

1 INTERNET-DRAFT **There are 7 issues highlighted like this**
2 <draft-ietf-ipp-not-spec-03.txt>

R. Herriot (editor)
Xerox Corporation
T. Hastings
Xerox Corporation
R. deBry
Utah Valley State College
S. Isaacson
Novell, Inc.
J. Martin
Underscore
M. Shepherd
Xerox Corporation
R. Bergman
Hitachi Koki Imaging Solutions
June 30~~29~~, 2000

3
4
5
6
7
8
9
10
11
12
13
14
15
16 Internet Printing Protocol (IPP):
17 **IPP Event Notification Specification**

18
19 Copyright (C) The Internet Society (2000). All Rights Reserved.

20
21 **Status of this Memo**

22 This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of
23 [RFC2026]. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its
24 areas, and its working groups. Note that other groups may also distribute working documents as Internet-
25 Drafts.

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

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

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

31 **Abstract**

32 This document describes an extension to the IPP/1.0, IPP/1.1, and future versions. This extension allows a
33 client to subscribe to printing related Events. Subscriptions are modeled as *Subscription Objects*. The
34 Subscription Object specifies that when one of the specified *Event* occurs, the Printer sends an
35 asynchronous *Event Notification* to the specified *Notification Recipient* via the specified *Delivery Method*
36 (i.e., protocol). A client associates Subscription Objects with a particular Job by performing the Create-
37 Job-Subscriptions operation or by submitting a Job with subscription information. A client associates
38 Subscription Objects with the Printer by performing a Create-Printer-Subscriptions operation. Four other
39 operations are defined for Subscription Objects: Get-Subscriptions-Attributes, Get-Subscriptions, Renew-
40 Subscription, and Cancel-Subscription.

41

42 The full set of IPP documents includes:

43 Design Goals for an Internet Printing Protocol [RFC2567]

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

45 Internet Printing Protocol/1.1: Model and Semantics [IPP-MOD]

46 Internet Printing Protocol/1.1: Encoding and Transport [IPP-PRO]

47 Internet Printing Protocol/1.1: Implementer's Guide [IPP-IIG]

48 Mapping between LPD and IPP Protocols [RFC2569]

49 The "Design Goals for an Internet Printing Protocol" document takes a broad look at distributed printing
50 functionality, and it enumerates real-life scenarios that help to clarify the features that need to be included
51 in a printing protocol for the Internet. It identifies requirements for three types of users: end users,
52 Operators, and Administrators. It calls out a subset of end user requirements that are satisfied in IPP/1.0.
53 Operator and Administrator requirements are out of scope for version 1.0. A few OPTIONAL Operator
54 operations have been added to IPP/1.1.

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

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

62 The "Internet Printing Protocol/1.1: Encoding and Transport" document is a formal mapping of the abstract
63 operations and attributes defined in the model document onto HTTP/1.1. It defines the encoding rules for a
64 new Internet MIME media type called "application/ipp". This document also defines the rules for
65 transporting over HTTP a message body whose Content-Type is "application/ipp". This document defines
66 a new scheme named 'ipp' for identifying IPP printers and jobs. Finally, this document defines
67 interoperability rules for supporting IPP/1.0 clients.

68 The "Internet Printing Protocol/1.1: Implementer's Guide" document gives insight and advice to
69 implementers of IPP clients and IPP objects. It is intended to help them understand IPP/1.0 and some of the
70 considerations that may assist them in the design of their client and/or IPP object implementations. For
71 example, a typical order of processing requests is given, including error checking. Motivation for some of
72 the specification decisions is also included.

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

75

76

Table of Contents

77	1	Introduction.....	7
78	1.1	Notification Overview	7
79	2	Models for Notification	9
80	2.1	Model for Notification (Simple Case).....	9
81	2.2	Model for Notification with Cascading Printers.....	10
82	2.3	Distributed Model for Notification.....	10
83	2.4	Extended Notification Recipient	10
84	3	Terminology.....	11
85	3.1	Conformance Terminology.....	11
86	3.2	Other Terminology	11
87	4	Object Relationships.....	13
88	4.1	Printer and Per-Printer Subscription Objects.....	13
89	4.2	Printer, Job and Per-Job Subscription Objects	13
90	5	Subscription Object	13
91	5.1	Rules for Support of Subscription Template Attributes	14
92	5.2	Rules for Processing Subscription Template Attributes	15
93	5.3	Subscription Template Attributes.....	18
94	5.3.1	notify-recipient-uri (uri).....	19
95	5.3.2	notify-events (1setOf type2 keyword)	19
96	5.3.3	notify-attributes (1setOf type2 keyword).....	24
97	5.3.4	notify-user-data (octetString(63)).....	25
98	5.3.5	notify-charset (charset)	26
99	5.3.6	notify-natural-language (naturalLanguage)	26
100	5.3.7	notify-lease-duration (integer(0:67108863))	26
101	5.3.8	notify-persistence (boolean)	27
102	5.4	Subscription Description Attributes	28
103	5.4.1	notify-subscription-id (integer (1:MAX)).....	29
104	5.4.2	notify-sequence-number (integer (0:MAX)).....	29
105	5.4.3	notify-lease-expiration-time (integer(0:MAX)).....	30
106	5.4.4	notify-printer-up-time (integer(1:MAX))	30
107	5.4.5	notify-printer-uri (uri)	30
108	5.4.6	notify-job-id (integer(1:MAX))	31
109	5.4.7	notify-subscriber-user-name (name(MAX)).....	31
110	6	Printer Description Attributes Related to Notification	31
111	6.1	notify-max-printer-subscriptions-supported (integer(0:MAX))	32
112	6.2	notify-max-job-subscriptions-supported (integer(0:MAX)).....	32
113	6.3	printer-state-change-time (integer(1:MAX))	33

114	6.4 printer-state-change-date-time (dateTime)	33
115	7 New Values for Existing Printer Description Attributes	33
116	7.1 operations-supported (1setOf type2 enum).....	33
117	8 Attributes Only in Event Notifications	34
118	8.1 notify-subscribed-event (type2 keyword)	34
119	8.2 notify-text (text(MAX)).....	34
120	9 Event Notification Content.....	35
121	9.1 Content of Machine Consumable Event Notifications	36
122	9.1.1 Attributes in Event Notification Content Common to All Events.....	36
123	9.1.2 Additional Attributes in Event Notification Content for Job Events.....	37
124	9.1.3 Additional Attributes in Event Notification Content for Printer Events	38
125	9.2 Content of Human Consumable Event Notification.....	38
126	9.2.1 Information in Event Notification Content Common to All Events.....	39
127	9.2.2 Additional Information in Event Notification Content for Job Events.....	40
128	9.2.3 Additional Information in Event Notification Content for Printer Events	41
129	10 Delivery Methods	42
130	11 Operations for Notification.....	43
131	11.1 Subscription Creation Operations.....	43
132	11.1.1 Create-Job-Subscriptions Operation.....	43
133	11.1.2 Create-Printer-Subscriptions operation	45
134	11.1.3 Job Creation Operation – Extensions for Notification	46
135	11.2 Other Operations.....	48
136	11.2.1 Validate-Job Operation - Extensions for Notification	48
137	11.2.2 Get-Printer-Attributes - Extensions for Notification	48
138	11.2.3 Get-Subscription-Attributes operation.....	49
139	11.2.4 Get-Subscriptions operation	51
140	11.2.5 Renew-Subscription operation.....	53
141	11.2.6 Cancel-Subscription operation.....	55
142	12 Conformance Requirements	57
143	13 IANA Considerations	58
144	13.1 Format and Requirements for IPP Delivery Method Registration Proposals	58
145	14 Internationalization Considerations	59
146	15 Security Considerations	59
147	16 Status Codes	60
148	16.1 successful-ok-ignored-subscriptions (0x0003).....	60
149	16.2 client-error-ignored-all-subscriptions (0x0414)	60

150	17	Status Codes in Subscription Attributes Groups	60
151	17.1	client-error-uri-scheme-not-supported (0x040C)	60
152	17.2	client-error-too-many-subscriptions (0x0415).....	61
153	17.3	successful-ok-too-many-events (0x0005).....	61
154	17.4	successful-ok-ignored-or-substituted-attributes (0x0001).....	61
155	18	Encodings of Additional Attribute Tags.....	61
156	19	References.....	61
157	20	Author's Addresses.....	62
158	A.	Appendix - Model for Notification with Cascading Printers.....	64
159	B.	Appendix - Distributed Model for Notification.....	65
160	C.	Appendix - Extended Notification Recipient	66
161	D.	Appendix - Details about Conformance Terminology	67
162	E.	Appendix - Object Model for Notification	67
163	E.1	Appendix - Object relationships	68
164	E.2	Printer Object and Per-Printer Subscription Objects	68
165	E.3	Job Object and Per-Job Subscription Objects.....	69
166	F.	Appendix - Per-Job versus Per-Printer Subscription Objects.....	69
167	G.	Appendix: Change History (to be removed for Internet-Draft)	69
168	G.1	Changes to the May 10, 2000 version to create the June 30, 2000 version	69
169	G.2	Changes to the March 8, 2000 version to create the May 10, 2000 version.....	71
170	G.3	Changes to the March 6, 2000 version to create the March 8, 2000 version.....	71
171	G.4	Changes to the February 2, 2000 version to create the March 6, 2000 version.....	71
172	G.5	Changes to the October 14, 1999 version to create the February 2, 2000 version	73
173	H.	Appendix: Full Copyright Statement.....	75
174			
175		Tables	
176		Table 1 – Subscription Template Attributes.....	18
177		Table 2 – Subscription Description Attributes	28
178		Table 3 – Printer Description Attributes Associated with Notification.....	32
179		Table 4 – Operation-id assignments	33
180		Table 5 – Attributes in Event Notification Content.....	37
181		Table 6 – Additional Attributes in Event Notification Content for Job Events	38
182		Table 7 – Combinations of Events and Subscribed Events for “job-impressions-completed”	38
183		Table 8 – Additional Attributes in Event Notification Content for Printer Events	38
184		Table 9 – Printer Name in Event Notification Content	40

185 Table 10 – Event Name in Event Notification Content..... 40

186 Table 11 – Event Time in Event Notification Content..... 40

187 Table 12 – Job Name in Event Notification Content for Job Events 40

188 Table 13 – Job State in Event Notification Content for Job Events..... 41

189 Table 14 – Printer State in Event Notification Content for Printer Events 41

190 Table 15 – Conformance Requirements for Operations 57

191 **Figures**

192 Figure 1 – Model for Notification 10

193 Figure 2 – Model for Notification with Cascading Printers 65

194 Figure 3 – Opaque Use of a Notification Service Transparent to the Client 66

195 Figure 4 – Use of an Extended Notification Recipient transparent to the Printer 67

196 Figure 5 – Object Model for Notification..... 68

197

198 1 Introduction

199 This IPP notification specification is an extension to IPP/1.0 [RFC2568, RFC2569] and IPP/1.1 [ipp-mod,
200 ipp-pro]. This document in combination with the following documents is intended to meet the notification
201 requirements described in [ipp-not-req]:

202 Internet Printing Protocol (IPP): “Job Progress Attributes” [ipp-prog]
203 One or more Delivery Method ~~definition documents~~Documents registered with IANA (see section
204 13).
205

206 Note: this document does not define any Delivery Methods, but it does define the rules for conformance for
207 Delivery Method Documents.

