivery Method document MUST contain a
 1353 reference to this document and call that reference [ipp-ntfy] because the table contains an [ipp-ntfy]
 1354 reference.

1355 If a Printer supports this Delivery Method, the following are its characteristics.

1356

Table 15 – 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?	
2. Is the Delivery Method REQUIRED, RECOMMENDED, or	

OPTIONAL for an IPP Printer to support?	
3. What transport and delivery protocols does the Printer use to deliver the Event Notification Content, i.e., what is the entire network stack?	
4. Can several Event Notifications be combined into a Compound Event Notification?	
5. Is the Delivery Method initiated by the Notification Recipient (pull), or by the Printer (push)?	
6. Is the Event Notification content Machine Consumable or Human 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?	
8. What are the latency and reliability of the transport and delivery protocol?	
9. What are the security aspects of the transport and delivery protocol, e.g., how it is handled in firewalls?	
10. What are the content length restrictions?	
11. What are the additional values or pieces of information that a Printer sends in an Event Notification content and the conformance requirements thereof?	
12. What are the additional Subscription Template and/or Subscription Description attributes and the conformance requirements thereof?	
13. What are the additional Printer Description attributes and the conformance requirements thereof?	

1357

1358 **11 Operations for Notification**

1359 This section defines all of the operations for Notification. Section 7.1 assigns the “operation-id” for
 1360 each operation. The following two sub-sections define Subscription Creation Operations, and other
 1361 operations.

1362 **11.1 Subscription Creation Operations**

1363 This section defines the Subscription Creation Operations. The first section on Create-Job-
 1364 Subscriptions gives most of the information. The other Subscription Creation Operations refer to the

1365 section on Create-Job-Subscriptions, even though the Create-Job-Subscriptions operation is the only
1366 OPTIONAL operation in this document (see section 12).

1367 A Printer MUST support Create-Printer-Subscriptions and the Subscription Template Attributes Group
1368 in Job Creation operations. It MAY support Create-Job-Subscriptions operations.

1369 **11.1.1 Create-Job-Subscriptions Operation**

1370 The operation creates one or more Per-Job Subscription Objects. The client supplies one or more
1371 Subscription Template Attributes Groups each containing one or more of Subscription Template
1372 Attributes (defined in section 5.3).

1373 Except for errors, the Printer MUST create exactly one Per-Job Subscription Object from each
1374 Subscription Template Attributes Group in the request, even if the newly created Subscription Object
1375 would have identical behavior to some existing Subscription Object. The Printer MUST associate each
1376 newly created Per-Job Subscription Object with the target Job, which is specified by the “notify-job-id”
1377 operation attribute.

1378 The Printer MUST accept the request in any of the target job’s ‘not-completed’ states, i.e., ‘pending’,
1379 ‘pending-held’, ‘processing’, or ‘processing-stopped’. The Printer MUST NOT change the job’s “job-
1380 state” attribute because of this operation. If the target job is in any of the ‘completed’ states, i.e.,
1381 ‘completed’, ‘canceled’, or ‘aborted’, then the Printer MUST reject the request and return the ‘client-
1382 error-not-possible’ status code; the response MUST NOT contain any Subscription Attribute Groups.

1383 Access Rights: To create Per-Job Subscription Objects, the authenticated user (see [RFC2911] section
1384 8.3) performing this operation MUST (1) be the job owner, (2) have Operator or Administrator access
1385 rights for this Printer (see [RFC2911] sections 1 and 8.5), or (3) be otherwise authorized by the
1386 Printer’s administrator-configured security policy to create Per-Job Subscription Objects for the target
1387 job. Otherwise the Printer MUST reject the operation and return: the ‘client-error-forbidden’, ‘client-
1388 error-not-authenticated’, or ‘client-error-not-authorized’ status code as appropriate.

1389 **11.1.1.1 Create-Job-Subscriptions Request**

1390 The following groups of attributes are part of the Create-Job-Subscriptions Request:

1391 Group 1: Operation Attributes

1392 Natural Language and Character Set:

1393 The “attributes-charset” and “attributes-natural-language” attributes as described in
1394 [RFC2911] section 3.1.4.1.

1395

1396 Target:

1397 The “printer-uri” attribute which defines the target for this operation as described in
1398 [RFC2911] section 3.1.5.

1399

1400 Requesting User Name:
1401 The "requesting-user-name" attribute SHOULD be supplied by the client as described in
1402 [RFC2911] section 8.3.

1403 **11.1.1.1 notify-job-id (integer(1:MAX))**

1404 The client MUST supply this attribute and it MUST specify the Job object to associate the
1405 Per-Job Subscription with. The value of "notify-job-id" MUST be the value of the "job-id" of
1406 the associated Job object. If the client does not supply this attribute, the Printer MUST reject
1407 this request with a 'client-error-bad-request' status code.
1408

1409 Group 2-N: Subscription Template Attributes

1410 For each occurrence of this group:

1411
1412 The client MUST supply one or more Subscription Template Attributes in any order. See
1413 section 5.3 for a description of each such attribute. See section 5.2 for details on processing
1414 these attributes.

1415 **11.1.1.2 Create-Job-Subscriptions Response**

1416 The Printer MUST return to the client the following sets of attributes as part of a Create-Job-
1417 Subscriptions response:

1418 Group 1: Operation Attributes

1419 Status Message:

1420 In addition to the REQUIRED status code returned in every response, the response
1421 OPTIONALLY includes a "status-message" (text(255)) and/or a "detailed-status-message"
1422 (text(MAX)) operation attribute as described in [RFC2911] sections 13 and 3.1.6.
1423

1424 In this group, the Printer can return any status codes defined in [RFC2911] and section 12.
1425 The following is a description of the important status codes:
1426

1427 **successful-ok:** the Printer created all Subscription Objects requested (see [RFC2911]).

1428 **successful-ok-ignored-subscriptions:** the Printer created some Subscription Objects
1429 requested but some failed. The Subscription Attributes Groups with a "notify-status-
1430 code" attribute are the ones that failed (see section 12.1).

1431 **client-error-ignored-all-subscriptions:** the Printer created no Subscription Objects
1432 requested and all failed. The Subscription Attributes Groups with a "notify-status-
1433 code" attribute are the ones that failed (see section 12.2).

1434 **client-error-not-possible:** For this operation and other Per-Job Subscription operations,
1435 this error can occur because the specified Job has already completed (see [RFC2911]),
1436 whether or not the Job is retained in the Job Retention and/or Job History phases (see
1437 [RFC2911] section 4.3.7.1).

1438

1439

Natural Language and Character Set:

1440

The “attributes-charset” and “attributes-natural-language” attributes as described in

1441

[RFC2911] section 3.1.4.2.

1442

1443

Group 2: Unsupported Attributes

1444

See [RFC2911] section 3.1.7 for details on returning Unsupported Attributes. This group does not contain any unsupported Subscription Template Attributes; they are returned in the Subscription Attributes Group (see below).

1445

1446

1447

1448

Group 3-N: Subscription Attributes

1449

These groups MUST be returned unless the Printer is unable to interpret the entire request, e.g., the “status-code” parameter returned in Group 1 has the value: ‘client-error-bad-request’.

1450

1451

1452

“notify-status-code” (type2 enum):

1453

Indicates the status of this subscription (see section 13 for the status code definitions).

1454

Section 5.2 defines when this attribute MUST be present in this group.

1455

1456

See section 5.2 for details on the contents of each occurrence of this group.

1457

1458

11.1.2 Create-Printer-Subscriptions operation

1459

The operation is identical to Create-Job-Subscriptions with exceptions noted in this section.

1460

The operation creates Per-Printer Subscription Objects instead of Per-Job Subscription Objects, and associates each newly created Per-Printer Subscription Object with the Printer specified by the operation target rather than with a specific Job.

1461

1462

1463

The Printer MUST accept the request in any of its states, i.e., ‘idle’, ‘processing’, or ‘stopped’. The Printer MUST NOT change its “printer-state” attribute because of this operation.

1464

1465

Access Rights: To create Per-Printer Subscription Objects, the authenticated user (see [RFC2911] section 8.3) performing this operation MUST have (1) Operator or Administrator access rights for this Printer (see [RFC2911] sections 1 and 8.5), or (2) be otherwise authorized by the Printer’s administrator-configured security policy to create Per-Printer Subscription Objects for this Printer. Otherwise, the Printer MUST reject the operation and return: the ‘client-error-forbidden’, ‘client-error-not-authenticated’, or ‘client-error-not-authorized’ status code as appropriate.

1466

1467

1468

1469

1470

1471

11.1.2.1 Create-Printer-Subscriptions Request

1472

The groups are identical to the Create-Job-Subscriptions (see section 11.1.1.1) except that the

1473

Operation Attributes group MUST NOT contain the “notify-job-id” attribute. If the client does supply

1474 the “notify-job-id” attribute, then the Printer MUST treat it as any other unsupported Operation
1475 attribute and MUST return it in the Unsupported Attributes group.

1476 **11.1.2.2 Create-Printer-Subscriptions Response**

1477 The groups are identical to the Create-Job-Subscriptions (see section 11.1.1.2).

1478 **11.1.3 Job Creation Operations – Extensions for Notification**

1479 This document extends the Job Creation operations (see section 3.2) to create Subscription Objects as a
1480 part of the operation.

1481 The Job Creation operations are identical to Create-Job-Subscriptions operation with exceptions noted
1482 in this section.

1483 Unlike the Create-Job-Subscriptions operation, a Job Creation operation associates the newly created
1484 Subscription Objects with the Job object created by this operation. The operation succeeds if and only
1485 if the Job creation succeeds. If the Printer does not create some or all of the requested Subscription
1486 Objects, the Printer MUST return a ‘successful-ok-ignored-subscriptions’ status-code instead of a
1487 ‘successful-ok’ status-code, but the Printer MUST NOT reject the operation because of a failure to
1488 create Subscription Objects.

1489 If the Job Creation operation includes a Job Template group, the client MUST supply it after the
1490 Operation Attributes group and before the first Subscription Template Attributes Group.

1491 If a Printer does not support this Notification specification, then it MUST treat the Subscription
1492 Attributes Group like an unknown group and ignore it (see [RFC2911] section 5.2.2). Because the
1493 Printer ignores the Subscription Attributes Group, it doesn’t return them in the response either, thus
1494 indicating to the client that the Printer doesn’t support Notification.

1495 After completion of a successful Job Creation operation, the Printer generates a ‘job-created’ event (see
1496 section 5.3.3.1.3).

1497 Access Rights: To create Per-Job Subscription Objects, the authenticated user (see [RFC2911] section
1498 8.3) performing this operation MUST either have permission to create Jobs on the Printer or have
1499 Operator or Administrator access rights for this Printer (see [RFC2911] sections 1 and 8.5). Otherwise
1500 the Printer MUST reject the operation and return: the ‘client-error-forbidden’, ‘client-error-not-
1501 authenticated’, or ‘client-error-not-authorized’ status code as appropriate.

1502 **11.1.3.1 Job Creation Request**

1503 The groups for this operation are sufficiently different from the Create-Job-Subscriptions operation that
1504 they are all presented here. The following groups of attributes are supplied as part of a Job Creation
1505 Request:

1506 Group 1: Operation Attributes

1507 Same as defined in [RFC2911] for Print-Job, Print-URI, and Create-Job requests.

1508

1509 Group 2: Job Template Attributes

1510 The client **OPTIONALLY** supplies a set of Job Template attributes as defined in [RFC2911]
1511 section 4.2.

1512

1513 Group 3 to N: Subscription Template Attributes

1514 The same as Group 2-N in Create-Job-Subscriptions. See section 11.1.1.1.

1515 Group N+1: Document Content (Print-Job only)

1516 The client **MUST** supply the document data to be processed.

1517

1518 11.1.3.2 Job Creation Response

1519 The Printer **MUST** return to the client the following sets of attributes as part of a Print-Job, Print-URI,
1520 and Create-Job Response:

1521 Group 1: Operation Attributes

1522 Status Message:

1523

1524 As defined in [RFC2911] for Print-Job, Print-URI, and Create-Job requests.

1525

1526 In this group, the Printer can return any status codes defined in [RFC2911] and section 12.
1527 The following is a description of the important status codes:

1528

1529 **successful-ok:** the Printer created the Job and all Subscription Objects requested (see
1530 [RFC2911]).