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

209 1.1 Notification Overview

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

215 When a client (called a *Subscribing Client*) performs an operation that creates a Subscription Object, the
216 operation contains one or more *Subscription Template Attributes Groups*. Each such group holds
217 information used by the Printer to initialize a newly created Subscription Object. The Printer creates one
218 Subscription Object for each Subscription Template Attributes Group in the operation. This group is like
219 the Job Template Attributes group defined in [ipp-mod]. The following is an example of the information
220 included in a Subscription Template Attributes Group (see section 5 for details on the Subscription Object
221 attributes):

- 222 1. The names of Subscribed Events that are of interest to the Notification Recipient.
- 223 2. The address (URL) of one Notification Recipient.
- 224 3. The Delivery Method (i.e., the protocol) which the Printer uses to send the Event Notification.
- 225 4. Some opaque data that the Printer sends to the Notification Recipient in the Event Notification. The
226 Notification Recipient might use this opaque data as a forwarding address for the Event
227 Notification.
- 228 5. The charset to use in text fields within an Event Notification
- 229 6. The natural language to use in the text fields of the Event Notification
- 230 7. The requested lease time in seconds for the Subscription Object

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

- 233 • **Job Creation operation:** When a client performs such an operation (Print-Job, Print-URI, and
234 Create-Job), a client can include zero or more Subscription Template Attributes Groups in the
235 request. The Printer creates one Subscription Object for each Subscription Template Attributes
236 Group in the request, and the Printer associates each such Subscription Object with the newly
237 created Job. This document extends these operations' definitions in [ipp-mod] by adding
238 Subscription Template Attributes Groups in the request and Subscription Attributes Groups in the
239 response.
- 240 • **Create-Job-Subscriptions operation:** A client can include one or more Subscription Template
241 Attributes Groups in the request. The Printer creates one Subscription Object for each Subscription
242 Template Attributes Group and associates each with the job that is the target of this operation.
- 243 • **Create-Printer-Subscriptions operation:** A client can include one or more Subscription Template
244 Attributes Groups in the request. The Printer creates one Subscription Object for each Subscription
245 Template Attributes Group and associates each with the Printer that is the target of this operation.
- 246 For each of the above operations:
- 247 • the Printer associates a Subscription Object with the Printer or a specific Job. When a Subscription
248 Object is associated with a Job Object, it is called a *Per-Job Subscription Object*. When a
249 Subscription Object is associated with a Printer Object, it is called a *Per-Printer Subscription*
250 *Object*.
- 251 • the response contains one Subscription Attributes Group for each Subscription Template Attributes
252 Group in the request and in the same order. When the Printer successfully creates a Subscription
253 Object, its corresponding Subscription Attributes Group contains the "notify-subscription-id"
254 attribute. This attribute uniquely identifies the Subscription Object and is analogous to a "job-id" for
255 a Job object. Some operations described below use the "notify-subscription-id" to identify the target
256 Subscription Object.
- 257 This document adds the following additional operations (see section 11.2 for further details)::
- 258 • **Validate-Job operation:** When a client performs this operation, a client can include zero or more
259 Subscription Template Attributes Groups in the request. The Printer determines if it could create
260 one Subscription Object for each Subscription Template Attributes Group in the request. This
261 document extends this operation's definition in [ipp-mod] by adding Subscription Template
262 Attributes Groups in the request and Subscription Attributes Groups in the response.
- 263 • **Get-Printer-Attributes operation:** This document extends this operation's definition in [ipp-mod]
264 by adding: Subscription Template Attributes, Printer Description Attributes, attributes to existing
265 group names, and new group names.
- 266 • **Get-Subscription-Attributes operation:** This operation allows a client to obtain the specified
267 attributes of a target Subscription Object.

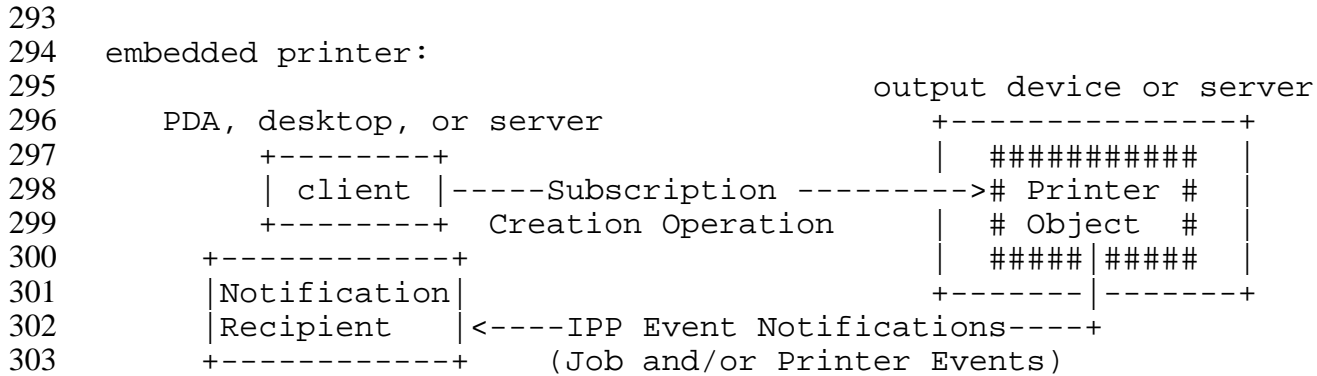
- 268 • **Get-Subscriptions operation:** This operation allows a client to obtain the specified attributes of all
269 Subscription Objects associated with the Printer or a specified Job.
- 270 • **Renew-Subscription operation:** This operation renews the lease on the target Per-Printer
271 Subscription Object before it expires. A newly created Per-Printer Subscription Object receives an
272 initial lease. It is the duty of the client to use this operation frequently enough to preserve a Per-
273 Printer Subscription Object. The Printer deletes a Per-Printer Subscription Object when its lease
274 expires. A Per-Job Subscription Object last exactly as long as its associated Job Object and thus
275 doesn't have a lease.
- 276 • **Cancel-Subscription operation:** This operation cancels the lease on the specified Per-Printer
277 Subscription Object and thereby deletes the Subscription Object.
- 278 When an Event occurs, the Printer finds all Subscription Objects listening for the Event (see section 9 for
279 details on finding such Subscription Objects). For each such Subscription Object, the Printer:
- 280 a) generates an Event Notification with information specified in section 9, AND
- 281 b) either:
- 282 i) delivers the Event Notification using the Delivery Method and target address identified in the
283 Subscription Object's "notify-recipient-uri" attribute if the Delivery Method is a "push", OR
- 284 ii) saves Event Notification for a time period defined by the Delivery Method if the Delivery
285 Method is a "pull", i.e., the Notification Recipient is expected to fetch the Event Notifications.

286 **2 Models for Notification**

287 **2.1 Model for Notification (Simple Case)**

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

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



304 **Figure 1 – Model for Notification**

305 2.2 Model for Notification with Cascading Printers

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

- 309 • directly to the Notification Recipient specified by the Subscribing Client or
- 310 • via the Print Server to the Notification Recipient specified by the Subscribing Client.

311 See Appendix A for more details.

312 2.3 Distributed Model for Notification

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

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

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

322 Appendix B describes this example in detail.

323 2.4 Extended Notification Recipient

324 The model allows for an extended Notification Recipient that is itself a Notification service that forwards
 325 each Event Notification to another recipient. The client contacts this Notification Recipient to arrange for

326 forwarding by means outside the scope of this document. The Printer need not be aware that the
327 Notification Recipient forwards Event Notifications.

328 Appendix C describes this example in detail.

329 **3 Terminology**

330 This section defines terminology used throughout this document.

331 **3.1 Conformance Terminology**

332 Capitalized terms, such as **MUST**, **MUST NOT**, **REQUIRED**, **SHOULD**, **SHOULD NOT**, **MAY**,
333 **NEED NOT**, and **OPTIONAL**, have special meaning relating to conformance to this specification.
334 These terms are defined in [ipp-mod section 13.1 on conformance terminology, most of which is
335 taken from RFC 2119 [RFC2119]. See Appendix D for complete details.

336 **READ-ONLY** - an adjective used in an attribute definition to indicate that an IPP Printer **MUST NOT**
337 allow the attribute's value to be modified with the Set-Job-Attributes or Set-Printer-Attributes
338 operations (see [ipp-set]). Note: there is no Set-Subscription operation so this term is not used for
339 Subscription object attributes.

340 **3.2 Other Terminology**

341 **Administrator** - A human user who establishes policy for and configures the print system.

342 **Operator** - A human user who carries out the policy established by the Administrator and controls the
343 day to day running of the print system.

344 **IPP Client (or client)** - The software component (PDA, desktop, or server) that performs an IPP
345 operation directed at an IPP Printer (server or output device).

346 **Job Creation operation** - One of the operations that creates a Job object: Print-Job, Print-URI and
347 Create-Job. The Validate-Job operation is not a Job Creation operation because no Job object is
348 created. Therefore, when a statement also applies to the Validate-Job operation, it is mentioned
349 explicitly.

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

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

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

- 357 **Subscribed Event** – an Event that the Subscribing Client expresses interest in by making it a value of
358 the “notify-events” attribute on a Subscription Object.
- 359 **Subscribed Job Event** – a Subscribed Event that is a Job Event.
- 360 **Subscribed Printer Event** – a Subscribed Event that is a Printer Event.
- 361 **Event Notification** - the information about an Event that the Printer sends when an Event occurs.
- 362 **Notification Recipient** - the entity to which the Printer sends an Event Notification.
- 363 **Delivery Method** - the mechanism by which the Printer delivers the Event Notification, e.g., via email
364 or via SNMP.
- 365 **Delivery Method Document** - a document, separate from this document, that defines a Delivery
366 Method.
- 367 **Subscription Object** - An object containing a set of attributes that indicate: the Notification Recipient,
368 the Delivery Method, the Subscribed Events that cause the Printer to send an Event Notification,
369 and the information to send in an Event Notification.
- 370 **Per-Job Subscription Object** - A Subscription Object that is associated with a single Job. The Create-
371 Job-Subscriptions operation and Job Creation operations create such an object.
- 372 **Per-Printer Subscription Object** - A Subscription Object that is associated with the Printer as a
373 whole. The Create-Printer-Subscriptions operation creates such an object.
- 374 **Subscribing Client** - The client that creates the Subscription Object.
- 375 **Subscription Creation Operation** - An operation that creates a Subscription Object: Job Creation
376 operations, Create-Job-Subscriptions operation, and Create-Printer-Subscriptions operation. In the
377 context of a Job Creation operation, a Subscription Creation Operation is the part of the Job
378 Creation operation that creates a Subscription object.
- 379 **Subscription Creation Request** – The request portion of a Subscription Creation Operation.
- 380 **Subscription Template Attributes** – Subscription Object attributes that a client can supply in a
381 Subscription Creation Operation and associated Printer Object attributes that specify supported and
382 default values for the Subscription Object attributes.
- 383 **Subscription Description Attributes** – Subscription Object attributes that a Printer supplies during a
384 Subscription Creation Operation.
- 385 **Subscription Template Attributes Group** – The attributes group in a request that contains
386 Subscription Object attributes that are Subscription Template Attributes.

387 **Subscription Attributes Group** – The attributes group in a response that contains Subscription Object
388 attributes.

389 **Human Consumable Event Notification** – localized text for human consumption only. There is no
390 standardized format and thus programs should not try to parse this text.

391 **Machine Consumable Event Notification** - bytes for program consumption. The bytes are formatted
392 according to the Delivery Method document.

393 **Printer** – the software that supports an output device or print server (see IPP/1.1 [ipp-mod] which uses
394 the terms Printer and Printer object interchangeably). This document extends the IPP/1.1 Printer
395 definition to include the software that implements Subscription Creation Operations and the sending
396 of Event Notifications, even if the software for such a Printer would be distributed across a network
397 (see section 2.3).

398 **Notification** – when not in the phrases ‘Event Notification’ and ‘Notification Recipient’ — the
399 concepts of this specification, i.e., Events, Subscription Objects, and Event Notifications.

400 **4 Object Relationships**

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

403 **4.1 Printer and Per-Printer Subscription Objects**

404 1. A Printer object can be associated with zero or more Per-Printer Subscription Objects.

405 2. Each Per-Printer Subscription Object is associated with exactly one Printer object.

406 **4.2 Printer, Job and Per-Job Subscription Objects**

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

408 2. Each Job object is associated with exactly one Printer object.

409 3. A Job object is associated with zero or more Per-Job Subscription Objects.

410 4. Each Per-Job Subscription Object is associated with exactly one Job object.

411 **5 Subscription Object**

412 A Subscribing Client creates a Subscription Object with a Subscription Creation Operation in order to
413 indicate its interest in certain Events. See section 11 for a description of these operations. When an Event

414 occurs, the Subscription Object specifies to the Printer where to send Event Notifications, how to send them
415 and what to put in them. See section 9 for details on the contents of an Event Notification.

416 Using the IPP Job Template attributes as a model (see [ipp-mod] section 4.2), the attributes of a
417 Subscription Object are divided into two categories: Subscription Template Attributes and Subscription
418 Description Attributes.

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

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

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

425 **5.1 Rules for Support of Subscription Template Attributes**

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

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

- 430 1. Each attribute's name starts with the prefix string "notify-" and this document calls such attributes
431 "notify-xxx".
- 432 2. For each "notify-xxx" Subscription Object attribute defined in column 1 of Table 1, Table 1
433 specifies corresponding Printer attributes: "notify-xxx-default", "notify-xxx-supported", "yyy-
434 supported" and "notify-max-xxx-supported" defined in column 2 of Table 1.
- 435 3. If a Printer supports "notify-xxx" in column 1 of Table 1, then the Printer MUST support all
436 associated attributes specified in column 2 of Table 1. For example, Table 1 shows that if the Printer
437 supports "notify-events", it MUST support "notify-events-default", "notify-events-supported" and
438 "notify-max-events-supported".
- 439 4. If a Printer does not support "notify-xxx" in column 1 of Table 1, then the Printer MUST NOT
440 support any associated "notify-yyy" attributes specified in column 2 of Table 1. For example, Table
441 1 shows that if the Printer doesn't support "notify-events", it MUST NOT support "notify-events-
442 default", "notify-events-supported" and "notify-max-events-supported". Note this rule does not
443 apply to attributes whose names do not start with the string "notify-" and are thus defined in another
444 object and used by other attributes.
- 445 5. Most "notify-xxx" attributes have a corresponding "yyy-supported" attribute that specifies the
446 supported values for "notify-xxx". Column 2 of Table 1 specifies the name of each "yyy-supported"

447 attribute. The naming rules of IPP/1.1 (see [ipp-mod]) are used when “yyy-supported” is “notify-
448 xxx-supported”.

449 6. Some “notify-xxx” attributes have a corresponding “notify-xxx-default” attribute that specifies the
450 value for “notify-xxx” if the client does not supply it. Column 2 of Table 1 specifies the name of
451 each “notify-xxx-default” attribute. The naming rules of IPP/1.1 (see [ipp-mod]) are used.

452 ~~7. A Printer MUST support the ‘subscription-template’ group name in the Get-Printer-Attributes
453 operation (see [ipp-mod] section 3.2.5 for the specification of group names) and return all of the
454 attributes in column 2 of Table 1 that the Printer supports. All of the attributes in column 2 of Table
455 1 MUST be included if the client supplies the ‘all’ group name.~~

456 If a client wishes to present an end user with a list of supported values from which to choose, the client
457 SHOULD query the Printer for its supported value attributes. The client SHOULD also query the default
458 value attributes. If the client then limits selectable values to only those values that are supported, the client
459 can guarantee that the values supplied by the client in the create request all fall within the set of supported
460 values at the Printer. When querying the Printer, the client MAY enumerate each attribute by name in the
461 Get-Printer-Attributes Request, or the client MAY just supply the ‘subscription-template’ group name in
462 order to get the complete set of supported attributes (both supported and default attributes).

463 5.2 Rules for Processing Subscription Template Attributes

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

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

- 471 1. If a client supplies a “notify-xxx” attribute from column 1 of Table 1 and the Printer supports it and
472 its value, the Printer MUST populate the attribute on the created Subscription Object.
- 473 2. If a client supplies a “notify-xxx” attribute from column 1 of Table 1 and the Printer doesn’t support
474 it or its value, the Printer MUST NOT populate the attribute on the created Subscription Object with
475 it. The Printer MUST do one of the following:
 - 476 a) If the value of the “notify-xxx” attribute is unsupported, the Printer MUST return the attribute
477 with its value in the Subscription Attributes Group of the response.
 - 478 b) If “notify-xxx” is an unsupported attribute, the Printer MUST return the attribute in the
479 Subscription Attributes Group of the response with the ‘unsupported’ out-of-band value.

480 Note: The rules of this step are the same as for Unsupported Attributes [ipp-mod] section 3.1.7.
481 except that the unsupported attributes are returned in the Subscription Attributes Group rather than

- 482 the Unsupported Attributes Group because Subscription Creation Operations can create more than
483 one Subscription Object).
- 484 3. If a client is REQUIRED to supply a “notify-xxx” attribute from column 1 of Table 1 and the
485 Printer doesn’t support the supplied value, the Printer MUST NOT create a Subscription Object.
486 The rules for Unsupported Attributes in step #2 still apply.
- 487 4. If a client does not supply a “notify-xxx” attribute from column 1 of Table 1 and the attribute is
488 REQUIRED for the client to supply, the Printer MUST reject the Subscription Creation Operation
489 (including Job Creation operations) without creating a Subscription Object, and MUST return in the
490 response:
- 491 c) the status code ‘client-error-bad-request’ AND
- 492 d) no Subscription Attribute Groups.
- 493 5. If a client does not supply a “notify-xxx” attribute from column 1 of Table 1 that is OPTIONAL for
494 the client to supply, and column 2 of Table 1 either:
- 495 a) specifies a “notify-xxx-default” attribute, the Printer MUST behave as if the client had supplied
496 the “notify-xxx-default” attribute (see step #1) and populate the Subscription object with the
497 value of the “notify-xxx-default” attribute as part of the Subscription Creation operation (unlike
498 Job Template attributes where the Printer does not populate the Job object with defaults - see
499 [ipp-mod]) OR
- 500 b) does not specify a “notify-xxx-default” attribute, the Printer MUST populate the “notify-xxx”
501 attribute on the Subscription Object according to the definition of the “notify-xxx” attribute in a
502 section 5.3. For some attributes, the “notify-xxx” is populated with the value of some other
503 attribute, and for others, the “notify-xxx” is NOT populated on the Subscription object at all.
- 504 6. A Printer MUST create a Subscription Object for each Subscription Template Attributes group in a
505 request unless the Printer:
- 506 a) encounters some attributes in a Subscription Template Attributes Group that require the Printer
507 not to create the Subscription Object OR
- 508 b) would be a Per-Job Subscription Object and the number of Per-Job Subscription Objects already
509 equals the value of the “notify-max-job-subscriptions-supported” Printer attribute OR
- 510 c) would be a Per-Printer Subscription Object and the number of Per-Printer Subscription Objects
511 already equals the value of the “notify-max-printer-subscriptions-supported” Printer attribute.
- 512 7. A response MUST contain one Subscription Attributes Group for each Subscription Template
513 Attributes Group in the request (and in the same order) whether the Printer creates a Subscription
514 Object from the Subscription Template Attributes Group or not. However, the attributes in each
515 Subscription Attributes Group can be in any order.

- 516 8. The Printer MUST populate each Subscription Attributes Group of the response such that each
517 contains:
- 518 a) the “notify-subscription-id” attribute (see section 5.4.1), if and only if the Printer creates a
519 Subscription Object.
- 520 b) the “notify-lease-duration” attribute (see section 5.3.7), if and only if the Printer creates a Per-
521 Printer Subscription Object. The value of this attribute is the value of the Subscription Object’s
522 “notify-lease-duration” attribute. This value MAY be different from the client-supplied value
523 (see section 5.3.7). If a client supplies this attribute in the creation of a Per-Job Subscription
524 Object, it MUST appear in this group with the out-of-band value ‘unsupported’ to indicate that
525 the Printer doesn’t support it in this context.
- 526 c) all of the unsupported Subscription Template Attributes from step #2.
- 527 d) the “notify-status-code” attribute if the Printer does not create the Subscription Object or if there
528 are unsupported attributes from step #2. The possible values of the “notify-status-code” attribute
529 are shown below (see section 17 for more details). The Printer returns the first value in the list
530 below that describes the status.
- 531 ‘client-error-uri-scheme-not-supported’: the Subscription Object was not created because
532 the scheme of the “notify-recipient-uri” attribute is not supported. See section 17.1 for
533 more details about this status code. See step #3 in this section for the case that causes
534 this error, and the resulting step #6a) that causes the Printer not to create the Subscription
535 Object.
- 536 ‘client-error-too-many-subscriptions’: the Subscription Object was not created because the
537 number of Subscription Objects would exceed the value of the Printer’s “notify-max-job-
538 subscriptions-supported” or “notify-max-printer-subscriptions-supported” attributes.
539 The client SHOULD try again later. See section 17.2 for more details about this status
540 code. See steps #6b) and #6c) in this section for the cases that causes this error.
- 541 ‘successful-ok-too-many-events’: the Subscription Object was created without the “notify-
542 events” values included in this Subscription Attributes Group because the “notify-
543 events” attribute contains too many values. See section 17.3 for more details about this
544 status code. See step #2 in this section and section 5.3.2 for the cases that cause this
545 status code.
- 546 ‘successful-ok-ignored-or-substituted-attributes’: the Subscription Object was created but
547 some supplied Subscription Template Attributes are unsupported. These unsupported
548 attributes are also in the Subscription Attributes Group. See section 17.4 for more details
549 about this status code. See step #2 in this section for the cases that cause this status code.
- 550 9. The Printer MUST validate all Subscription Template Attributes and MUST return all unsupported
551 attributes and values in the corresponding Subscription Attributes Group of the response (see step
552 #2) unless it determines that it could not create additional Subscription Objects because of condition
553 #6b) or condition #6c). Then, the Printer NEED NOT validate these additional Subscription

554 Template Attributes and the client MUST NOT expect to find unsupported attributes from step #2
 555 in such additional Subscription Attribute Groups.

556 5.3 Subscription Template Attributes

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

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

- 559 • **Attribute in Subscription Object:** the name and attribute syntax of each Subscription Object
 560 Attribute that is a Subscription Template Attribute
- 561 • **Default and Supported Printer Attributes:** the default attribute and supported Printer attributes
 562 that are associated with the attribute in column 1.

563 A Printer MUST support all attributes in Table 1 below except for “notify-attributes” (and “notify-
 564 attributes-supported”). A client MUST supply “notify-recipient-uri” and MAY omit any of the rest of the
 565 attributes in column 1 of Table 1 in a Subscription Creation Request.

566 **Table 1 – Subscription Template Attributes**

Attribute in Subscription Object	Default and Supported Printer Attributes
notify-recipient-uri (uri)	notify-schemes-supported (1setOf uriScheme)
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-languages (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-persistence (boolean)	notify-persistence-default (boolean)

Attribute in Subscription Object	Default and Supported Printer Attributes
	notify-persistence-supported (1setOf boolean)

567 **5.3.1 notify-recipient-uri (uri)**

568 This attribute's value is a URL, which is a special case of a URI. Its value consists of a scheme and an
 569 address. The address specifies the Notification Recipient and the scheme specifies the Delivery Method for
 570 each Event Notification associated with this Subscription Object.

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

572 A client **MUST** supply this attribute in Subscription Creation Operation. Thus there is no need for a default
 573 attribute.

574 The “notify-schemes-supported (1setOf uriScheme)” attribute **MUST** specify the schemes supported for
 575 this attribute.

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

579 **5.3.2 notify-events (1setOf type2 keyword)**

580 This attribute contains a set of Subscribed Events. When an Event occurs and it “matches” a value of this
 581 attribute, the Printer sends an Event Notification using information in the Subscription Object. The details
 582 of “matching” are described subsection 5.3.2.2.

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

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

587 Each value of this attribute on a Subscription Object **MUST** be one of the values of the “notify-events-
 588 supported (1setOf type2 keyword)” attribute.