1531 **successful-ok-ignored-subscriptions:** the Printer created the Job and not all of the
1532 Subscription Objects requested (see section 12.1). This status-code hides
1533 ‘successful-ok-xxx’ status-codes that could reveal problems in Job creation. The
1534 Printer **MUST NOT** return the ‘client-error-ignored-all-subscriptions’ status code for
1535 Job Creation operations because the Printer returns an error status-code only when it
1536 fails to create a Job.

1537

1538 Natural Language and Character Set:

1539 The “attributes-charset” and “attributes-natural-language” attributes as described in
1540 [RFC2911] section 3.1.4.2.

1541

1542 Group 2: Unsupported Attributes

1543 See [RFC2911] section 3.1.7 for details on returning Unsupported Attributes. This group
1544 does not contain any unsupported Subscription Template Attributes; they are returned in the
1545 Subscription Attributes Group (see below).
1546

1547 Group 3: Job Object Attributes

1548 The “job-id” of the Job Object just created, etc., as defined in [RFC2911] for Print-Job, Print-
1549 URI, and Create-Job requests.
1550

1551 Group 4 to N: Subscription Attributes

1552 These groups MUST be returned if and only if the client supplied Subscription Template
1553 Attributes and the operation was accepted.

1554 See section 5.2 for details on the contents of each occurrence of this group.
1555

1556 11.2 Other Operations

1557 This section defines other operations on Subscription objects.

1558 11.2.1 Restart-Job Operation – Extensions for Notification

1559 The Restart-Job operation [RFC2911] is neither a Job Creation operation nor a Subscription Creation
1560 operation (see section 3.2). For the Restart-Job operation, the client MUST NOT supply any Job
1561 Subscription Attributes Groups. The Printer MUST treat any supplied Job Subscription Attributes as
1562 unsupported attributes.

1563 For this operation, the Printer does not return a job-id or any Subscription Attributes groups because
1564 the Printer reuses the existing Job object with the same job-id and the existing Per-Job Subscription
1565 Objects with the same subscription-ids. However, after successful completion of this operation, the
1566 Printer generates a ‘job-created’ event (see section 5.3.3.1.3).

1567 11.2.2 Validate-Job Operation – Extensions for Notification

1568 A client can test whether one or more Subscription Objects could be created using the Validate-Job
1569 operation. The client supplies one or more Subscription Template Attributes Groups (defined in
1570 section 5.3), just as in a Job Creation request.

1571 A Printer MUST support this extension to this operation.

1572 The Printer MUST accept requests that are identical to the Job Creation request defined in section
1573 11.1.3.1, except that the request MUST NOT contain document data.

1574 The Printer MUST return the same groups and attributes as the Print-Job operation (section 11.1.3.1)
1575 with the following exceptions. The Printer MUST NOT return a Job Object Attributes Group because

1576 no Job is created. The Printer MUST NOT return the “notify-subscription-id” attribute in any
1577 Subscription Attribute Group because no Subscription Object is created.

1578 If the Printer would succeed in creating a Subscription Object, the corresponding Subscription
1579 Attributes Group either has no ‘status-code’ attribute or a ‘status-code’ attribute with a value of
1580 ‘successful-ok-too-many-events’ or ‘successful-ok-ignored-or-substituted-attributes’ (see sections 5.2
1581 and 13). The status-codes have the same meaning as in Job Creation except the results state what
1582 “would happen”.

1583 The Printer MUST validate Subscription Template Attributes Groups in the same manner as the Job
1584 Creation operations.

1585 **11.2.3 Get-Printer-Attributes – Extensions for Notification**

1586 This operation is extended so that it returns Printer attributes defined in this document.

1587 A Printer MUST support this extension to this operation.

1588 In addition to the requirements of [RFC2911] section 3.2.5, a Printer MUST support the following
1589 additional values for the “requested-attributes” Operation attribute in this operation and return such
1590 attributes in the Printer Object Attributes group of its response.

- 1591 1. **Subscription Template Attributes:** Each supported attribute in column 2 of Table 1.
- 1592 2. **New Printer Description Attributes:** Each supported attribute in section 6.
- 1593 3. **New Group Name:** The ‘subscription-template’ group name, which names all supported
1594 Subscription Template Attribute in column 2 of Table 1. This group name is also used in the
1595 Get-Subscription-Attributes and Get-Subscriptions operation with an analogous meaning.
- 1596 4. **Extended Group Name:** The ‘all’ group name, which names all Printer attributes according to
1597 [RFC2911] section 3.2.5. In this extension ‘all’ names all attributes specified in [RFC2911]
1598 plus those named in items 1 and 2 of this list.

1599 **11.2.4 Get-Subscription-Attributes operation**

1600 This operation allows a client to request the values of the attributes of a Subscription Object.

1601 A Printer MUST support this operation.

1602 This operation is almost identical to the Get-Job-Attributes operation (see [RFC2911] section 3.3.4).
1603 The only differences are that the operation is directed at a Subscription Object rather than a Job object,
1604 and the returned attribute group contains Subscription Object attributes rather than Job object
1605 attributes.

1606 Access Rights: The authenticated user (see [RFC2911] section 8.3) performing this operation MUST
1607 (1) be the Subscription Object owner, (2) have Operator or Administrator access rights for this Printer
1608 (see [RFC2911] sections 1 and 8.5), or (3) be otherwise authorized by the Printer's administrator-
1609 configured security policy to query the Subscription Object for the target job. Otherwise the Printer
1610 MUST reject the operation and return: the 'client-error-forbidden', 'client-error-not-authenticated', or
1611 'client-error-not-authorized' status code as appropriate. Furthermore, the Printer's security policy
1612 MAY limit which attributes are returned, in a manner similar to the Get-Job-Attributes operation (see
1613 [RFC2911] end of section 3.3.4.2).

1614 **11.2.4.1 Get-Subscription-Attributes Request**

1615 The following groups of attributes are part of the Get-Subscription-Attributes request:

1616 Group 1: Operation Attributes

1617 Natural Language and Character Set:

1618 The "attributes-charset" and "attributes-natural-language" attributes as described in section
1619 [RFC2911] 3.1.4.1.

1620

1621 Target:

1622 The "printer-uri" attribute which defines the target for this operation as described in
1623 [RFC2911] section 3.1.5.

1624

1625 Requesting User Name:

1626 The "requesting-user-name" attribute SHOULD be supplied by the client as described in
1627 [RFC2911] section 8.3.

1628 **11.2.4.1.1 "notify-subscription-id" (integer (1:MAX))**

1629 The client MUST supply this attribute. The Printer MUST support this attribute. This
1630 attribute specifies the Subscription Object from which the client is requesting attributes. If the
1631 client omits this attribute, the Printer MUST reject this request with the 'client-error-bad-
1632 request' status code.

1633 **11.2.4.1.2 "requested-attributes" (1setOf keyword)**

1634 The client OPTIONALLY supplies this attribute. The Printer MUST support this attribute.
1635 This attribute specifies the attributes of the specified Subscription Object that the Printer
1636 MUST return in the response. Each value of this attribute is either an attribute name (defined
1637 in sections 5.3 and 5.4) or an attribute group name. The attribute group names are:

1638

1639 - 'subscription-template': all attributes that are both defined in section 5.3 and present on
1640 the specified Subscription Object (column 1 of Table 1).

1641 - 'subscription-description': all attributes that are both defined in section 5.4 and present
1642 on the specified Subscription Object (Table 2).

1643 - 'all': all attributes that are present on the specified Subscription Object.

1644
1645 A Printer MUST support all these group names.

1646 If the client omits this attribute, the Printer MUST respond as if this attribute had been
1647 supplied with a value of 'all'.
1648

1649 **11.2.4.2 Get-Subscription-Attributes Response**

1650 The Printer returns the following sets of attributes as part of the Get-Subscription-Attributes Response:

1651 Group 1: Operation Attributes

1652 Status Message:
1653 Same as [RFC2911].
1654

1655 Natural Language and Character Set:
1656 The "attributes-charset" and "attributes-natural-language" attributes as described in
1657 [RFC2911] section 3.1.4.2. The "attributes-natural-language" MAY be the natural language
1658 of the Subscription Object, rather than the one requested.
1659

1660 Group 2: Unsupported Attributes

1661 See [RFC2911] section 3.1.7 and section 3.2.5.2 for details on returning Unsupported
1662 Attributes.
1663

1664 The response NEED NOT contain the "requested-attributes" operation attribute with any
1665 supplied keyword values that were requested by the client but are not supported by the IPP
1666 object. If the Printer object does return unsupported attributes referenced in the "requested-
1667 attributes" operation attribute, the values of the "requested-attributes" attribute returned
1668 MUST include only the unsupported keywords that were requested by the client. If the client
1669 had requested a group name, such as 'all', the resulting unsupported attributes returned MUST
1670 NOT include attribute keyword names described in the standard but not supported by the
1671 implementation.
1672

1673 Group 3: Subscription Attributes

1674 This group contains a set of attributes with their current values. Each attribute returned in this
1675 group:
1676

1677 a) MUST be specified by the "requested-attributes" attribute in the request, AND

1678 b) MUST be present on the specified Subscription Object AND

1679 c) MUST NOT be restricted by the security policy in force. For example, a Printer MAY
1680 prohibit a client who is not the creator of a Subscription Object from seeing some or all
1681 of its attributes. See [RFC2911] end of section 3.3.4.2 and section 8.

1682 The Printer can return the attributes of the Subscription Object in any order. The client
1683 MUST accept the attributes in any order.
1684

1685 **11.2.5 Get-Subscriptions operation**

1686 This operation allows a client to retrieve the values of attributes of all Subscription Objects belonging
1687 to a Job or Printer.

1688 A Printer MUST supported this operation.

1689 This operation is similar to the Get-Subscription-Attributes operation, except that this Get-
1690 Subscriptions operation returns attributes from possibly more than one object.

1691 This operation is similar to the Get-Jobs operation (see [RFC2911] section 3.2.6), except that the
1692 operation returns Subscription Objects rather than Job objects.

1693 Access Rights: To query Per-Job Subscription Objects of the specified job (client supplied the “notify-
1694 job-id” operation attribute - see section 11.2.5.1.1), the authenticated user (see [RFC2911] section 8.3)
1695 performing this operation MUST (1) be the Subscription Object owner, (2) have Operator or
1696 Administrator access rights for this Printer (see [RFC2911] sections 1 and 8.5), or (3) be otherwise
1697 authorized by the Printer’s administrator-configured security policy to query the Subscription Object
1698 for the target job. To query Per-Printer Subscription Objects of the Printer (client omits the “notify-
1699 job-id” operation attribute - see section 11.2.5.1.1), the authenticated user (see [RFC2911] section 8.3)
1700 performing this operation MUST (1) have Operator or Administrator access rights for this Printer (see
1701 [RFC2911] sections 1 and 8.5), or (2) be otherwise authorized by the Printer’s administrator-
1702 configured security policy to query Per-Printer Subscription Objects for the target Printer. Otherwise
1703 the Printer MUST reject the operation and return: the ‘client-error-forbidden’, ‘client-error-not-
1704 authenticated’, or ‘client-error-not-authorized’ status code as appropriate. Furthermore, the Printer’s
1705 security policy MAY limit which attributes are returned, in a manner similar to the Get-Jobs and Get-
1706 Printer-Attributes operations (see [RFC2911] end of sections 3.2.6.2 and 3.2.5.2).