589 The number of values of this attribute **MUST NOT** exceed the value of the “notify-max-events-supported”
 590 attribute. A Printer **MUST** support at least 2 values per Subscription Object. If the number of values
 591 supplied by a client in a Subscription Creation Operation exceeds the value of this attribute, the Printer
 592 **MUST** treat extra values as unsupported values and **MUST** use the value of ‘successful-ok-too-many-
 593 events’ for the “notify-status-code” attribute in the Subscription Attributes Group of the response.

594 **ISSUE 01: OK that we changed the number from 5 to 2 because we have rearranged the categories of**
 595 **Events to have group events?**

596 **5.3.2.1 Standard Values for Subscribed Events**

597 Each value of this attribute is a keyword and it specifies a Subscribed Event that represents certain changes.
598 Some keywords represent a subset of changes of another keyword, e.g., 'job-completed' is an Event value
599 which is a sub-value of 'job-state-change'. See section 5.3.2.2 for the case where this attribute contains both
600 a value and a sub-value.

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

602 A Printer **MUST** support the Events indicated as "REQUIRED" and **MAY** support the Events indicated as
603 "OPTIONAL".

604 **5.3.2.1.1 No Events**

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

606 **'none'**: REQUIRED - no Event Notifications for any Events. As the sole value of "notify-events-
607 supported", this value means that the Printer does not support the sending of Event Notifications. As
608 the sole value of "notify-events-default", this value means that a client **MUST** specify the "notify-
609 events" attribute in order for a Subscription Creation Operation to succeed. If the Printer receives
610 this value as the sole value of a Subscription Creation Operation, it does not create a Subscription
611 Object. If a Printer receives this value with other values of a Subscription Creation Operation, the
612 Printer **MUST** treat this value as an unsupported value.

613 **5.3.2.1.2 Subscribed Printer Events**

614 For a Printer, the first Printer Event **MUST** be 'printer-restarted' and the last Printer Event **MUST** be
615 'printer-shutdown'.

616 The standard keyword values for Subscribed Printer Events are:

617 **'printer-state-changed'**: REQUIRED - the Printer changed state from any state to any other state.
618 Specifically, the value of the Printer's "printer-state", "printer-state-reasons" or "printer-is-
619 accepting-jobs" attributes changed.

620
621 This Subscribed Event value has the following sub-values: 'printer-restarted' and 'printer-
622 shutdown'. A client can listen for any of these sub-values if it doesn't want to listen to all printer-
623 state changes:

624 **'printer-restarted'**: OPTIONAL - when the printer is powered up or the Restart-Printer operation
625 is performed (see [ipp-set2]). This event is the first Printer Event that can be received from a
626 Printer.

627 **'printer-shutdown'**: OPTIONAL - when the device is being powered down or the Shutdown-
628 Printer operation has been performed (see [ipp-set2]). This event is the last Printer Event that
629 can be received from a Printer.

630 **'printer-config-changed'**: OPTIONAL - when the configuration of a Printer has changed, i.e., the
631 value of the "printer-message-from-operator" or any "configuration" Printer attribute has changed.

632 A “configuration” Printer attribute is an attribute which can change value because of some human
633 interaction either direct or indirect, and which is not covered by one of the other Events in this
634 section. Examples of “configuration” Printer attributes are any of the Job Template attributes, such
635 as “xxx-supported”, “xxx-ready” and “xxx-default”. Often, such a change is the result of a client
636 performing a Set-Printer-Attributes operation (see [ipp-set]) on the Printer. The client has to
637 perform a Get-Printer-Attributes to find out the new values of these changed attributes. This Event
638 is useful for GUI clients and drivers to update the available printer capabilities to the user.

639
640 This Event value has the following sub-values: ‘printer-media-changed’ and ‘printer-finishings-
641 changed’. A client can listen for any of these sub-values if it doesn’t want to listen to all printer-
642 configuration changes:

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

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

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

656 **‘printer-no-longer-full’**: OPTIONAL - when the Printer has just become able to accept a Job
657 Creation operation, Send-Document operation, or Send-URI operation. A Printer sends this Event
658 when it has acquired more buffer space to accept jobs after it previously did not have room to accept
659 any more jobs and would have rejected a Job Creation Operation, a Send-Document operation, or
660 Send-URI operation. A Notification Recipient listens for this Event when there is more than one
661 client feeding a printer/server (fan-in).

662 **‘printer-full’**: OPTIONAL - when the Printer has just become unable to accept a Job Creation
663 operation, Send-Document operation, or Send-URI operation due to lack of buffer space. It is
664 intended that a Notification Recipient use this Event to stop whatever the ‘printer-no-longer-full’
665 Event starts.

666 **ISSUE 02: OK to add ‘printer-full’ Event?**

667 **‘printer-almost-idle’**: OPTIONAL - when the Printer needs another Job in order to stay busy. A
668 Printer that is an output device MAY use this Event to request a new job sufficiently ahead of time
669 so as not to run out of work between jobs. A Printer that is a fan-out spooler MAY listen for this

670 Event and hold pending Jobs until a downstream Printer sends this Event to indicate that it needs
671 another Job in order to stay busy.

672 **‘printer-not-almost-idle’**: OPTIONAL - when the Printer no-longer needs another Job in order to stay
673 busy. It is intended that a Notification Recipient use this Event to stop whatever the ‘printer-almost-
674 idle’ Event starts.

675 **ISSUE 03: OK to add ‘printer-not-almost-idle’ Event?**

676 **5.3.2.1.3 Subscribed Job Events**

677 For each Job object, the first Job Event MUST be ‘job-created’ and the last Job Event MUST be ‘job-
678 completed’.

679 The standard keyword values for Subscribed Job Events are:

680 **‘job-state-changed’**: REQUIRED - the job has changed from any state to any other state. Specifically,
681 the Printer sends this Event whenever the value of the “job-state” attribute or “job-state-reasons”
682 attribute changes. When a Job is removed from the Job History (see [ipp-mod] 4.3.7.1), no Event is
683 generated.

684
685 This Event value has the following sub-values: ‘job-created’, ‘job-completed’ and ‘job-purged’. A
686 client can listen for any of these sub-values if it doesn’t want to listen to all ‘job-state changes’.

687 **‘job-created’**: REQUIRED - the Printer has accepted a Job Creation operation and the job’s “time-
688 at-creation” attribute value is set (see [ipp-mod] section 4.3.14.1). The Printer puts the job in
689 the ‘pending’, ‘pending-held’ or ‘processing’ states. This event is the first Job Event that can be
690 received from a Job.

691 **‘job-completed’**: REQUIRED - the job has reached one of the completed states, i.e., the value of
692 the job’s “job-state” attribute has changed to: ‘completed’, ‘aborted’, or ‘canceled’. The Job’s
693 “time-at-completed” and “date-time-at-completed” (if supported) attributes are set (see [ipp-
694 mod] section 4.3.14). This event is the last Job Event that can be received from a Job.

695 **‘job-purged’**: OPTIONAL - when a ‘not-completed’ job (i.e., not ‘completed’, ‘canceled’, or
696 ‘aborted’) was purged from the printer using the Purge-Jobs operation. The Printer MUST
697 immediately send a ‘job-completed’ event after this event to meet the requirement that ‘job-
698 completed’ is the last event for the Job.

699 **‘job-config-changed’**: OPTIONAL - when the configuration of a job has changed, i.e., the value of
700 the “job-message-from-operator” or any of the “configuration” Job attributes have changed. A
701 “configuration” Job attribute is an attribute that can change value because of some human
702 interaction either direct or indirect. Examples of “configuration” Job attributes are any of the job
703 template attributes and the “job-name” attribute. Often, such a change is the result of the user or the
704 Operator performing a Set-Job-Attributes operation (see [ipp-set]) on the Job object. The client
705 performs a Get-Job-Attributes to find out the new values of the changed attributes. This Event is
706 useful for GUI clients and drivers to update the job information to the user.

707 '**job-progress**': OPTIONAL – an impression, sheet, or copy has completed. See the separate [ipp-
708 prog] specification.

709 **5.3.2.2 Rules for Matching of Subscribed Events**

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

713 **5.3.2.2.1 Rules for Matching of Printer Events**

714 Suppose that the Printer causes Printer Event E to occur. For each Per-Job or Per-Printer Subscription S in
715 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 S, the
716 Printer MUST generate an Event Notification.

717 Consider the example. There are three Subscription Objects each with the Subscribed Printer Event
718 ‘printer-state-changed’. Subscription Object A is a Per-Printer Subscription Object. Subscription Object
719 B is a Per-Job Subscription Object for Job 1, and Subscription Object C is a Per-Job Subscription
720 Object for Job 2. When the Printer enters the ‘stopped’ state, the Printer sends an Event Notification to
721 the Notification Recipients of Subscription Objects A, B, and C because this is a Printer Event. Note if
722 Job 1 has already completed, the Printer would not send an Event Notification for its Subscription
723 Object.

724 **5.3.2.2.2 Rules for Matching of Job Events**

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

- 726 3. For each Per-Printer Subscription S in the Printer, if E equals a value of this attribute in S or E is a
727 sub-value of a value of this attribute in S, the Printer MUST generate an Event Notification.
- 728 4. For each Per-Job Subscription S associated with Job J, if E equals a value of this attribute in S or E
729 is a sub-value of a value of this attribute in S, the Printer MUST generate an Event Notification.
- 730 5. For each Per-Job Subscription S that is NOT associated Job J, if E equals a value of this attribute in
731 S or E is a sub-value of a value of this attribute in, the Printer MUST NOT generate an Event
732 Notification from S.

733 Consider the example: There are three Subscription Objects listening for the Job Event ‘job-completed’.
734 Subscription Object A is a Per-Printer Subscription Object. Subscription Object B is a Per-Job
735 Subscription Object for Job 1, and Subscription Object C is a Per-Job Subscription Object for Job 2. In
736 addition, Per-Printer Subscription Object D is listening for the Job Event ‘job-state-changed’. When Job
737 1 completes, the Printer sends an Event Notification to the Notification Recipient of Subscription
738 Object A (because it is Per-Printer) and Subscription Object B because it is a Per-Job Subscription
739 Object associated with the Job generating the Event. The Printer also sends an Event Notification to the
740 Notification Recipient of Subscription Object D because ‘job-completed’ is a sub-value of ‘job-state-
741 changed’ – the value that Subscription Object D is listening for. The Printer does not send an Event
742 Notification to the Notification Recipients of Subscription Object C because it is a Per-Job Subscription
743 Object associated with some Job other than the Job generating the Event.

744 **5.3.2.2.3 Special Cases for Matching Rules**

745 This section contains rule for special cases.

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

752 If an Event matches two values of this “notify-events” attribute in a single Subscription object (e.g., a value
753 and its sub-value), a Printer **MAY** send one Event Notification for each matched value in the Subscription
754 Object or it **MAY** send only one Event Notification per Subscription Object. The rules in sections 5.3.2.2.1
755 and 5.3.2.2.2 are purposefully ambiguous about the number of Event Notification sent when Event E
756 matches two or more values in a Subscription Object.

757 Consider the example: There are two Per-Printer Subscription Objects when a Job completes.
758 Subscription Object A has the Subscribed Job Event ‘job-state-changed’. Subscription Object B has the
759 Subscribed Job Events ‘job-state-changed’ and ‘job-completed’. The Printer sends an Event
760 Notification to the Notification Recipient of Subscription Object A with the value of ‘job-state-
761 changed’ for the “notify-subscribing-event” attribute. The Printer sends either one or two Event
762 Notifications to the Notification Recipient of Subscription Object B, depending on implementation. If it
763 sends two Event Notifications, one has the value of ‘job-state-changed’ for the “notify-subscribing-
764 event” attribute, and the other has the value of ‘job-completed’ for the “notify-subscribing-event”
765 attribute. If it sends one Event Notification, it has the value of either ‘job-state-changed’ or ‘job-
766 completed’ for the “notify-subscribing-event” attribute, depending on implementation. The algorithm
767 for choosing such a value is implementation dependent.

768 In addition, Delivery Methods **MAY** allow the Printer to moderate certain high frequency events (see
769 section 9).

770 **5.3.3 notify-attributes (1setOf type2 keyword)**

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

774 A Printer **MAY** support this attribute.

775 A client **MAY** supply this attribute in a Subscription Creation Operation. If the client does not supply this
776 attribute in Subscription Creation Operation or the Printer does not support this attribute, the Subscription
777 Object **MUST NOT** contain the “notify-attributes” attribute. There is no “notify-attributes-default”
778 attribute.

779 Each keyword value of this attribute on a Subscription Object **MUST** be a value of the “notify-attributes-
780 supported (1setOf type2 keyword)” attribute. The “notify-attributes-supported” **MAY** contain any Printer

781 attribute, Job attribute or Subscription Object attribute that the Printer supports in an Event Notification. It
782 MUST NOT contain any of the attributes in Section 9.1 that a Printer automatically puts in an Event
783 Notification; it would be redundant. If a client supplies an attribute in Section 9.1, the Printer MUST treat it
784 as an unsupported attribute value of the “notify-attributes” attribute.

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

- 787 a) a Subscription attribute, the Printer MUST use the attribute N in the Subscription Object that is
788 being used to generate the Event Notification.
- 789 b) a Job attribute and the Printer is generating an Event Notification from a Per-Job Subscription
790 Object S, the Printer MUST use the attribute N in the Job object associated with S.
- 791 c) a Job attribute and the Printer is generating an Event Notification from a Per-Printer Subscription
792 Object and the Event is:
- 793 • a Job Event, the Printer MUST use the attribute N in the Job object that caused the Event.
 - 794 • a Printer Event, the Printer MUST use the attribute N in the active Job.

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

- 797 a) the attributes specified in section 9.1 and
798 b) each attribute named by this attribute.

799 **5.3.4 notify-user-data (octetString(63))**

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

- 802 • the identity of the Subscriber
- 803 • a path or index to some Subscriber information
- 804 • a key that identifies to the Notification Recipient the ultimate recipient of the Event Notification
- 805 • the id for a Notification Recipient that had previously registered with an Instant Messaging Service

806 A Printer MUST support this attribute.

807 A client MAY supply this attribute in a Subscription Creation Operation. If the client does not supply this
808 attribute in Subscription Creation Operation, the Subscription Object MUST NOT contain the “notify-user-
809 data” attribute. There is no “notify-user-data-default” attribute.

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

812 **5.3.5 notify-charset (charset)**

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

815 A Printer MUST support this attribute.

816 A client MAY supply this attribute in a Subscription Creation Operation. If the client does not supply this
817 attribute in Subscription Creation Operation or supplies an unsupported value, the Printer MUST populate
818 this attribute in the Subscription Object with the value of the “attributes-charset” operation attribute, which
819 is a REQUIRED attribute in all IPP requests (see [ipp-mod]). If the value of the “attributes-charset”
820 attribute is unsupported, the Printer MUST populate this attribute in the Subscription Object with the value
821 of the Printer’s “charset-configured” attribute. There is no “notify-charset-default” attribute.

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

824 **5.3.6 notify-natural-language (naturalLanguage)**

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

828 A Printer MUST support this attribute.

829 A client MAY supply this attribute in a Subscription Creation Operation. If the client does not supply this
830 attribute in Subscription Creation Operation or supplies an unsupported value, the Printer MUST populate
831 this attribute in the Subscription Object with the value of the “attributes-natural-language” operation
832 attribute, which is a REQUIRED attribute in all IPP requests (see [ipp-mod]). If the value of the “attributes-
833 natural-language” attribute is unsupported, the Printer MUST populate this attribute in the Subscription
834 Object with the value of the Printer’s “natural-language-configured” attribute. There is no “notify-natural-
835 language-default” attribute.

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

838 **5.3.7 notify-lease-duration (integer(0:67108863))**

839 This attribute specifies the duration of the lease associated with the Per-Printer Subscription Object at the
840 time the Subscription Object was created or the lease was renewed. The duration of the lease is infinite if
841 the value is 0, i.e., the lease never expires.

842 This attribute is not present on a Per-Job Subscription Object because the Subscription Object lasts exactly
843 as long as the associated Job object. See section 5.4.3 on “notify-lease-expiration-time (integer(0:MAX))”
844 for more details.

845 A Printer MUST support this attribute.

846 For a Subscription Object Creation operation of a Per-Job Subscription Object, the client MUST NOT
847 supply this attribute. If the client does supply this attribute, the Printer MUST treat it as an unsupported
848 attribute.

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

855 After the Printer has populated this attribute with a supported value, the value represents the “granted
856 duration” of the lease and the Printer sets the value of the Subscription Object’s “notify-lease-expiration-
857 time” attribute as specified in section 5.4.3.

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

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

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

866 **5.3.8 notify-persistence (boolean)**

867 This attribute specifies whether the Printer preserves the Subscription Object across power cycles.

868 A Printer MUST support this attribute.

869 A client MAY supply this attribute in a Subscription Creation Operation. If the client does not supply this
870 attribute in Subscription Creation Operation, the Printer MUST populate this attribute with its “notify-
871 persistence-default” (boolean) attribute value. If the client supplies this attribute with an unsupported value,
872 the Printer MUST populate this attribute with a supported value. The Printer MAY populate this attribute
873 with a value other than the one the client requests. For example, if the client specifies ‘true’ and the Printer
874 doesn’t have space for another Subscription Object, it sets the value of this attribute to ‘false’. If the client
875 specifies ‘false’ and the Printer has a policy of setting this attribute to ‘true’ if there is space, the Printer sets
876 this attribute to ‘true’.

877 The value of this attribute on a Subscription Object MUST be a value of the “notify-persistence-supported
878 (1setOf boolean)” attribute. The “notify-persistence-supported” (1setOf boolean) attribute can have one of
879 the following three values:

- 880 • true: all Subscription Objects are persistent (if there is space).
- 881 • false: no Subscription Objects are persistent
- 882 • true, false: some Subscription Objects are persistent and others are not. For example, the Printer
883 may have room for only 2 Subscription Objects.

884 It is RECOMMENDED that all Subscription Objects be persistent. If Jobs are persistent, the Per-Job
885 Subscription Objects MUST be persistent too.

886 **ISSUE 04:** it would be better for this attribute to be a Subscription Description attribute that the Printer sets
887 to show whether the Object is persistent or not. Agree?

888 **5.4 Subscription Description Attributes**

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

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

892 A client MUST NOT supply the attributes in Table 2 in a Subscription [Template Attributes Group of a](#)
893 [Subscription](#) Creation Operation. If the client supplies them, the Printer MUST NOT set them and MUST
894 treat them as unsupported attributes. There are no corresponding default or supported attributes.

895 **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))

Subscription Object attributes:
notify-subscriber-user-name (name(MAX))

896 5.4.1 notify-subscription-id (integer (1:MAX))

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

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

900 The Printer **SHOULD NOT** assign the value of this attribute sequentially as it creates Subscription Objects.
901 Sequential assignment makes it easy for rogue clients to guess the value of this attribute on other
902 Subscription Objects.

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

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

908 5.4.2 notify-sequence-number (integer (0:MAX))

909 The value of this attribute indicates the number of times that the Printer has generated and attempted to
910 send an Event Notification. When an Event Notification contains this attribute, the Notification Recipient
911 can determine whether it missed some Event Notifications (i.e., numbers skipped) or received duplicates
912 (i.e., same number twice).

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

914 When the Printer creates a Subscription Object, it **MUST** set the value of this attribute to 0. This value
915 indicates that the Printer has not sent any Event Notifications for this Subscription Object.

916 Each time the Printer sends a newly generated Event Notification, it **MUST** increase the value of this
917 attribute by 1. For some Delivery Methods, the Printer **MUST** include this attribute in each Event
918 Notification, and the value **MUST** be the value after it is increased by 1. That is, the value of this attribute
919 in the first Event Notification after Subscription object creation **MUST** be 1, the second **MUST** be 2, etc. If
920 a Delivery Method is defined such that the Notification Recipient returns a response, the Printer can re-try
921 sending an Event Notification a certain number of times with the same sequence number when the
922 Notification Recipient fails to return a response.

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

925 **5.4.3 notify-lease-expiration-time (integer(0:MAX))**

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

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

930 When the Printer creates a Per-Job Subscription Object, this attribute **MUST NOT** be present – the
931 Subscription Object lasts exactly as long as the associated Job object.

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

937 When the Printer powers up, it **MUST** set the value of this attribute in each persistent Subscription Object
938 using the algorithm in the previous paragraph.

939 When the “printer-up-time” equals the value of this attribute, the Printer **MUST** delete the Subscription
940 Object. A client can extend a lease of a Per-Printer Subscription Object with the Renew-Subscription
941 operation (see section 11.2.5).

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

945 **5.4.4 notify-printer-up-time (integer(1:MAX))**

946 This attribute is an alias for the Printer’s “printer-up-time” attribute “ (see [ipp-mod] section 4.4.29).

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

948 When the Printer creates a Per-Job Subscription Object, this attribute **MUST NOT** be present. When the
949 Printer creates a Per-Printer Subscription Object, this attribute **MUST** be present.

950 Note: this attribute exists in a Per-Printer Subscription Object so that a client using the Get-Subscription-
951 Attributes or Get-Subscription operations can convert the Per-Printer Subscription’s “notify-lease-
952 expiration-time” attribute to wall clock time with one request. If the value of the “notify-lease-expiration-
953 time” attribute is not 0 (i.e., no expiration time), then the difference between the “notify-lease-expiration-
954 time” attribute and the “notify-printer-up-time” is the remaining number of seconds on the lease from the
955 current time.

956 **5.4.5 notify-printer-uri (uri)**

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

958 A Printer MUST support this attribute.

959 During a Subscription Creation Operation, the Printer MUST populate this attribute with the value of the
960 “printer-uri” operation attribute in the request. From the Printer URI, the client can, for example, determine
961 what security scheme was used.

962 **5.4.6 notify-job-id (integer(1:MAX))**

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

965 A Printer MUST support this attribute.

966 If this attribute is not present, the Subscription Object MUST be a Per-Printer Subscription. If this attribute
967 is present, the Subscription Object MUST be a Per-Job Subscription Object and this attribute MUST
968 identify the Job with which the Subscription Object is associated.

969 Note: This attribute could be useful to a Notification Recipient that receives an Event Notification
970 generated from a Per-Job Subscription Object and caused by a Printer Event. The Event Notification gives
971 access to the Printer and the Subscription Object. The Event Notification gives access to the associated Job
972 only via this attribute. **ISSUE 05:** OK that we added the REQUIRED “notify-job-id” attribute because it is
973 needed for a Notification Recipient client to determine from a random subscription-id whether a
974 Subscription is Per-Printer or Per-Job and if the latter which Job.

975 **5.4.7 notify-subscriber-user-name (name(MAX))**

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

977 A Printer MUST support this attribute.

978 The Printer sets this attribute to the most authenticated printable name that it can obtain from the
979 authentication service over which the Subscription Creation Operation was received. The Printer uses the
980 same mechanism for determining the value of this attribute as it does for a Job’s “job-originating-user-
981 name” (see [ipp-mod] section 4.3.6).

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

985 **6 Printer Description Attributes Related to Notification**

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

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

Printer object attributes:	REQUIRED	READ-ONLY
notify-max-printer-subscriptions-supported (integer(0:MAX))	Yes	No
notify-max-job-subscriptions-supported (integer(0:MAX))	Yes	No
printer-state-change-time (integer(1:MAX))	No	Yes
printer-state-change-date-time (dateTime)	No	Yes

990 **6.1 notify-max-printer-subscriptions-supported (integer(0:MAX))**

991 This attribute specifies the maximum number of un-expired Per-Printer Subscription Objects that the
992 Printer supports at one time. A value of MAX indicates no effective maximum.

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

994 A Printer **MUST** support at least 1 Per-Printer Subscription Object. An implementation **MAY** allow an
995 Administrator to set the value of this attribute to 0 in order to disable creation of Per-Printer Subscription
996 Objects.