1707 **11.2.5.1 Get-Subscriptions Request**

1708 The following groups of attributes are part of the Get-Subscriptions request:

1709 Group 1: Operation Attributes

1710 Natural Language and Character Set:

1711 The “attributes-charset” and “attributes-natural-language” attributes as described in
1712 [RFC2911] section 3.1.4.1.

1713

1714 Target:

1715 The “printer-uri” attribute which defines the target for this operation as described in
1716 [RFC2911] section 3.1.5.

1717

1718 Requesting User Name:
1719 The "requesting-user-name" attribute SHOULD be supplied by the client as described in
1720 [RFC2911] section 8.3.

1721 **11.2.5.1.1 "notify-job-id" (integer(1:MAX))**

1722 If the client specifies this attribute, the Printer returns the specified attributes of all Per-Job
1723 Subscription Objects associated with the Job whose "job-id" attribute value equals the value
1724 of this attribute. If the client does not specify this attribute, the Printer returns the specified
1725 attributes of all Per-Printer Subscription Objects. Note: there is no way to get all Per-Job
1726 Subscriptions known to the Printer in a single operation. A Get-Jobs operation followed by a
1727 Get-Subscriptions operation for each Job will return all Per-Job Subscriptions.

1728 **11.2.5.1.2 "limit" (integer(1:MAX))**

1729 The client OPTIONALLY supplies this attribute. The Printer MUST support this attribute. It
1730 is an integer value that determines the maximum number of Subscription Objects that a client
1731 will receive from the Printer even if the "my-subscriptions" attribute constrains which
1732 Subscription Objects are returned. The limit is a "stateless limit" in that if the value supplied
1733 by the client is 'N', then only the first 'N' Subscription Objects are returned in the Get-
1734 Subscriptions Response. There is no mechanism to allow for the next 'M' Subscription
1735 Objects after the first 'N' Subscription Objects. If the client does not supply this attribute, the
1736 Printer responds with all applicable Subscription Objects.

1737 **11.2.5.1.3 "requested-attributes" (1setOf type2 keyword)**

1738 The client OPTIONALLY supplies this attribute. The Printer MUST support this attribute.
1739 This attribute specifies the attributes of the specified Subscription Objects that the Printer
1740 MUST return in the response. Each value of this attribute is either an attribute name (defined
1741 in sections 5.3 and 5.4) or an attribute group name (defined in section 11.2.4.1). If the client
1742 omits this attribute, the Printer MUST respond as if the client had supplied this attribute with
1743 the one value: 'notify-subscription-id'.

1744 **11.2.5.1.4 "my-subscriptions" (boolean)**

1745 The client OPTIONALLY supplies this attribute. The Printer MUST support this attribute. If
1746 the value is 'false', the Printer MUST consider the Subscription Objects from all users as
1747 candidates. If the value is 'true', the Printer MUST return the Subscription Objects created by
1748 the requesting user of this request. If the client does not supply this attribute, the Printer
1749 MUST respond as if the client had supplied the attribute with a value of 'false'. The means
1750 for authenticating the requesting user and matching the Subscription Objects is similar to that
1751 for Jobs which is described in [RFC2911] section 8.
1752

1753 **11.2.5.2 Get-Subscriptions Response**

1754 The Printer returns the following sets of attributes as part of the Get-Subscriptions Response:

1755 Group 1: Operation Attributes

1756 Status Message:

1757 Same as [RFC2911].

1758

1759 Natural Language and Character Set:

1760 The “attributes-charset” and “attributes-natural-language” attributes as described in
1761 [RFC2911] section 3.1.4.2.

1762

1763 Group 2: Unsupported Attributes

1764 Same as for Get-Subscription-Attributes.

1765

1766 Groups 3 to N: Subscription Attributes

1767 The Printer responds with one Subscription Attributes Group for each requested Subscription
1768 Object (see the “notify-job-id” attribute in the Operation Attributes Group of this operation).

1769

1770 The Printer returns Subscription Objects in any order.

1771

1772 If the “limit” attribute is present in the Operation Attributes group of the request, the number
1773 of Subscription Attributes Groups in the response MUST NOT exceed the value of the “limit”
1774 attribute.

1775

1776 If there are no Subscription Objects associated with the specified Job or Printer, the Printer
1777 MUST return zero Subscription Attributes Groups and it MUST NOT treat this case as an
1778 error, i.e., the status-code MUST be ‘successful-ok’ unless something else causes the status
1779 code to have some other value.

1780

1781 See the Group 3 response (Subscription Attributes Group) of the Get-Subscription-Attributes
1782 operation (section 11.2.4.2) for the attributes that a Printer returns in this group.

1783

1784 11.2.6 Renew-Subscription operation

1785 This operation allows a client to request the Printer to extend the lease on a Per-Printer Subscription
1786 Object.

1787 The Printer MUST support this operation.

1788 The Printer MUST accept this request for a Per-Printer Subscription Object in any of the target
1789 Printer’s states, i.e., ‘idle’, ‘processing’, or ‘stopped’, but MUST NOT change the Printer’s “printer-
1790 state” attribute.

1791 The Printer **MUST** reject this request for a Per-Job Subscription Object because it has no lease (see
1792 section 5.4.3). The status code returned **MUST** be ‘client-error-not-possible’.

1793 *Access Rights:* The authenticated user (see [RFC2911] section 8.3) performing this operation **MUST**
1794 (1) be the owner of the Per-Printer Subscription Object, (2) have Operator or Administrator access
1795 rights for the Printer (see [RFC2911] sections 1 and 8.5), or (3) be otherwise authorized by the
1796 Printer’s administrator-configured security policy to renew Per-Printer Subscription Objects for the
1797 target Printer. Otherwise, the Printer **MUST** reject the operation and return: the ‘client-error-
1798 forbidden’, ‘client-error-not-authenticated’, or ‘client-error-not-authorized’ status code as appropriate.

1799 **11.2.6.1 Renew-Subscription Request**

1800 The following groups of attributes are part of the Renew-Subscription Request:

1801 Group 1: Operation Attributes

1802 Natural Language and Character Set:

1803 The “attributes-charset” and “attributes-natural-language” attributes as described in
1804 [RFC2911] section 3.1.4.1.

1805

1806 Target:

1807 The “printer-uri” attribute which defines the target for this operation as described in
1808 [RFC2911] section 3.1.5.

1809

1810 Requesting User Name:

1811 The “requesting-user-name” (name(MAX)) attribute **SHOULD** be supplied by the client as
1812 described in [RFC2911] section 8.3.

1813

1814 **11.2.6.1.1 “notify-subscription-id” (integer (1:MAX))**

1815 The client **MUST** supply this attribute. The Printer **MUST** support this attribute. This
1816 attribute specifies the Per-Printer Subscription Object whose lease the Printer **MUST** renew.
1817 If the client omits this attribute, the Printer **MUST** reject this request with the ‘client-error-
1818 bad-request’ status code.

1819

1820 Group 2: Subscription Template Attributes

1821 **11.2.6.1.2 “notify-lease-duration” (integer(0:MAX))**

1822 The client **MAY** supply this attribute. It indicates the number of seconds to renew the lease
1823 for the specified Subscription Object. A value of 0 requests an infinite lease (which **MAY**
1824 require Operator access rights). If the client omits this attribute, the Printer **MUST** use the
1825 value of the Printer’s “notify-lease-duration-default” attribute. See section 5.3.8 for more
1826 details.

1827

1828 **11.2.6.2 Renew-Subscription Response**

1829 The Printer returns the following sets of attributes as part of the Renew-Subscription Response:

1830 Group 1: Operation Attributes

1831 Status Message:

1832 Same as [RFC2911].

1833

1834 The following are some of the status codes returned (see [RFC2911]):

1835

1836 **successful-ok:** The operation successfully renewed the lease on the Subscription Object
1837 for the requested duration.1838 **successful-ok-ignored-or-substituted-attributes:** The operation successfully renewed
1839 the lease on the Subscription Object for some duration other than the amount
1840 requested.1841 **client-error-not-possible:** The operation failed because the “notify-subscription-id”
1842 Operation attribute identified a Per-Job Subscription Object.1843 **client-error-not-found:** The operation failed because the “notify-subscription-id”
1844 Operation attribute identified a non-existent Subscription Object.

1845

1846 Natural Language and Character Set:

1847 The “attributes-charset” and “attributes-natural-language” attributes as described in
1848 [RFC2911] section 3.1.4.2. The “attributes-natural-language” MAY be the natural language
1849 of the Subscription Object, rather than the one requested.

1850

1851 Group 2: Unsupported Attributes

1852 See [RFC2911] section 3.1.7 for details on returning Unsupported Attributes.

1853

1854 Group 3: Subscription Attributes

1855 The Printer MUST return the following Subscription Attribute:

1856 **11.2.6.2.1 “notify-lease-duration” (integer(0:MAX))**1857 The value of this attribute MUST be the number of seconds that the Printer has granted for the
1858 lease of the Subscription Object (see section 5.3.8 for details, such as the value of this
1859 attribute when the Printer doesn't support the requested value).1860 **11.2.7 Cancel-Subscription operation**1861 This operation allows a client to delete a Subscription Object and stop the Printer from sending more
1862 Event Notifications. Once performed, there is no way to reference the Subscription Object.

1863 A Printer MUST supported this operation.

1864 The Printer MUST accept this request in any of the target Printer's states, i.e., 'idle', 'processing', or
1865 'stopped', but MUST NOT change the Printer's "printer-state" attribute.

1866 If the specified Subscription Object is a Per-Job Subscription Object, the Printer MUST accept this
1867 request in any of the target Job's states, but MUST NOT change the Job's "job-state" attribute or affect
1868 the Job.

1869 Note: There is no way to change any attributes on a Subscription Object, except the "notify-lease-
1870 duration" attribute (using the Renew-Subscription operation). In order to change other attributes, a
1871 client performs a Subscription Creation Operation and Cancel-Subscription operation on the old
1872 Subscription Object. If the client wants to avoid missing Event Notifications, it performs the
1873 Subscription Creation Operation first. If this order would create too many Subscription Objects on the
1874 Printer, the client reverses the order.

1875 *Access Rights:* The authenticated user (see [RFC2911] section 8.3) performing this operation MUST
1876 (1) be the owner of the Subscription Object, (2) have Operator or Administrator access rights for the
1877 Printer (see [RFC2911] sections 1 and 8.5), or (3) be otherwise authorized by the Printer's
1878 administrator-configured security policy to cancel the target Subscription Object. Otherwise, the
1879 Printer MUST reject the operation and return: the 'client-error-forbidden', 'client-error-not-
1880 authenticated', or 'client-error-not-authorized' status code as appropriate.

1881 **11.2.7.1 Cancel-Subscription Request**

1882 The following groups of attributes are part of the Cancel-Subscription Request:

1883 Group 1: Operation Attributes

1884 Natural Language and Character Set:

1885 The "attributes-charset" and "attributes-natural-language" attributes as described in
1886 [RFC2911] section 3.1.4.1.

1887

1888 Target:

1889 The "printer-uri" attribute which defines the target for this operation as described in
1890 [RFC2911] section 3.1.5.