997 If the number of Per-Printer Subscription Objects equals the value of this attribute during a Subscription
998 Creation Operation, the Printer **MUST NOT** create any additional Per-Printer Subscription Objects. See
999 section 11.1.2 for details on the creation of Subscription Objects and how the Printer indicates such failure
1000 in a Subscription Creation Operation.

1001 **ISSUE 06: OK to use MAX to mean no limit and 0 to mean that an admin has turned off subscriptions?**

1002 **6.2 notify-max-job-subscriptions-supported (integer(0:MAX))**

1003 This attribute specifies the maximum number of Per-Job Subscription Objects that the Printer supports for
1004 each job. For example, if a Printer can hold 2 Jobs and this attribute has the value of 3, it can hold a total of
1005 6 Per-Job Subscription Objects. A value of MAX indicates no effective maximum.

1006 A Printer **MUST** support this attribute.

1007 A Printer **MUST** support at least 1 Per-Job Subscription Object per Job. An implementation **MAY** allow
1008 an Administrator to set the value of this attribute to 0 in order to disable creation of Per-Job Subscription
1009 Objects.

1010 If the number of Per-Job Subscription Objects associated with the specified Job equals the value of this
1011 attribute during a Subscription Creation Operation, the Printer **MUST NOT** create any additional Per-Job
1012 Subscription Objects. See section 11.1 for details on the creation of Subscription Objects and how the
1013 Printer indicates such failure in a Subscription Creation Operation.

1014 **ISSUE 07: OK to use MAX to mean no limit and 0 to mean that an admin has turned off subscriptions?**

1015 **6.3 printer-state-change-time (integer(1:MAX))**

1016 This attribute records the most recent time at which the ‘printer-state-changed’ Printer Event occurred
 1017 whether or not any Subscription objects were listening for this event. This attribute helps a client or
 1018 operator to determine how long the Printer has been in its current state.

1019 A Printer MAY support this attribute and if so, the attribute MUST be READ-ONLY.

1020 On power-up, the Printer MUST set the value of this attribute to be the value of its “printer-up-time”
 1021 attribute, so that it always has a value. Whenever the ‘printer-state-changed’ Printer Event occurs, the
 1022 Printer MUST set this attribute to the value of the Printer’s “printer-up-time” attribute.

1023 **6.4 printer-state-change-date-time (dateTime)**

1024 This attribute records the most recent time at which the ‘printer-state-changed’ Printer Event occurred
 1025 whether or not there were any Subscription Objects listening for this event. This attribute helps a client or
 1026 operator to determine how long the Printer has been in its current state.

1027 A Printer MAY support this attribute and if so, the attribute MUST be READ-ONLY.

1028 On power-up, the Printer MUST set the value of this attribute to be the value of its “printer-current-time”
 1029 attribute, so that it always has a value (see [ipp-mod] section 4.4.30 on “printer-current-time”). Whenever
 1030 the ‘printer-state-changed’ Printer Event occurs, the Printer MUST set this attribute to the value of the
 1031 Printer’s “printer-current-time” attribute.

1032 **7 New Values for Existing Printer Description Attributes**

1033 **7.1 operations-supported (1setOf type2 enum)**

1034 The following “operation-id” values are added in order to support the new operations defined in this
 1035 document:

1036

Table 4 – Operation-id assignments

Value	Operation Name
0x0016	Create-Printer-Subscriptions
0x0017	Create-Job-Subscriptions
0x0018	Get-Subscription-Attributes

Value	Operation Name
0x0019	Get-Subscriptions
0x001A	Renew-Subscription
0x001B	Cancel-Subscription

1037 **8 Attributes Only in Event Notifications**

1038 This section contains those attributes that exist only in Event Notifications.

1039 **8.1 notify-subscribed-event (type2 keyword)**

1040 This attribute indicates the Subscribed Event that caused the Printer to send this Event Notification. This
1041 attribute exists only in Event Notifications.

1042 The Printer **MUST** send this attribute. This attribute exists only in Event Notifications.

1043 This attribute **MUST** contain one of the values of the “notify-events” attribute in the Subscription Object,
1044 i.e., one of the Subscribed Event values. Its value is the Subscribed Event that “matches” the Event that
1045 caused the Printer to send this Event Notification. This Subscribed Event value may be identical to the
1046 Event or the Event may be a sub-value of the Subscribed Event. For example, the ‘job-completed’ Event
1047 (which is a sub-event of the ‘job-state-changed’ event) would cause the Printer to send an Event
1048 Notification for either the ‘job-completed’ or ‘job-state-changed’ Subscribed Events and to send the ‘job-
1049 completed’ or ‘job-state-changed’ value for this attribute, respectively,. See section 5.3.2.2 for the
1050 “matching” rules of Subscribed Events and for additional examples.

1051 The Delivery Method Document specifies whether the Printer includes the value of this attribute in an
1052 Event Notification.

1053 **8.2 notify-text (text(MAX))**

1054 This attribute contains a Human Consumable text message (see section 9.2). This message describes the
1055 Event and is encoded as plain text, i.e., ‘text/plain’ with the charset specified by Subscription Object’s
1056 “notify-charset” attribute.

1057 The Delivery Method Document specifies whether the Printer includes this attribute in an Event
1058 Notification.

1059 The Printer **MAY** support this attribute. If a Printer supports a Delivery Method that requires this attribute,
1060 then the Printer **MUST** support this attribute

1061 **9 Event Notification Content**

1062 This section defines the Event Notification content that the Printer sends when an Event occurs.

1063 When an Event occurs, the Printer MUST find each Subscription object whose “notify-events” attribute
1064 “matches” the Event. See section 5.3.2.2 for details on “matching”. For each matched Subscription Object,
1065 the Printer MUST create an Event Notification with the content and format that the Delivery Method
1066 Document specifies. The content contains the value of attributes specified by the Delivery Method
1067 Document. The Printer obtains the values immediately after the Event occurs. For example, if the “printer-
1068 state” attribute changes from ‘idle’ to ‘processing’, the Event ‘printer-state-changed’ occurs and the Printer
1069 puts various attributes into the Event Notification, including “printer-up-time” and “printer-state” with the
1070 values that they have immediately after the Event occurs, i.e., the value of “printer-state” is ‘processing’.

1071 If two different Events occur simultaneously, or nearly so (e.g., “printer-up-time” has the same value for
1072 both), the Printer MUST create a separate Event Notification for each Event, even if the associated
1073 Subscription Object is the same for both Events. For example, suppose that two nearly-simultaneously
1074 Events represent two successive ‘printer-state-changed’ Events, one from ‘idle’ to ‘processing’ and another
1075 from ‘processing’ to ‘stopped’. These two Events have the same name but are different instances of the
1076 Event. Then the Printer MUST create a separate Event Notification for each Event and SHOULD accurately
1077 report the “printer-state” of the first Event as ‘processing’ and the second Event as ‘stopped’.

1078 If the same Event occurs several times in quick succession (e.g., ‘job-progress’), the Printer MUST create a
1079 separate Event Notification for each Event unless the Delivery Method Document specifies that the Event is
1080 moderated. Events might be moderated by a time interval (e.g., every 10 seconds) or by the number of
1081 Events (every 10th occurrence of the Event).

1082 If a Subscription Object contains more than one Subscribed Event, and several matching Events occur in
1083 quick succession, the Printer MUST generate a separate Event Notification for each Event. Depending on
1084 the Delivery Method, the Printer MAY combine several Event Notifications into a single compound Event
1085 Notification.

1086 After the Printer has created the Event Notification, the Printer delivers it via either a:

1087 Push Delivery Method: The Printer sends the Event Notification shortly after an Event occurs. For
1088 some Push Delivery Methods, the Notification Recipient MUST send a response; for others it
1089 MUST NOT send a response.

1090 Pull Delivery Method: The Printer saves Event Notifications for some event-lease time and expects
1091 the Notification Recipient to request Event Notifications. The Printer returns the Event Notifications
1092 in a response to such a request.

1093 The next two sections describe the values that a Printer sends in the content of Machine Consumable and
1094 Human Consumable Event Notifications, respectively.

1095 **9.1 Content of Machine Consumable Event Notifications**

1096 This section defines the attributes that a Delivery Method **MUST** mention in a Delivery Method Document
1097 when specifying the Machine Consumable Event Notification's contents.

1098 This document does not define the order of attributes in Event Notifications. However, Delivery Method
1099 Documents **MAY** define the order of some or all of the attributes.

1100 A Delivery Method Document **MUST** specify additional attributes (if any) that a Printer implementation
1101 sends in a Machine Consumable Event Notification.

1102 Notification Recipients **MUST** be able to accept Event Notifications containing attributes they do not
1103 recognize. What a Notification Recipient does with an unrecognized attribute is implementation-
1104 dependent. Notification Recipients **MAY** attempt to display unrecognized attributes anyway or **MAY**
1105 ignore them.

1106 The next three sections define the attributes in Event Notification Contents that are:

1107 a) for all Events

1108 b) for Job Events only

1109 c) for Printer Events only

1110 **9.1.1 Attributes in Event Notification Content Common to All Events**

1111 This section lists the attributes that a Delivery Method **MUST** specify for all Events.

1112 The tables in this section and following sections contain the following columns:

1113 a) **Source Value:** the name of the attribute that supplies the value for the Event Notification.
1114 Asterisks in this field refer to a note below the table.

1115 b) **Sends:** if the Printer supports the value (column 1) on the Source Object (column 3) the
1116 Delivery Method **MUST** specify:

1117 **MUST:** that the Printer **MUST** send the value.

1118 **SHOULD:** either that the Printer **MUST** send the value or that the value is incompatible
1119 with the Delivery Method.

1120 **MAY:** that the Printer **MUST**, **SHOULD**, **MAY**, **MUST NOT**, **SHOULD NOT**, or **NEED**
1121 **NOT** send the value.

1122 c) **Source Object:** the object from which the source value comes. If the object is "Event
1123 Notification", the Printer fabricates the value when it sends the Event Notification. See section
1124 8.

1125 Table 5 lists potential values in each Event Notification.

1126 **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

1127 * If the Subscription Object does not contain a “notify-user-data” attribute and the Delivery Method
 1128 document REQUIRES the Printer to send the “notify-user-data” source value in the Event Notification, the
 1129 Printer MUST send an octet-string of length 0.

1130 ** The last three rows represent additional attributes that a client MAY request via the “notify-attributes”
 1131 attribute. A Printer MAY support the “notify-attributes” attribute. The Delivery Method MUST say that the
 1132 Printer MUST, SHOULD, MAY, MUST NOT, SHOULD NOT, or NEED NOT support the “notify-
 1133 attributes” attribute and specific values of this attribute. The Delivery Method MAY say that support for the
 1134 “notify-attributes” is conditioned on support of the attribute by the Printer or it MAY say that Printer
 1135 MUST support the “notify-attribute” attribute if the Printer supports the Delivery Method.

1136 **9.1.2 Additional Attributes in Event Notification Content for Job Events**

1137 This section lists the additional attributes that a Delivery Method MUST specify for Job Events. See Table
 1138 6.

1139 **Table 6 – Additional Attributes in Event Notification Content for Job Events**

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

1140 * The Printer MUST send the “job-impressions-completed” attribute in an Event Notification only for the
 1141 combinations of Events and Subscribed Events shown in Table 7.

1142 **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’

1143

1144 9.1.3 Additional Attributes in Event Notification Content for Printer Events

1145 This section lists the additional attributes that a Delivery Method MUST specify for Printer Events. See
 1146 Table 8.

1147 **Table 8 – Additional Attributes in Event Notification Content for Printer Events**

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

1148 9.2 Content of Human Consumable Event Notification

1149 This section defines the information that a Delivery Method MUST mention in a Delivery Method
 1150 Document when specifying the Human Consumable Event Notifications contents or the value of the
 1151 “notify-text” attribute.