1891

1892 Requesting User Name:

1893 The "requesting-user-name" attribute SHOULD be supplied by the client as described in
1894 [RFC2911] section 8.3.

1895 **11.2.7.1.1 "notify-subscription-id" (integer (1:MAX))**

1896 The client MUST supply this attribute. The Printer MUST support this attribute. This
1897 attribute specifies the Subscription Object that the Printer MUST cancel. If the client omits

1898 this attribute, the Printer MUST reject this request with the ‘client-error-bad-request’ status
1899 code.
1900

1901 **11.2.7.2 Cancel-Subscription Response**

1902 The Printer returns the following sets of attributes as part of the Cancel-Subscription Response:

1903 Group 1: Operation Attributes

1904 Status Message:
1905 Same as [RFC2911].

1906
1907 The following are some of the status codes returned (see [RFC2911]):

1908
1909 **successful-ok:** The operation successfully canceled (deleted) the Subscription Object.
1910 **client-error-not-found:** The operation failed because the “notify-subscription-id”
1911 Operation attribute identified a non-existent Subscription Object.

1912
1913 Natural Language and Character Set:
1914 The “attributes-charset” and “attributes-natural-language” attributes as described in
1915 [RFC2911] section 3.1.4.2. The “attributes-natural-language” MAY be the natural language
1916 of the Subscription Object, rather than the one requested.

1917
1918 Group 2: Unsupported Attributes

1919 See [RFC2911] section 3.1.7 for details on returning Unsupported Attributes.
1920

1921 **12 Status Codes**

1922 The following status codes are defined as extensions for Notification and are returned as the value of
1923 the “status-code” parameter in the Operation Attributes Group of a response (see [RFC2911] section
1924 3.1.6.1). Operations in this document can also return the status codes defined in section 13 of
1925 [RFC2911]. The ‘successful-ok’ status code is an example of such a status code.

1926 **12.1 successful-ok-ignored-subscriptions (0x0003)**

1927 The Subscription Creation Operation was unable to create all requested Subscription Objects.

1928 For a Create-Job-Subscriptions or Create-Printer-Subscriptions operation, this status code means that
1929 the Printer created one or more Subscription Objects, but not all requested Subscription Objects.

1930 For a Job Creation operation, this status code means that the Printer created the Job along with zero or
1931 more Subscription Objects. The Printer returns this status code even if other job attributes are
1932 unsupported or in conflict. That is, if an IPP Printer finds a warning that would allow it to return

1933 'successful-ok-ignored-subscriptions' and either 'successful-ok-ignored-or-substituted-attributes'
1934 and/or 'successful-ok-conflicting-attributes', it MUST return 'successful-ok-ignored-subscriptions'.

1935 **12.2 client-error-ignored-all-subscriptions (0x0414)**

1936 This status code is the same as 'successful-ok-ignored-subscriptions' except that only the Create-Job-
1937 Subscriptions and Create-Printer-Subscriptions operation return it. They return this status code only
1938 when the Printer creates zero Subscription Objects.

1939 **13 Status Codes in Subscription Attributes Groups**

1940 This section contains values of the "notify-status-code" (type2 enum) attribute that the Printer returns
1941 in a Subscription Attributes Group in a response when the corresponding Subscription Object:

- 1942 1. is not created or
- 1943 2. is created and some of the client-supplied attributes are not supported.

1944 The following sections are ordered in decreasing order of importance of the status-codes.

1945 **13.1 client-error-uri-scheme-not-supported (0x040C)**

1946 This status code is defined in [RFC2911]. This document extends its meaning and allows it to be in a
1947 Subscription Attributes Group of a response.

1948 The scheme of the client-supplied URI in a "notify-recipient-uri" Subscription Template Attribute in a
1949 Subscription Creation Operation is not supported. See section 5.3.1.

1950 **13.2 client-error-attributes-or-values-not-supported (0x040B)**

1951 This status code is defined in [RFC2911]. This document extends its meaning and allows it to be in a
1952 Subscription Attributes Group of a response.

1953 The method of the client-supplied keyword in a "notify-pull-method" Subscription Template Attribute
1954 in a Subscription Creation Operation is not supported. See section 5.3.2.

1955 **13.3 client-error-too-many-subscriptions (0x0415)**

1956 The number of Subscription Objects supported by the Printer would be exceeded if this Subscription
1957 Object were created (see section 5.2).

1958 **13.4 successful-ok-too-many-events (0x0005)**

1959 The client supplied more Events in the “notify-events” operation attribute of a Subscription Creation
 1960 Operation than the Printer supports, as indicated in its “notify-max-events-supported” Printer attribute
 1961 (see section 5.3.3).

1962 **13.5 successful-ok-ignored-or-substituted-attributes (0x0001)**

1963 This status code is defined in [RFC2911]. This document extends its meaning to include unsupported
 1964 Subscription Template Attributes and it can appear in a Subscription Attributes Group.

1965 **14 Encodings of Additional Attribute Tags**

1966 This section assigns values to two attributes tags as extensions to the encoding defined in [RFC2910]).

1967 The “subscription-attributes-tag” delimits Subscription Template Attributes Groups in requests and
 1968 Subscription Attributes Groups in responses.

1969 The “event-notification-attributes-tag” delimits Event Notifications in Delivery Methods that use an
 1970 IPP-like encoding.

1971 The following table specifies the values for the delimiter tags:

Tag Value (Hex)	Meaning
0x06	“subscription-attributes-tag”
0x07	“event-notification-attributes-tag”

1972 **15 Conformance Requirements**

1973 It is OPTIONAL for IPP clients and Printers to implement this Event Notification specification.

1974 **15.1 Conformance requirements for clients**

1975 If this Event Notification specification is implemented by a client, the client MUST support the
 1976 ‘ippget’ Pull Delivery Method and meet the conformance requirements as defined in [ipp-get-method]
 1977 for clients. A client MAY support additional Delivery Methods.

1978 **15.2 Conformance requirements for Printers**

1979 If this Event Notification specification is implemented by a Printer, the Printer MUST:

1980 - meet the Conformance Requirements detailed in section 5 of [RFC2911].

- 1981 - support the Subscription Template Attributes Group in requests and the Subscription
1982 Attributes Group in responses.
- 1983 - support all of the following attributes:
- 1984 a. REQUIRED Subscription Object attributes in section 5.
- 1985 b. REQUIRED Printer Description object attributes in section 6.
- 1986 c. REQUIRED attributes in Event Notification content in section 8.
- 1987 - support the 'ippget' Pull Delivery Method and meet the conformance requirements as defined
1988 in [ipp-get-method] for Printers. The Printer MAY support additional Push and Pull Delivery
1989 Methods.
- 1990 - send Event Notifications that conform to the requirements of section 9 and the requirements
1991 of the Delivery Method Document for each supported Delivery Method (the conformance
1992 requirements for Delivery Method Documents is specified in section 10).
- 1993 - for all of the Job Creation Operations that the Printer supports, MUST support the
1994 REQUIRED extensions for notification defined in section 11.1.3.
- 1995 - meet the conformance requirements for operations as described in Table 16 and meet the
1996 requirements for Printers as specified in the indicated sub-sections of section 11:

Table 16 – Printer Conformance Requirements for Operations

Operation	Printer Conformance Requirements
Create-Printer-Subscriptions (section 11.1.2)	REQUIRED
Create-Job-Subscriptions (section 11.1.1)	OPTIONAL
Get-Subscription-Attributes (section 11.2.3)	REQUIRED
Get-Subscriptions (section 11.2.5)	REQUIRED
Renew-Subscription (section 11.2.6)	REQUIRED
Cancel-Subscription (section 11.2.7)	REQUIRED

1998

1999 16 Normative References

- 2000 [ipp-get-method]
- 2001 Herriot, R., and T. Hastings, "Internet Printing Protocol (IPP): The 'ippget' Delivery Method for
2002 Event Notifications", <draft-ietf-ipp-notify-get-07.txt>, June 27, 2002.
- 2003 [ipp-prog]
- 2004 Hastings, T., Bergman, R., and H. Lewis, "IPP: Job Progress Attributes", <draft-ietf-ipp-job-prog-
2005 03.txt> work in progress, July 17, 2001.

- 2006 [RFC2119]
2007 S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119 , March 1997
- 2008 [RFC2396]
2009 Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifiers (URI): Generic
2010 Syntax", RFC 2396, August 1998.
- 2011 [RFC2717]
2012 R. Petke and I. King, "Registration Procedures for URL Scheme Names", RFC 2717, November
2013 1999.
- 2014 [RFC2910]
2015 Herriot, R., Butler, S., Moore, P., and R. Turner, "Internet Printing Protocol/1.1: Encoding and
2016 Transport", RFC 2910, September 2000.
- 2017 [RFC2911]
2018 deBry, R., Hastings, T., Herriot, R., Isaacson, S., and P. Powell, "Internet Printing Protocol/1.1:
2019 Model and Semantics", RFC 2911, September 2000.

2020 **17 Informative References**

- 2021 [IANA-CON]
2022 Narte, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs",
2023 BCP 26, RFC 2434, October 1998.
- 2024 [ipp-not-req]
2025 deBry, R., Lewis, H., and T. Hastings, "Internet Printing Protocol/1.1: Requirements for IPP
2026 Notifications", <draft-ietf-ipp-not-06.txt>, work in progress, July 17, 2001.
- 2027 [RFC2565]
2028 Herriot, R., Butler, S., Moore, P., and R. Turner, "Internet Printing Protocol/1.0: Encoding and
2029 Transport", RFC 2565, April 1999.
- 2030 [RFC2566]
2031 deBry, R., , Hastings, T., Herriot, R., Isaacson, S., and P. Powell, "Internet Printing Protocol/1.0:
2032 Model and Semantics", RFC 2566, April 1999.
- 2033 [RFC2567]
2034 Wright, D., "Design Goals for an Internet Printing Protocol", RFC 2567, April 1999.
- 2035 [RFC2568]
2036 Zilles, S., "Rationale for the Structure and Model and Protocol for the Internet Printing Protocol",
2037 RFC 2568, April 1999.

- 2038 [RFC2569]
2039 Herriot, R., Hastings, T., Jacobs, N., and J. Martin, "Mapping between LPD and IPP Protocols",
2040 RFC 2569, April 1999.
- 2041 [RFC2616]
2042 Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee,
2043 "Hypertext Transfer Protocol - HTTP/1.1", RFC 2616, June 1999.
- 2044 [RFC3196]
2045 Hastings, T., Manros, C., Zehler, P., Kugler, C., and H. Holst, "Internet Printing Protocol/1.1:
2046 Implementer's Guide", RFC3196, November 2001.

2047 **18 Security Considerations**

2048 Clients submitting Notification requests to the IPP Printer have the same security issues as submitting
2049 an IPP/1.1 print job request (see [RFC2911] section 3.2.1 and section 8). The same mechanisms used
2050 by IPP/1.1 can therefore be used by the client Notification submission. Operations that require
2051 authentication can use the HTTP authentication. Operations that require privacy can use the
2052 HTTP/TLS privacy. As with IPP/1.1 Print Job Objects, if there is no security on Subscription Objects,
2053 sequential assignment of subscription-ids exposes the system to a passive traffic monitoring threat.

2054 **18.1 Client access rights**

2055 The Subscription Object access control model is the same as the access control model for Job objects.
2056 The client **MUST** have the following access rights for the indicated Subscription operations:

- 2057 1. Create-Job-Subscriptions (see section 11.1.1): A Per-Job Subscription object is associated with
2058 a Job. To create Per-Job Subscription Objects, the authenticated user (see [RFC2911] section
2059 8.3) performing this operation **MUST** (1) be the job owner, (2) have Operator or Administrator
2060 access rights for this Printer (see [RFC2911] sections 1 and 8.5), or (3) be otherwise authorized
2061 by the Printer's administrator-configured security policy to create Per-Job Subscription Objects
2062 for the target job.
- 2063 2. Create-Printer-Subscriptions (see section 11.1.2): A Per-Printer Subscription object is
2064 associated with the Printer. To create Per-Printer Subscription Objects, the authenticated user
2065 (see [RFC2911] section 8.3) performing this operation **MUST** (1) have Operator or
2066 Administrator access rights for this Printer (see [RFC2911] sections 1 and 8.5) or (2) be
2067 otherwise authorized by the Printer's administrator-configured security policy to create Per-
2068 Printer Subscription Objects for this Printer.
- 2069 3. Get-Subscription-Attributes (see section 11.2.4): The access control model for this operation is
2070 the same as that of the Get-Job-Attributes operation (see [RFC2911] section 3.3.4). The
2071 primary difference is that a Get-Subscription-Attributes operation is directed at a Subscription
2072 Object rather than at a Job object, and a returned attribute group contains Subscription Object
2073 attributes rather than Job object attributes. To query the specified Subscription Object, the

2074 authenticated user (see [RFC2911] section 8.3) performing this operation MUST (1) be the
2075 Subscription Object owner, (2) have Operator or Administrator access rights for this Printer
2076 (see [RFC2911] sections 1 and 8.5), or (3) be otherwise authorized by the Printer's
2077 administrator-configured security policy to query the Subscription Object for the target job.
2078 Furthermore, the Printer's security policy MAY limit which attributes are returned, in a manner
2079 similar to the Get-Job-Attributes operation (see [RFC2911] end of section 3.3.4.2).

2080 4. Get-Subscriptions (see section 11.2.5): The access control model for this operation is the same
2081 as that of the Get-Jobs operation (see [RFC2911] section 3.2.6). The primary difference is that
2082 the operation is directed at Subscription Objects rather than at Job objects, and the returned
2083 attribute groups contain Subscription Object attributes rather than Job object attributes. To
2084 query Per-Job Subscription Objects of the specified job (client supplied the "notify-job-id"
2085 operation attribute - see section 11.2.5.1.1), the authenticated user (see [RFC2911] section 8.3)
2086 performing this operation MUST (1) be the Subscription Object owner, (2) have Operator or
2087 Administrator access rights for this Printer (see [RFC2911] sections 1 and 8.5), or (3) be
2088 otherwise authorized by the Printer's administrator-configured security policy to query the
2089 Subscription Object for the target job. To query Per-Printer Subscription Objects of the Printer
2090 (client omits the "notify-job-id" operation attribute - see section 11.2.5.1.1), the authenticated
2091 user (see [RFC2911] section 8.3) performing this operation MUST (1) have Operator or
2092 Administrator access rights for this Printer (see [RFC2911] sections 1 and 8.5), or (2) be
2093 otherwise authorized by the Printer's administrator-configured security policy to query Per-
2094 Printer Subscription Objects for the target Printer. Furthermore, the Printer's security policy
2095 MAY limit which attributes are returned, in a manner similar to the Get-Job-Attributes
2096 operation (see [RFC2911] end of section 3.2.6.2).

2097 5. Renew-Subscriptions (see section 11.2.6): The authenticated user (see [RFC2911] section 8.3)
2098 performing this operation MUST (1) be the owner of the Per-Printer Subscription Object, (2)
2099 have Operator or Administrator access rights for the Printer (see [RFC2911] sections 1 and
2100 8.5), or (3) be otherwise authorized by the Printer's administrator-configured security policy to
2101 renew Per-Printer Subscription Objects for the target Printer

2102 6. Cancel-Subscription (see section 11.2.7): The authenticated user (see [RFC2911] section 8.3)
2103 performing this operation MUST (1) be the owner of the Subscription Object, (2) have Operator
2104 or Administrator access rights for the Printer (see [RFC2911] sections 1 and 8.5), or (3) be
2105 otherwise authorized by the Printer's administrator-configured security policy to cancel the
2106 target Subscription Object.

2107 The standard security concerns (delivery to the right user, privacy of content, tamper proof content)
2108 apply to each Delivery Method. Some Delivery Methods are more secure than others. Each Delivery
2109 Method Document MUST discuss its Security Considerations.

2110 **18.2 Printer security threats**

2111 Notification trap door: If a Printer supports the OPTIONAL "notify-attributes" Subscription Template
2112 attribute (see section 5.3.4) where the client can request that the Printer return any specified Job,

2113 Printer, and Subscription object attributes, the Printer MUST apply the same security policy to these
 2114 requested attributes in the Get-Notifications request as it does for the Get-Jobs, Get-Job-Attributes,
 2115 Get-Printer-Attributes, and Get-Subscription-Attributes requests.

2116 18.3 Notification Recipient security threats

2117 Unwanted Events Notifications (spam): For any Push Delivery Method, by far the biggest security
 2118 concern is the abuse of notification: sending unwanted Event Notifications to third parties (i.e., spam).
 2119 The problem is made worse by notification addresses that may be redistributed to multiple parties.
 2120 There exist scenarios where third party notification is used (see Scenario #2 and #3 in [ipp-not-req]).
 2121 Any fully secure solution would require active agreement of all recipients before sending out anything.

2122 19 IANA Considerations

2123 This section contains the registration information for IANA to add to the various IPP Registries
 2124 according to the procedures defined in RFC 2911 [RFC2911] section 6 to cover the definitions in this
 2125 document. In addition, this section defines how Events and Delivery Methods will be registered when
 2126 they are defined in other documents.

2127 *Note to RFC Editors: Replace RFC NNNN below with the RFC number for this document, so that it*
 2128 *accurately reflects the content of the information for the IANA Registry.*

2129 19.1 Attribute Registrations

2130 The following table lists all the attributes defined in this document. These are to be registered
 2131 according to the procedures in RFC 2911 [RFC2911] section 6.2.

2132	Subscription Template attributes:	Ref.	Section:
2133	notify-recipient-uri (uri)	RFC NNNN	5.3.1
2134	notify-schemes-supported (1setOf uriScheme)	RFC NNNN	5.3.1
2135	notify-pull-method (type2 keyword)	RFC NNNN	5.3.2
2136	notify-pull-method-supported (1setOf type2 keyword)		
2137		RFC NNNN	5.3.2
2138	notify-events (1setOf type2 keyword)	RFC NNNN	5.3.3
2139	notify-events-default (1setOf type2 keyword)	RFC NNNN	5.3.3
2140	notify-events-supported (1setOf type2 keyword)	RFC NNNN	5.3.3
2141	notify-max-events-supported (integer(2:MAX))	RFC NNNN	5.3.3
2142	notify-attributes (1setOf type2 keyword)	RFC NNNN	5.3.4
2143	notify-attributes-supported (1setOf type2 keyword)		
2144		RFC NNNN	5.3.4
2145	notify-user-data (octetString(63))	RFC NNNN	5.3.5
2146	notify-charset (charset)	RFC NNNN	5.3.6
2147	notify-natural-language (naturalLanguage)	RFC NNNN	5.3.7
2148	notify-lease-duration (integer(0:67108863))	RFC NNNN	5.3.8
2149	notify-lease-duration-default (integer(0:67108863))		
2150		RFC NNNN	5.3.8

2151	notify-lease-duration-supported (1setOf (integer(0: 67108863)			
2152	rangeOfInteger(0:67108863)))	RFC	NNNN	5.3.8
2153	notify-time-interval (integer(0:MAX))	RFC	NNNN	5.3.9
2154				
2155	Subscription Description Attributes:			
2156	notify-subscription-id (integer (1:MAX)))	RFC	NNNN	5.4.1
2157	notify-sequence-number (integer (0:MAX)))	RFC	NNNN	5.4.2
2158	notify-lease-expiration-time (integer(0:MAX)))	RFC	NNNN	5.4.3
2159	notify-printer-up-time (integer(1:MAX)))	RFC	NNNN	5.4.4
2160	notify-printer-uri (uri))	RFC	NNNN	5.4.5
2161	notify-job-id (integer(1:MAX)))	RFC	NNNN	5.4.6
2162	notify-subscriber-user-name (name(MAX)))	RFC	NNNN	5.4.7
2163				
2164	Printer Description Attributes:			
2165	printer-state-change-time (integer(1:MAX)))	RFC	NNNN	6.1
2166	printer-state-change-date-time (dateTime))	RFC	NNNN	6.2
2167				
2168	Attributes Only in Event Notifications			
2169	notify-subscribed-event (type2 keyword)	RFC	NNNN	8.1
2170	notify-text (text(MAX))	RFC	NNNN	8.2

2171
 2172 The resulting attribute registrations will be published in the
 2173 <ftp://ftp.iana.org/in-notes/iana/assignments/ipp/attributes/>
 2174 area.
 2175

2176 19.2 Additional Enum Attribute Value Registrations for the “operations-supported” Printer 2177 Attribute

2178 The following table lists all the new enum attribute values defined in this document as additional type2
 2179 enum values for use with the “operations-supported” Printer Description attribute. These are to be
 2180 registered according to the procedures in RFC 2911 [RFC2911] section 6.1.

2181	type2 enum Attribute Values:	Value	Ref.	Section:
2182	Create-Printer-Subscriptions	0x0016	RFC NNNN	7.1
2183	Create-Job-Subscriptions	0x0017	RFC NNNN	7.1
2184	Get-Subscription-Attributes	0x0018	RFC NNNN	7.1
2185	Get-Subscriptions	0x0019	RFC NNNN	7.1
2186	Renew-Subscription	0x001A	RFC NNNN	7.1
2187	Cancel-Subscription	0x001B	RFC NNNN	7.1

2188
 2189 The resulting enum attribute value registrations will be published in the
 2190 <ftp://ftp.iana.org/in-notes/iana/assignments/ipp/attribute-values/operations-supported/>
 2191 area.
 2192

2193 19.3 Operation Registrations

2194 The following table lists all of the operations defined in this document. These are to be registered
2195 according to the procedures in RFC 2911 [RFC2911] section 6.4.

2196	Operations:	Ref.	Section:
2197	Create-Job-Subscriptions Operation	RFC NNNN	11.1.1
2198	Create-Printer-Subscriptions Operation	RFC NNNN	11.1.2
2199	Job Creation Operations - Extensions	RFC NNNN	11.1.3
2200	Validate-Job Operation - Extensions	RFC NNNN	11.2.2
2201	Get-Printer-Attributes - Extensions	RFC NNNN	11.2.3
2202	Get-Subscription-Attributes Operation	RFC NNNN	11.2.4
2203	Get-Subscriptions Operation	RFC NNNN	11.2.5
2204	Renew-Subscription Operation	RFC NNNN	11.2.6
2205	Cancel-Subscription Operation	RFC NNNN	11.2.7

2207 The resulting operation registrations will be published in the
2208 <ftp://ftp.iana.org/in-notes/iana/assignments/ipp/operations/>
2209 area.
2210

2211 19.4 Status code Registrations

2212 The following table lists all the status codes defined in this document. These are to be registered
2213 according to the procedures in RFC 2911 [RFC2911] section 6.6.

2214	Status codes:	Ref.	Section:
2215	successful-ok-ignored-subscriptions (0x0003)	RFC NNNN	12.1
2216	client-error-ignored-all-subscriptions (0x0414)	RFC NNNN	12.2
2217			
2218	Status Codes in Subscription Attributes Groups:		
2219	client-error-uri-scheme-not-supported (0x040C)	RFC NNNN	13.1
2220	client-error-attributes-or-values-not-supported (0x040B)		
2221		RFC NNNN	13.2
2222	client-error-too-many-subscriptions (0x0415)	RFC NNNN	13.3
2223	successful-ok-too-many-events (0x0005)	RFC NNNN	13.4
2224	successful-ok-ignored-or-substituted-attributes (0x0001)		
2225		RFC NNNN	13.5

2227 The resulting status code registrations will be published in the
2228 <ftp://ftp.iana.org/in-notes/iana/assignments/ipp/status-codes/>
2229 area.
2230

2231 19.5 Attribute Group tag Registrations

2232 The following table lists all the attribute group tags defined in this document. These are to be
2233 registered according to the procedures in RFC 2911 [RFC2911] section 6.5.