- 1152 Such a Delivery Method **MUST** specify the following information and a Printer **SHOULD** send it:
- 1153 a) the Printer name (see Table 9)
 - 1154 b) the time of the Event (see Table 11)
 - 1155 c) for Printer Events only:
 - 1156 i) the Event (see Table 10) and/or Printer state information (see Table 14)
 - 1157 d) for Job Events only:
 - 1158 i) the job identity (see Table 12)
 - 1159 ii) the Event (see Table 10) and/or Job state information (see Table 13)
- 1160 The subsections of this section specify the attributes that a Printer **MUST** use to obtain this information.
- 1161 A Delivery Method Document **MUST** specify additional information (if any) that a Printer implementation
1162 sends in a Human Consumable Event Notification or in the “notify-text” attribute.
- 1163 A client **MUST NOT** request additional attributes via the “notify-attributes” attribute because this attribute
1164 works only for Machine Consumable Event Notifications.
- 1165 Notification Recipients **MUST NOT** expect to be able to parse the Human Consumable Event Notification
1166 contents or the value of the “notify-text” attribute.
- 1167 The next three sections define the attributes in Event Notification Contents that are:
- 1168 a) for all Events
 - 1169 b) for Job Events only
 - 1170 c) for Printer Events only
- 1171 **9.2.1 Information in Event Notification Content Common to All Events**
- 1172 This section lists the source of the information that a Delivery Method **MUST** specify for all Events.
- 1173 There is a separate table for each piece of information. Each row in the table represents a source value for
1174 the information and the values are listed in order of preference, with the first one being the preferred one.
1175 An implementation **SHOULD** use the source value from the earliest row in each table. The tables in this
1176 section and following contain the following columns for each piece of information:
- 1177 a) **Source of Value:** the name of the attribute that supplies the value for the Event Notification
 - 1178 b) **Source Object:** the object from which the source value comes.
- 1179 The tables in this section do not contain a “Sends” column because all rows would have a “**SHOULD**” as
1180 defined in section 9.1.1.
- 1181 Table 9 lists the source of the information for the Printer Name. The “printer-name” is more user-friendly
1182 unless the Notification Recipient is in a place where the Printer name is not meaningful.

1183 **Table 9 – Printer Name in Event Notification Content**

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

1184

1185 Table 10 lists the source of the information for the Event name. A Printer MAY combine this information
 1186 with state information described for Jobs in Table 13 or for Printers in Table 14.

1187 **Table 10 – Event Name in Event Notification Content**

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

1188

1189 Table 11 lists the source of the information for the time that the Event occurred. A Printer can send this
 1190 value only if it supports the Printer’s “printer-current-time” attribute. If a Printer does not support the
 1191 “printer-current-time” attribute, it MUST NOT send the “printer-up-time” value instead, since it is not an
 1192 allowed option for human consumable information.

1193 **Table 11 – Event Time in Event Notification Content**

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

1194

1195 **9.2.2 Additional Information in Event Notification Content for Job Events**

1196 This section lists the source of the additional information that a Delivery Method MUST specify for Job
 1197 Events.

1198 Table 12 lists the source of the information for the job name. The “job-name” is likely more meaningful to
 1199 a user than “job-id”.

1200 **Table 12 – Job Name in Event Notification Content for Job Events**

Source Value	Source Object
job-name (name(MAX))	Job

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

1201

1202 Table 13 lists the source of the information for the job state. If a Printer supports the “job-state-message”
 1203 and “job-detailed-state-message” attributes, it SHOULD use those attributes for the job state information,
 1204 otherwise, it should fabricate such information from the “job-state” and “job-state-reasons”. For some
 1205 Events, a Printer MAY combine this information with Event information.

1206 **Table 13 – Job State in Event Notification Content for Job Events**

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

1207 **9.2.3 Additional Information in Event Notification Content for Printer Events**

1208 This section lists the source of the additional information that a Delivery Method MUST specify for Printer
 1209 Events.

1210 Table 14 lists the source of the information for the printer state. If a Printer supports the “printer-state-
 1211 message”, it SHOULD use that attribute for the job state information, otherwise it SHOULD fabricate such
 1212 information from the “printer-state” and “printer-state-reasons”. For some Events, a Printer MAY combine
 1213 this information with Event information.

1214 **Table 14 – Printer State in Event Notification Content for Printer Events**

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

1215 **10 Delivery Methods**

1216 A Delivery Method is the mechanism, i.e., protocol, by which the Printer delivers an Event Notification to a
1217 Notification Recipient. There are several potential Delivery Methods for Event Notifications, standardized,
1218 as well as proprietary. This document does not define any of these delivery mechanisms. Each Delivery
1219 Method **MUST** be defined in a Delivery Method Document that is separate from this document. New
1220 Delivery Methods will be created as needed using an extension to the registration procedures defined in
1221 [ipp-mod]. Such documents are registered with IANA (see section 13).

1222 The following sorts of Delivery Methods are expected:

- 1223 – The Notification Recipient polls for Event Notifications at intervals directed by the Printer
- 1224 – The Printer sends Event Notifications to the Notification Recipient using http as the transport.
- 1225 – The Printer sends an email message.

1226 This section specifies how to define a Delivery Method Document and what to put in such a document.

1227 A Delivery Method Document:

- 1228 1. **MUST** define a URL scheme name for the Delivery Method.
- 1229 2. **MUST** indicate whether the delivery method is **REQUIRED** or **OPTIONAL** for an IPP Printer to
1230 support if it supports Event Notification.
- 1231 3. **MUST** define the transport and delivery protocol for the Event Notification content that a Printer
1232 **MUST** use, i.e., the entire network stack.
- 1233 4. **MUST** indicate whether or not several Event Notifications can be combined into a compound Event
1234 Notification.
- 1235 5. **MUST** describe how the Delivery Method is initiated, i.e., is it initiated by the receiving user (pull), or
1236 is it initiated by the Printer (push).
- 1237 6. **MUST** indicate whether the Delivery Method is **Machine Consumable** or **Human Consumable**.
- 1238 7. **MUST** define the representation and encoding that a Printer **MUST** use for each value or piece of
1239 information listed in section 9 (9.1 for Machine Consumable Event Notification and/or section 9.2 for
1240 Human Consumable Event Notification).
- 1241 8. **MUST** specify for each attribute in section 9 whether a Printer **MUST**, **SHOULD**, **MAY**, **MUST NOT**,
1242 **SHOULD NOT** or **NEED NOT** send the attribute in an Event Notification content.
- 1243 9. **MUST** define what frequently occurring Events **MUST** be moderated, if any, and whether the
1244 moderation mechanism is configurable. Also whether Events are moderated by sending one per time
1245 unit or one per number of Events.

- 1246 10. MUST discuss the latency and reliability of the transport and delivery protocol.
- 1247 11. MUST discuss the security aspects of the transport and delivery protocol, e.g., how it is handled in
1248 firewalls.
- 1249 12. MUST identify content length restrictions, if any.
- 1250 13. MAY define additional values or pieces of information that a Printer MUST, SHOULD or MAY send
1251 in a Notification content.
- 1252 14. MAY define additional Subscription Template and/or Subscription Description attributes and the
1253 conformance requirements thereof.
- 1254 15. MAY define additional Printer Description attributes and the conformance requirements thereof.

1255 **11 Operations for Notification**

1256 This section defines all of the operations for Notification. Section 7.1 assigns of the “operation-id” for each
1257 operation. The following two sub-sections define Subscription Creation Operations, and other operations.

1258 **11.1 Subscription Creation Operations**

1259 This section defines the Subscription Creation Operations. The first section on Create-Job-Subscriptions
1260 gives most of the information. The other Subscription Creation Operations refer to the section on Create-
1261 Job-Subscriptions, even though the Create-Job-Subscriptions operation is the only OPTIONAL operation in
1262 this document (see section 12).

1263 A Printer MUST support Create-Printer-Subscriptions and the Subscription Template Attributes Group in
1264 Job Creation operations. It MAY support Create-Job-Subscriptions operations.

1265 **11.1.1 Create-Job-Subscriptions Operation**

1266 The operation creates one or more Per-Job Subscription Objects. The client supplies one or more
1267 Subscription Template Attributes Groups each containing one or more of Subscription Template Attributes
1268 (defined in section 5.3).

1269 Except for errors, the Printer MUST create exactly one Per-Job Subscription Object from each Subscription
1270 Template Attributes Group in the request, even if the newly created Subscription Object would have
1271 identical behavior to some existing Subscription Object. The Printer MUST associate each newly created
1272 Per-Job Subscription Object with the target Job, which is specified by the “notify-job-id” operation
1273 attribute.

1274 The Printer MUST accept the request in any of the target job’s ‘not-completed’ states, i.e., ‘pending’,
1275 ‘pending-held’, ‘processing’, or ‘processing-stopped’. The Printer MUST NOT change the job’s “job-state”
1276 attribute because of this operation. If the target job is in any of the ‘completed’ states, i.e., ‘completed’,

1277 'canceled', or 'aborted, then the Printer MUST reject the request and return the 'client-error-not-possible'
1278 status code; the response MUST NOT contain any Subscription Attribute Groups.

1279 Access Rights: To create Per-Job Subscription Objects, the authenticated user (see [IPP-MOD] section 8.3)
1280 performing this operation MUST either be the job owner or have Operator or Administrator access rights
1281 for this Printer (see [IPP-MOD] sections 1 and 8.5). Otherwise the Printer MUST reject the operation and
1282 return: the 'client-error-forbidden', 'client-error-not-authenticated', or 'client-error-not-authorized' status
1283 code as appropriate.

1284 **11.1.1.1 Create-Job-Subscriptions Request**

1285 The following groups of attributes are part of the Create-Job-Subscriptions Request:

1286 Group 1: Operation Attributes

1287 Natural Language and Character Set:

1288 The "attributes-charset" and "attributes-natural-language" attributes as described in [ipp-mod]
1289 section 3.1.4.1.

1290
1291 Target:

1292 The "printer-uri" attribute which defines the target for this operation as described in [ipp-mod]
1293 section 3.1.5.

1294
1295 Requesting User Name:

1296 The "requesting-user-name" attribute SHOULD be supplied by the client as described in [ipp-mod]
1297 section 8.3.

1298
1299 notify-job-id (integer(1:MAX)):

1300 The client MUST supply this attribute and it MUST specify the Job object to associate the Per-Job
1301 Subscription with. The value of "notify-job-id" MUST be the value of the "job-id" of the associated
1302 Job object. If the client does not supply this attribute, the Printer MUST reject this request with a
1303 'client-error-bad-request' status code.

1304 Group 2-N: Subscription Template Attributes

1305 For each occurrence of this group:

1306 The client MUST supply one or more Subscription Template Attributes in any order. See section
1307 5.3 for a description of each such attribute. See section 5.2 for details on processing these
1308 attributes.