	Attribute Group Tags:	Tag Value:	Ref.	Section:
2234	subscription-attributes-tag	0x06	RFC NNNN	14
2235	event-notification-attributes-tag	0x07	RFC NNNN	14
2236				
2237				

2238 The resulting attribute group tag registrations will be published in the
 2239 <ftp://ftp.iana.org/in-notes/iana/assignments/ipp/attribute-group-tags/>
 2240 area.
 2241

2242 19.6 Registration of Events

2243 When other document define additional type2 keywords to be used with the “notify-events”
 2244 Subscription Template attribute (see section 5.3.3)), these event keywords will be registered according
 2245 to the procedures of [RFC2911] section 7.1 as additional attribute values for use with the “notify-
 2246 events” Subscription Template attribute, i.e., the "notify-events", "notify-events-default", and "notify-
 2247 events-supported" attributes.

2248 Therefore, the IPP Registry entry for an Event will be of the form:

	type2 enum Attribute Values:	Ref.	Section:
2249	<scheme name>	RFC xxxx	m.n
2250			
2251			

2252 The resulting type2 keyword attribute values will be published in the
 2253 <ftp://ftp.iana.org/in-notes/iana/assignments/ipp/attribute-values/notify-events/>
 2254 area.
 2255

2256 19.7 Registration of Event Notification Delivery Methods

2257 This section describes the requirements and procedures for registration and publication of Event
 2258 Notification Delivery Methods and for the submission of such proposals.

2259 19.7.1 Requirements for Registration of Event Notification Delivery Methods

2260 Registered IPP Event Notification Delivery Methods are expected to follow a number of requirements
 2261 described below.

2262 19.7.1.1 Required Characteristics

2263 A Delivery Method Document MUST either (1) contain all of the semantics of the Delivery Method or
 2264 (2) contain the IPP Delivery Method registration requirements and a profile of some other protocol that
 2265 in combination is the Delivery Method (e.g., mailto). The Delivery Method Document (and any
 2266 documents it requires) MUST define either (1) a URL for a Push Delivery Method that the meets the
 2267 requirements of [RFC2717]. or (2) a keyword for a Pull Delivery method.
 2268

2269 IPP Event Notification Delivery Method Documents **MUST** meet the requirements of this document
2270 (see sections 9 and 10).

2271 In addition, a Delivery Method Document **MUST** contain the following information:

2272

2273 Type of registration: IPP Event Notification Delivery Method

2274 Name of this delivery method:

2275 Proposed URL scheme name of this Push Delivery Method or the keyword name of this Pull

2276 Delivery Method:

2277 Name of proposer:

2278 Address of proposer:

2279 Email address of proposer:

2280 Is this delivery method **REQUIRED** or **OPTIONAL** for conformance to the IPP Event Notification
2281 and Subscriptions document:

2282 Is this delivery method defining Machine Consumable and/or Human Consumable content:

2283

2284 **19.7.1.2 Naming Requirements**

2285 Exactly one (URL scheme or keyword) name **MUST** be assigned to each Delivery Method.

2286 Each assigned name **MUST** uniquely identify a single Delivery Method. All Push Delivery Method
2287 names **MUST** conform to the rules for URL scheme names, according to [RFC2396] and [RFC2717]
2288 for schemes in the IETF tree. All Pull Delivery Method names **MUST** conform to the rules for
2289 keywords according to [RFC2911].

2290 **19.7.1.3 Functionality Requirements**

2291 Delivery Methods **MUST** function as a protocol that is capable of delivering (push or pull) IPP Event
2292 Notifications to Notification Recipients.

2293 **19.7.1.4 Usage and Implementation Requirements**

2294 Use of a large number of Delivery Methods may hamper interoperability. However, the use of a large
2295 number of undocumented and/or unlabelled Delivery Methods hampers interoperability even more.

2296 A Delivery Method should therefore be registered **ONLY** if it adds significant functionality that is
2297 valuable to a large community, **OR** if it documents existing practice in a large community. Note that
2298 Delivery Methods registered for the second reason should be explicitly marked as being of limited or
2299 specialized use and should only be used with prior bilateral agreement.

2300 **19.7.1.5 Publication Requirements**

2301 Delivery Method Documents **MUST** be published in a standards track, informational, or experimental
2302 RFCs.

2303 **19.7.2 Registration Procedure**

2304 The IPP WG is developing a small number of Delivery Methods which are intended to be published as
2305 standards track RFCs. However, some parties may wish to register additional Delivery Methods in the
2306 future. This section describes the procedures for these additional Delivery Methods.

2307 **19.7.2.1 Present the proposal to the Community**

2308 First the Delivery Method Document **MUST** be an Internet-Draft with a target category of standards
2309 track, informational, or experimental. The same **MUST** be true for any documents that it references.

2310 Send the proposed Delivery Method Document proposal to the “ipp@pwg.org” mailing list. This
2311 mailing list has been established by [RFC2911] for reviewing proposed registrations and discussing
2312 other IPP matters. Proposed Delivery Method Documents are not formally registered and **MUST NOT**
2313 be used until approved.

2314 The intent of the public posting is to solicit comments and feedback on the definition and suitability of
2315 the Delivery Method and the name chosen for it over a four week period.

2316 **19.7.2.2 Delivery Method Reviewer**

2317 The Delivery Method Reviewer is the same person who has been appointed by the IETF Application
2318 Area Director(s) as the IPP Designated Expert according to [RFC2911] and [IANA-CON]. When the
2319 four week period is over and the IPP Designated Expert is convinced that consensus has been achieved,
2320 the IPP Designated Expert either approves the request for registration or rejects it. Rejection may
2321 occur because of significant objections raised on the list or objections raised externally.

2322 Decisions made by the Reviewer must be posted to the ipp@pwg.org mailing list within 14 days.
2323 Decisions made by the Reviewer may be appealed to the IESG.

2324 **19.7.2.3 IANA Registration**

2325 Provided that the Delivery Method registration proposal has either passed review or has been
2326 successfully appealed to the IESG, the IANA will register the Delivery Method and make it available to
2327 the community.

2328 19.7.3 Delivery Method Document Registrations

2329 Each Push Delivery Method Document defines a URI scheme which is registered as an additional value
2330 of the “notify-schemes-supported” Printer attribute. These uriScheme values will be registered
2331 according to the procedures of [RFC2911] section 7.1 for additional attribute values. Therefore, the
2332 IPP Registry entry for a Push Delivery Method will be of the form:

2333	uriScheme Attribute Values:	Ref.	Section:
2334	<scheme name>	RFC xxxx	m.n

2335

2336 The resulting Delivery Method URI schemes will be published in the
2337 ftp://ftp.iana.org/in-notes/iana/assignments/ipp/attribute-values/notify-schemes-supported/
2338 area.

2339

2340 Each Pull Delivery Method Document defines a keyword method which is registered as an additional
2341 value of the “notify-pull-method-supported” Printer attribute. These keyword values will be registered
2342 according to the procedures of [RFC2911] section 7.1 for additional attribute values. Therefore, the
2343 IPP Registry entry for a Pull Delivery Method will be of the form:

2344	keyword Attribute Values:	Ref.	Section:
2345	<method name>	RFC xxxx	m.n

2346

2347 The resulting Delivery Method URI schemes will be published in the
2348 ftp://ftp.iana.org/in-notes/iana/assignments/ipp/attribute-values/notify-pull-method-supported/
2349 area.

2350

2351 19.7.4 Registration Template

2352 To: ipp@pwg.org
2353 Subject: Registration of a new Delivery Method

2354

2355 Delivery Method name:

2356

2357 (All Push Delivery Method names must be suitable for use as the value of a URL scheme in the IETF
2358 tree and all Pull Delivery Method names must be suitable IPP keywords according to [RFC2911])

2359

2360 Published specification(s):

2361

2362 (A specification for the Delivery Method must be openly available that accurately describes what is
2363 being registered.)

2364

2365 Person & email address to contact for further information:

2366 **20 Internationalization Considerations**

2367 This IPP Notification specification continues support for the internationalization of [RFC2911] of
2368 attributes containing text strings and names. Allowing a Subscribing Client to specify a different
2369 natural language and charset for each Subscription Object increases the internationalization support.

2370 The Printer MUST be able to localize the content of Human Consumable Event Notifications and to
2371 localize the value of “notify-text” attribute in Machine Consumable Event Notifications that it sends to
2372 Notification Recipients. For localization, the Printer MUST use the value of the “notify-charset”
2373 attribute and the “notify-natural-language” attribute in the Subscription Object supplied by the
2374 Subscribing Client.

2375 **21 Contributors**

2376 The following people made significant contributions to the design and review of this specification:

2377 Scott A. Isaacson
2378 Novell, Inc.
2379 122 E 1700 S
2380 Provo, UT 84606

2381
2382 Phone: 801-861-7366
2383 Fax: 801-861-2517
2384 e-mail: sisacson@novell.com

2385
2386 Roger deBry
2387 Utah Valley State College
2388 Orem, UT 84058

2389
2390 Phone: (801) 222-8000
2391 EMail: debryro@uvsc.edu

2392
2393 Jay Martin
2394 Underscore Inc.
2395 9 Jacqueline St.
2396 Hudson, NH 03051-5308
2397 603-889-7000
2398 fax: 775-414-0245
2399 e-mail: jkm@underscore.com

2400
2401 Michael Shepherd
2402 Xerox Corporation
2403 800 Phillips Road MS 128-51E
2404 Webster, NY 14450

2405

2406 Phone: 716-422-2338
2407 Fax: 716-265-8871
2408 e-mail: mshepherd@crt.xerox.com
2409
2410 Ron Bergman
2411 Hitachi Koki Imaging Solutions
2412 1757 Tapo Canyon Road
2413 Simi Valley, CA 93063-3394
2414
2415 Phone: 805-578-4421
2416 Fax: 805-578-4001
2417 Email: rbergma@hitachi-hkis.com

2418 **22 Author's Addresses**

2419 Robert Herriot
2420 706 Colorado Ave.
2421 Palo Alto, CA 94303
2422
2423 Phone: 650-327-4466
2424 Fax: 650-327-4466
2425 Email: bob@herriot.com

2426
2427 Tom Hastings
2428 Xerox Corporation
2429 737 Hawaii St. ESAE 231
2430 El Segundo, CA 90245

2431
2432 Phone: 310-333-6413
2433 Fax: 310-333-5514
2434 e-mail: hastings@cp10.es.xerox.com

2435
2436 IPP Web Page: <http://www.pwg.org/ipp/>
2437 IPP Mailing List: ipp@pwg.org

2438
2439 To subscribe to the ipp mailing list, send the following email:

- 2440 1) send it to majordomo@pwg.org
2441 2) leave the subject line blank
2442 3) put the following two lines in the message body:
2443 subscribe ipp
2444 end

2445
2446 Implementers of this specification document are encouraged to join the IPP Mailing List in order to
2447 participate in any discussions of clarification issues and review of registration proposals for additional

2448 attributes and values. In order to reduce spam the mailing list rejects mail from non-subscribers, so
2449 you must subscribe to the mailing list in order to send a question or comment to the mailing list.

2450 **A. Appendix - Model for Notification with Cascading Printers**

2451 With this model (see Figure 2), there is an intervening Print server between the human user and the
2452 output-device. So the system effectively has two Printers. There are two cases to consider.

- 2453 1. When the Printer 1 (in the server) generates Events, the system behaves like the client and Printer
2454 in Figure 1. In this case, Printer 1 sends Event Notifications that are shown as Event Notifications
2455 (A) of Figure 2,.
- 2456 2. When the Printer 2 (in the output-device) generates Events, there are two possible system
2457 configurations:
 - 2458 a) Printer 1 forwards the client-supplied Subscription Creation Operations to the downstream
2459 Printer 2 and lets Printer 2 send the Event Notifications directly to the Notification Recipients
2460 supplied by the Client (Event Notifications(C) in the diagram).
 - 2461 b) Printer 1 performs the client-supplied Subscription Creation Operations and also forwards the
2462 Subscription Creation Operations to Printer 2 with the Notification Recipient changed to be the
2463 Printer 1. When an Event occurs in Printer 2, Printer 2 sends the Event Notification (B) to
2464 Notification Recipient of Printer 1, which relays the received Event Notification (B) to the
2465 client-supplied Notification Recipient (as Event Notifications(A) in the diagram). Note, when a
2466 client performs a Subscription Creation Operation, Printer 1 need not forward the Subscription
2467 Creation Operation to Printer 2 if it would create a duplicate Subscription Object on Printer 2.

2468 Note: when Printer 1 is forwarding Subscription Creation Operations to Printer 2, it may request
2469 Printer 2 to create additional Subscription Objects (called "piggy-backing"). Piggy-backing is useful
2470 when:

- 2471 - Device A is configured to accept (IPP or non-IPP) requests from other servers.
- 2472 - Server S wants to receive Job Events that the client didn't request and Server S wants these
2473 Events for jobs it submits and not for other jobs.

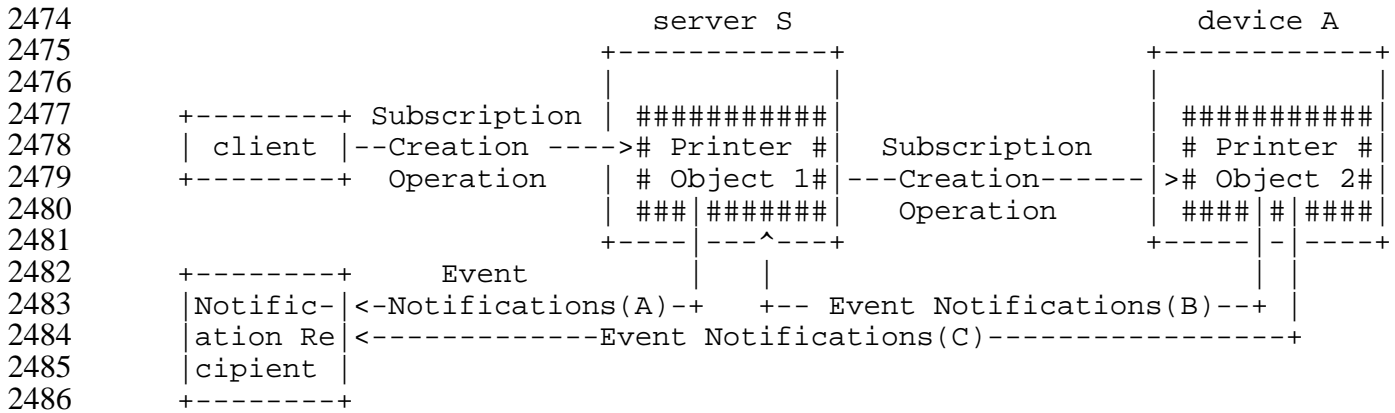


Figure 2 – Model for Notification with Cascading Printers

B. Appendix - Distributed Model for Notification

A Printer implementation could use some other remote notification service to provide some or most of the service. For example, the remote notification service could send Event Notifications using Delivery Methods that are not directly supported by the output device or server. Or, the remote notification service could store Subscription Objects (passed to it from the output device in response to Subscription Creation requests), accept Events, format the Event Notification in the natural language of the Notification Recipient, and send the Event Notifications to the Notification Recipient(s).

Figure 3 shows this partitioning. The interface between the output device (or server) and the remote notification service is outside the scope of this document and is intended to be transparent to the client and this document. The combination of the output device (or server) and the notification service together constitute an IPP Printer conforming to this Notification document.

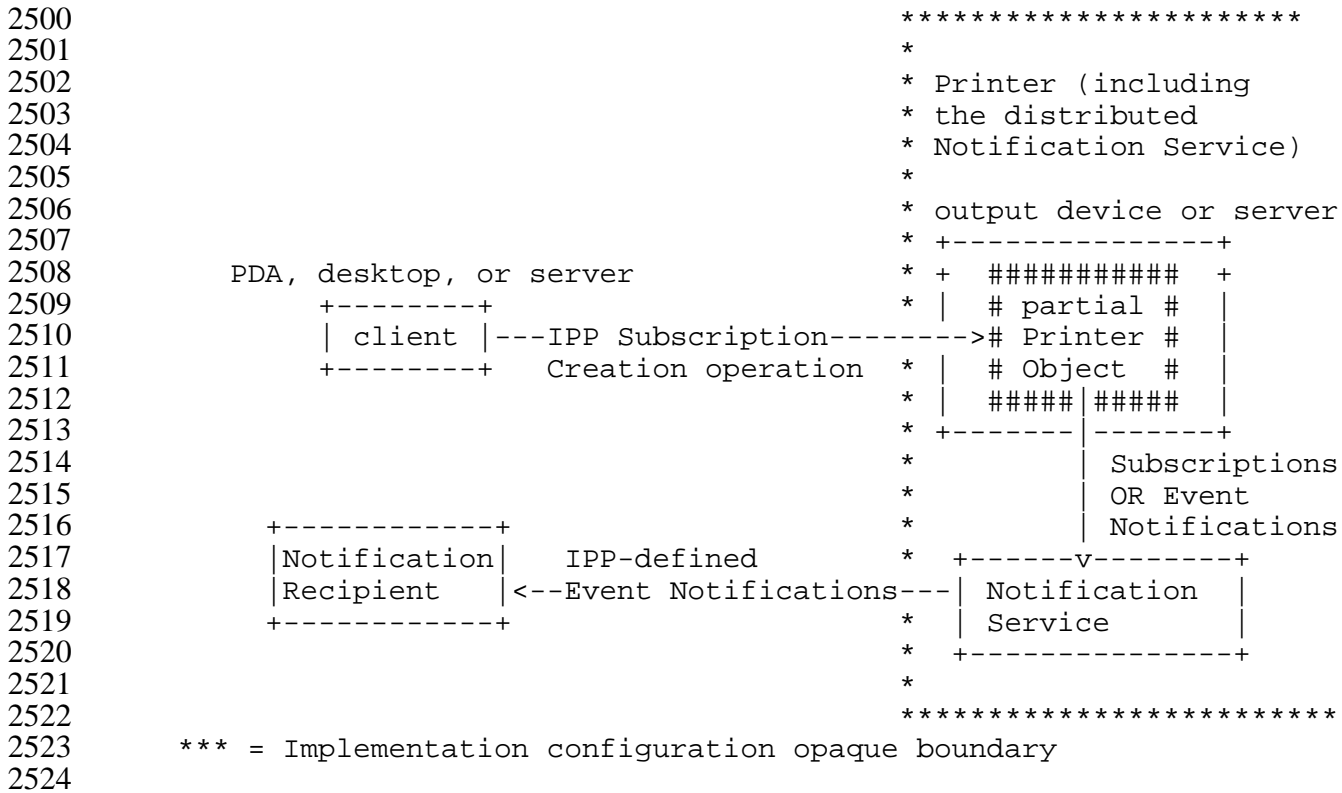


Figure 3 – Opaque Use of a Notification Service Transparent to the Client

C. Appendix - Extended Notification Recipient

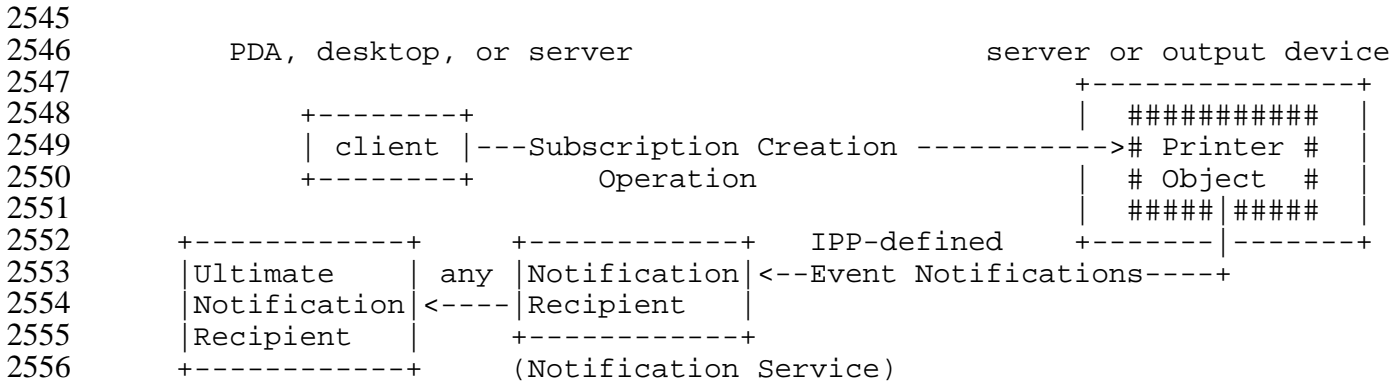
The model allows for an extended Notification Recipient that is itself a notification service that forwards each Event Notification to another recipient (called the Ultimate Notification Recipient in this section). The Delivery Method to the Ultimate Recipient is probably different from the Delivery Method used by the Printer to the extended Notification Recipient.

This extended Notification Recipient is transparent to the Printer but not to the client.

When a client performs a Subscription Creation Operation, it specifies the extended Notification Recipient as it would any Notification Recipient. In addition, the client specifies the Ultimate Notification Recipient in the Subscription Creation Operation in a manner specified by the extended Notification Recipient. Typically, it is either some bytes in the value of “notify-user-data” or some additional parameter in the value of “notify-recipient-uri”. The client also subscribes directly with the extended Notification Recipient (by means outside this document), since it is a notification service in its own right.

The IPP Printer treats the extended Notification Recipient like any other Notification Recipient and the IPP Printer is not aware of the forwarding. The Delivery Method that the extended Notification Recipient uses for delivering the Event Notification to the Ultimate Notification Recipient is beyond the scope of this document and is transparent to the IPP Printer.

2543 Examples of this extended Notification Recipient are paging, immediate messaging services, general
2544 notification services, and NOS vendors' infrastructure. Figure 4 shows this approach.



2557 **Figure 4 – Use of an Extended Notification Recipient transparent to the Printer**

2558 **D. Appendix - Details about Conformance Terminology**

2559 The following paragraphs provide more details about conformance terminology.

2560 **REQUIRED** - an adjective used to indicate that a conforming IPP Printer implementation **MUST**
2561 support the indicated operation, object, attribute, attribute value, status code, or out-of-band value
2562 in requests and responses. See [RFC2911] "Appendix A - Terminology for a definition of
2563 "support". *Since support of this entire Notification specification is OPTIONAL for*
2564 *conformance to IPP/1.1, the use of the term REQUIRED in this document means*
2565 *"REQUIRED if this OPTIONAL Notification specification is implemented".*

2566 **RECOMMENDED** - an adjective used to indicate that a conforming IPP Printer implementation is
2567 recommended to support the indicated operation, object, attribute, attribute value, status code, or
2568 out-of-band value in requests and responses. *Since support of this entire Notification*
2569 *specification is OPTIONAL for conformance to IPP/1.1, the use of the term RECOMMENDED*
2570 *in this document means "RECOMMENDED if this OPTIONAL Notification specification is*
2571 *implemented".*

2572 **OPTIONAL** - an adjective used to indicate that a conforming IPP Printer implementation **MAY**, but is
2573 **NOT REQUIRED** to, support the indicated operation, object, attribute, attribute value, status code,
2574 or out-of-band value in requests and responses.

2575 **E. Appendix - Object Model for Notification**

2576 This section describes the Notification object model that adds a Subscription Object which together
2577 with the Job and Printer object provide the complete Notification semantics.

2578 The object relationships can be seen pictorially as:

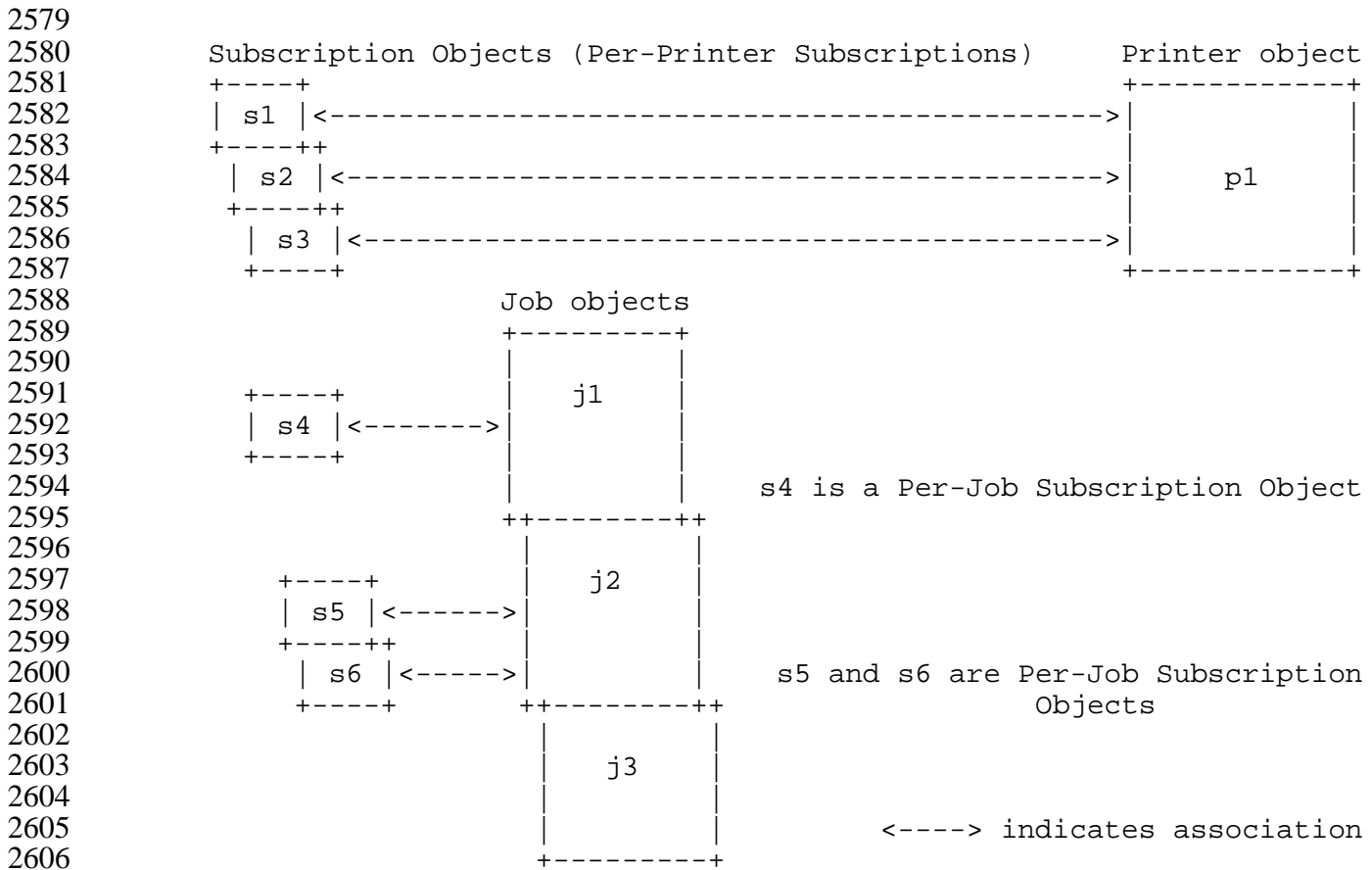


Figure 5 – Object Model for Notification

2608 s1, s2, and s3 are Per-Printer Subscription Objects and can identify Printer and/or Job Events.
2609 s4, s5, and s6 are Per-Job Subscription Objects and can identify Printer and/or Job Events.

2610 **E.1 Appendix - Object relationships**

2611 This sub-section defines the object relationships between the Printer, Job, and Subscription Objects by
2612 example. Whether Per-Printer Subscription Objects are actually contained in a Printer object or are
2613 just bi-directionally associated with them in some way is IMPLEMENTATION DEPENDENT and is
2614 transparent to the client. Similarly, whether Per-Job Subscription Objects are actually contained in a
2615 Job object or are just bi-directionally associated with them in some way is IMPLEMENTATION
2616 DEPENDENT and is transparent to the client. The object relationships are defined as follows:

2617 **E.2 Printer Object and Per-Printer Subscription Objects**

- 2618 1. The Printer object contains (is associated with) zero or more Per-Printer Subscription Objects
2619 (p1 contains s1-s3 Per-Printer Subscription Objects).
- 2620 2. Each Per-Printer Subscription Object (s1, s2, and s3) is contained in (or is associated with)
2621 exactly one Printer object (p1).

2622 **E.3 Job Object and Per-Job Subscription Objects**

- 2623 1. A Job object (j1, j2, j3) is associated with zero or more Per-Job Subscription Objects (s4-s6).
2624 Job j1 is associated with Per-Job Subscription Object s4, Job j2 is associated with Per-Job
2625 Subscription Objects s5 and s6, and Job j3 is not associated with any Per-Job Subscription
2626 Object.
- 2627 2. Each Per-Job Subscription Object is associated with exactly one Job object.

2628 **F. Appendix - Per-Job versus Per-Printer Subscription Objects**

2629 Per-Job and Per-Printer Subscription Objects are quite similar. Either type of Subscription Object can
2630 subscribe to Job Events, Printer Events, or both. Both types of Subscription Objects can be queried
2631 using the Get-Subscriptions and Get-Subscription-Attributes operations and canceled using the Cancel-
2632 Subscription operation. Both types of Subscription Objects create Subscription Objects which have the
2633 same Subscription Object attributes defined. However, there are some semantic differences between
2634 Per-Job Subscription Objects and Per-Printer Subscription Objects. A Per-Job Subscription Object is
2635 established by the client when submitting a job and after creating the job using the Create-Job-
2636 Subscriptions operation by specifying the “job-id” of the Job with the “notify-job-id” attribute. A Per-
2637 Printer Subscription Object is established between a client and a Printer using the Create-Printer-
2638 Subscriptions operation. Some specific differences are:

- 2639 1. A client usually creates one or more Per-Job Subscription Objects as part of the Job Creation
2640 operations (Create-Job, Print-Job, and Print-URI), rather than using the OPTIONAL Create-Job-
2641 Subscriptions operation, especially since Printer implementations NEED NOT support the
2642 Create-Job-Subscriptions operation, since it is OPTIONAL.
- 2643 2. For Per-Job Subscription Objects, the Subscription Object is only valid while the job is “not-
2644 complete” (see sections 5.4.3) while for the Per-Printer Subscription Objects, the Subscription
2645 Object is valid until the time (in seconds) that the Printer returned in the “notify-lease-
2646 expiration-time” operation attribute.
- 2647 3. Job Events in a Per-Job Subscription Object apply only to “one job” (the Job created by the Job
2648 Creation operation or references by the Create-Job-Subscriptions operation) while Job Events in
2649 a Per-Printer Subscription Object apply to ALL jobs contained in the IPP Printer.

2650 **G. Appendix - Description of the base IPP documents**

2651 The base set of IPP documents includes:

- 2652 Design Goals for an Internet Printing Protocol [RFC2567]
2653 Rationale for the Structure and Model and Protocol for the Internet Printing Protocol [RFC2568]
2654 Internet Printing Protocol/1.1: Model and Semantics [RFC2911]
2655 Internet Printing Protocol/1.1: Encoding and Transport [RFC2910]
2656 Internet Printing Protocol/1.1: Implementer's Guide [RFC3196]
2657 Mapping between LPD and IPP Protocols [RFC2569]

2658

2659 The "Design Goals for an Internet Printing Protocol" document takes a broad look at distributed
2660 printing functionality, and it enumerates real-life scenarios that help to clarify the features that need to
2661 be included in a printing protocol for the Internet. It identifies requirements for three types of users:
2662 end users, operators, and administrators. It calls out a subset of end user requirements that are satisfied
2663 in IPP/1.0 [RFC2566, RFC2565]. A few OPTIONAL operator operations have been added to IPP/1.1
2664 [RFC2911, RFC2910].

2665

2666 The "Rationale for the Structure and Model and Protocol for the Internet Printing Protocol" document
2667 describes IPP from a high level view, defines a roadmap for the various documents that form the suite
2668 of IPP specification documents, and gives background and rationale for the IETF IPP working group's
major decisions.

2669

2670 The "Internet Printing Protocol/1.1: Model and Semantics" document describes a simplified model
2671 with abstract objects, their attributes, and their operations. The model introduces a Printer and a Job.
2672 The Job supports multiple documents per Job. The model document also addresses how security,
internationalization, and directory issues are addressed.

2673

2674 The "Internet Printing Protocol/1.1: Encoding and Transport" document is a formal mapping of the
2675 abstract operations and attributes defined in the model document onto HTTP/1.1 [RFC2616]. It also
2676 defines the encoding rules for a new Internet MIME media type called "application/ipp". This
2677 document also defines the rules for transporting over HTTP a message body whose Content-Type is
"application/ipp". This document defines the 'ipp' scheme for identifying IPP printers and jobs.

2678

2679 The "Internet Printing Protocol/1.1: Implementer's Guide" document gives insight and advice to
2680 implementers of IPP clients and IPP objects. It is intended to help them understand IPP/1.1 and some
2681 of the considerations that may assist them in the design of their client and/or IPP object
2682 implementations. For example, a typical order of processing requests is given, including error
checking. Motivation for some of the specification decisions is also included.

2683

2684 The "Mapping between LPD and IPP Protocols" document gives some advice to implementers of
gateways between IPP and LPD (Line Printer Daemon) implementations.

2685

H. Appendix - Full Copyright Statement

2686

Copyright (C) The Internet Society (1998,1999,2000,2001,2002). All Rights Reserved

2687

2688 This document and translations of it may be copied and furnished to others, and derivative works that
2689 comment on or otherwise explain it or assist in its implementation may be prepared, copied, published
2690 and distributed, in whole or in part, without restriction of any kind, provided that the above copyright
2691 notice and this paragraph are included on all such copies and derivative works. However, this
2692 document itself may not be modified in any way, such as by removing the copyright notice or
2693 references to the Internet Society or other Internet organizations, except as needed for the purpose of
2694 developing Internet standards in which case the procedures for copyrights defined in the Internet
Standards process must be followed, or as required to translate it into languages other than English.

2695 The limited permissions granted above are perpetual and will not be revoked by the Internet Society or
2696 its successors or assigns.

2697 This document and the information contained herein is provided on an "AS IS" basis and THE
2698 INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL
2699 WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY
2700 WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY
2701 RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A
2702 PARTICULAR PURPOSE.

2703 **Acknowledgement**

2704
2705 Funding for the RFC Editor function is currently provided by the Internet Society.