1309 **11.1.1.2 Create-Job-Subscriptions Response**

1310 The Printer MUST return to the client the following sets of attributes as part of a Create-Job-Subscriptions
1311 response:

1312 Group 1: Operation Attributes

- 1313 Status Message:
1314 As defined in [ipp-mod].
1315
1316 The Printer can return any status codes defined in [ipp-mod] and section 16. The following is a
1317 description of the important status codes:
1318
1319 **successful-ok:** the Printer created all Subscription Objects requested.
1320 **successful-ok-ignored-subscriptions:** the Printer created some Subscription Objects requested
1321 but some failed. The Subscription Attributes Groups with a “notify-status-code” attribute are
1322 the ones that failed.
1323 **client-error-ignored-all-subscriptions:** the Printer created no Subscription Objects requested
1324 and all failed. The Subscription Attributes Groups with a “notify-status-code” attribute are
1325 the ones that failed
1326 **client-error-not-possible:** For this operation and other Per-Job Subscription operations, this
1327 error can occur because the specified Job has already completed.
1328
1329 Natural Language and Character Set:
1330 The “attributes-charset” and “attributes-natural-language” attributes as described in [ipp-mod]
1331 section 3.1.4.2.
1332
1333 Group 2: Unsupported Attributes
1334 See [ipp-mod] section 3.1.7 for details on returning Unsupported Attributes. This group does not
1335 contain any unsupported Subscription Template Attributes; they are returned in the Subscription
1336 Attributes Group (see below).
1337
1338 Group 3-N: Subscription Attributes
1339 These groups MUST be returned if and only if the ~~operation status code~~ “status-code” parameter
1340 returned in Group 1 has the values: ‘successful-ok’, ‘successful-ok-ignored-
1341 subscriptions’, ~~‘successful-ok-too-many-events’~~, or ‘client-error-ignored-all-subscriptions’.
1342
1343 See section 5.2 for details on the contents of each occurrence of this group.
1344 **11.1.2 Create-Printer-Subscriptions operation**
1345 The operation is identical to Create-Job-Subscriptions with exceptions noted in this section.
1346 The operation creates Per-Printer Subscription Objects instead of Per-Job Subscription Objects, and
1347 associates each newly created Per-Printer Subscription Object with the Printer specified by the operation
1348 target rather than with a specific Job.
1349 The Printer MUST accept the request in any of its states, i.e., ‘idle’, ‘processing’, or ‘stopped’. The Printer
1350 MUST NOT change its “printer-state” attribute because of this operation.
1351 Access Rights: To create Per-Printer Subscription Objects, the authenticated user (see [IPP-MOD] section
1352 8.3) performing this operation MUST have Operator or Administrator access rights for this Printer (see

1353 [IPP-MOD] sections 1 and 8.5). Otherwise, the Printer MUST reject the operation and return: the ‘client-
1354 error-forbidden’, ‘client-error-not-authenticated’, or ‘client-error-not-authorized’ status code as appropriate.

1355 **11.1.2.1 Create-Printer-Subscriptions Request**

1356 The groups are identical to the Create-Job-Subscriptions (see section 11.1.1.1) except that the Operation
1357 Attributes group MUST NOT contain the “notify-job-id” attribute. If the client does supply the “notify-
1358 job-id” attribute, then the Printer MUST treat it as any other unsupported Operation attribute and MUST
1359 return it in the Unsupported Attributes group.

1360 **11.1.2.2 Create-Printer-Subscriptions Response**

1361 The groups are identical to the Create-Job-Subscriptions (see section 11.1.1.2).

1362

1363 **11.1.3 Job Creation Operation – Extensions for Notification**

1364 This document extends the Job Creation operations to create Subscription Objects as a part of the operation.

1365 The operation is identical to Create-Job-Subscriptions with exceptions noted in this section.

1366 Unlike the Create-Job-Subscriptions operation, this operation associates the newly created Subscription
1367 Objects with the Job object created by this operation. The operation succeeds if and only if the Job creation
1368 succeeds. If the Printer does not create some or all of the requested Subscription Objects, the Printer MUST
1369 return a ‘successful-ok-ignored-subscriptions’ status-code instead of a ‘successful-ok’ status-code, but the
1370 Printer MUST NOT reject the operation because of a failure to create Subscription Objects.

1371 If the operation includes a Job Template group, the client MUST supply it after the Operation Attributes
1372 group and before the first Subscription Template Attributes Group.

1373 If a Printer does not support this Notification specification, then it MUST treat the Subscription Attributes
1374 Group like an unknown group and ignore it (see [ipp-mod] section 5.2.2). Because the Printer ignores the
1375 Subscription Attributes Group, it doesn’t return them in the response either, thus indicating to the client that
1376 the Printer doesn’t support Notification.

1377 Access Rights: To create Per-Job Subscription Objects, the authenticated user (see [IPP-MOD] section 8.3)
1378 performing this operation MUST either have permission to create Jobs on the Printer. Otherwise the Printer
1379 MUST reject the operation and return: the ‘client-error-forbidden’, ‘client-error-not-authenticated’, or
1380 ‘client-error-not-authorized’ status code as appropriate.

1381 **11.1.3.1 Job Creation Request**

1382 The groups for this operation are sufficiently different from the Create-Job-Subscriptions operation that
1383 they are all presented here. The following groups of attributes are supplied as part of a Job Creation
1384 Request:

1385 Group 1: Operation Attributes

1386 Same as defined in [ipp-mod] for Print-Job, Print-URI, and Create-Job requests.

1387 Group 2: Job Template Attributes

1388 The client OPTIONALLY supplies a set of Job Template attributes as defined in [ipp-mod] section
1389 4.2.

1390 Group 3 to N: Subscription Template Attributes

1391 The same as Group 2-N in Create-Job-Subscriptions. See section 11.1.1.1.

1392 Group N+1: Document Content (Print-Job only)

1393 The client MUST supply the document data to be processed.

1394 11.1.3.2 Job Creation Response

1395 The Printer MUST return to the client the following sets of attributes as part of a Print-Job, Print-URI, and
1396 Create-Job Response:

1397 Group 1: Operation Attributes

1398

1399 Status Message:

1400

1401 As defined in [ipp-mod] for Print-Job, Print-URI, and Create-Job requests.

1402

1403 The Printer can return any status codes defined in [ipp-mod] and section 16. The following is a
1404 description of the important status codes:

1405

1406 **successful-ok:** the Printer created the Job and all Subscription Objects requested.

1407 **successful-ok-ignored-subscriptions:** the Printer created the Job and not all of the Subscription
1408 Objects requested. This status-code hides ‘successful-ok-xxx’ status-codes that could reveal
1409 problems in Job creation. The Printer MUST not return the ‘client-error-ignored-all-
1410 subscriptions’ status code for Job Creation operations because the Printer returns an error
1411 status-code only when it fails to create a Job.

1412

1413 Natural Language and Character Set:

1414 The “attributes-charset” and “attributes-natural-language” attributes as described in [ipp-mod]
1415 section 3.1.4.2.

1416

1417 Group 2: Unsupported Attributes

1418 See [ipp-mod] section 3.1.7 for details on returning Unsupported Attributes. This group does not
1419 contain any unsupported Subscription Template Attributes; they are returned in the Subscription
1420 Attributes Group (see below).

1421

1422 Group 3: Job Object Attributes

1423 As defined in [ipp-mod] for Print-Job, Print-URI, and Create-Job requests.
1424

1425 Group 4 to N: Subscription Attributes

1426 These groups MUST be returned if and only if the client supplied Subscription Template Attributes
1427 and the operation was accepted.

1428
1429 See section 5.2 for details on the contents of each occurrence of this group.
1430

1431 **11.2 Other Operations**

1432 This section defines other operations on Subscription objects.

1433 **11.2.1 Validate-Job Operation - Extensions for Notification**

1434 A client can test whether one or more Subscription Objects could be created using the Validate-Job
1435 operation. The client supplies one or more Subscription Template Attributes Groups (defined in section
1436 5.3), just as in a Job Creation request.

1437 A Printer MUST support this extension to this operation.

1438 The Printer MUST accept requests that are identical to the Job Creation request defined in section 11.1.3.1,
1439 except that the request MUST not contain document data.

1440 The Printer MUST return the same groups and attributes as the Print-Job operation (section 11.1.3.1) with
1441 the following exceptions. The Printer MUST NOT return a Job Object Attributes Group because no Job is
1442 created. The Printer MUST NOT return the “notify-subscription-id” attribute in any Subscription Attribute
1443 Group because no Subscription Object is created.

1444 If the Printer would succeed in creating a Subscription Object, the corresponding Subscription Attributes
1445 Group either has no ‘status-code’ attribute or a ‘status-code’ attribute with a value of ‘successful-ok-too-
1446 many-events’ or ‘successful-ok-ignored-or-substituted-attributes’ (see sections 5.2 and 17). The status-
1447 codes have the same meaning as in Job Creation except the results state what “would happen”.

1448 The Printer MUST validate Subscription Template Attributes Groups in the same manner as the Job
1449 Creation operations. However, to cause the Printer to validate as many Subscription Template Attributes as
1450 possible, the Printer MUST assume that it can create up to the number of Subscription Objects equal to the
1451 value of “notify-max-job-subscriptions-supported”.

1452 **11.2.2 Get-Printer-Attributes - Extensions for Notification**

1453 This operation is extended so that it returns Printer attributes defined in this document.

1454 A Printer MUST support this extension to this operation.

1455 In addition to the requirements of [ipp-mod] section 3.2.5, a Printer MUST support the following additional
1456 values for the “requested-attributes” Operation attribute in this operation and return such attributes in the
1457 Printer Object Attributes group of its response.

1458 1. **Subscription Template Attributes:** Each supported attribute in column 2 of Table 1.

1459 2. **New Printer Description Attributes:** Each supported attribute in section 6.

1460 3. **New Group Name:** The ‘subscription-template’ group name, which names all supported
1461 Subscription Template Attribute in column 2 of Table 1. Note: This group name is also used in the
1462 Get-Subscription-Attributes and Get-Subscriptions operation with an analogous meaning.

1463 4. **Extended Group Name ‘printer-description’:** The ‘printer-description’ group name, which names
1464 all Printer Description attributes according to [ipp-mod] section 3.2.5. In this extension ‘printer-
1465 description’ names all attributes specified in [ipp-mod] plus those named in item 2 of this list.

1466 5. **Extended Group Name ‘all’:** The ‘all’ group name, which names all Printer attributes according to
1467 [ipp-mod] section 3.2.5. In this extension ‘all’ names all attributes specified in [ipp-mod] plus those
1468 named in items 1 and 2 of this list.

1469

1470 **11.2.3 Get-Subscription-Attributes operation**

1471 This operation allows a client to request the values of the attributes of a Subscription Object.

1472 A Printer MUST support this operation.

1473 This operation is almost identical to the Get-Job-Attributes operation (see [ipp-mod] section 3.3.4). The
1474 only differences are that the operation is directed at a Subscription Object rather than a Job object, and the
1475 returned attribute group contains Subscription Object attributes rather than Job object attributes.

1476 **11.2.3.1 Get-Subscription-Attributes Request**

1477 The following groups of attributes are part of the Get-Subscription-Attributes request:

1478 Group 1: Operation Attributes

1479 Natural Language and Character Set:

1480 The “attributes-charset” and “attributes-natural-language” attributes as described in section [ipp-
1481 mod] 3.1.4.1.

1482

1483 Target:

1484 The “printer-uri” attribute which defines the target for this operation as described in [ipp-mod]
1485 section 3.1.5.

1486

1487 “notify-subscription-id” (integer (1:MAX)):
1488 The client MUST supply this attribute. The Printer MUST support this attribute. This attribute
1489 specifies the Subscription Object from which the client is requesting attributes. If the client omits
1490 this attribute, the Printer MUST reject this request with the ‘client-error-bad-request’ status code.
1491

1492 Requesting User Name:
1493 The “requesting-user-name” attribute SHOULD be supplied by the client as described in [ipp-mod]
1494 section 8.3.
1495

1496 “requested-attributes” (1setOf keyword):
1497 The client OPTIONALLY supplies this attribute. The Printer MUST support this attribute. This
1498 attribute specifies the attributes of the specified Subscription Object that the Printer MUST return in
1499 the response. Each value of this attribute is either an attribute name (defined in sections 5.3 and 5.4)
1500 or an attribute group name. The attribute group names are:
1501

- 1502 - ‘subscription-template’: ~~the subset of the Subscription Template~~all attributes that apply to a
1503 Subscription object are both defined in section 5.3 ~~) that the implementation supports for~~and
1504 present on the specified Subscription Object (column 1 of Table 1).
- 1505 - ‘subscription-description’: ~~the subset of the Subscription Description~~all attributes that are both
1506 defined in section 5.4 ~~that the implementation supports for~~and present on the specified
1507 Subscription Object (Table 2).
- 1508 - ‘all’: all attributes that are present on the specified Subscription Object.

1509 A Printer MUST support all these group names.
1510 If the client omits this attribute, the Printer MUST respond as if this attribute had been supplied with
1511 a value of ‘all’.

1512 11.2.3.2 Get-Subscription-Attributes Response

1513 The Printer returns the following sets of attributes as part of the Get-Subscription-Attributes Response:

1514 Group 1: Operation Attributes

1515 Status Message:
1516 Same as [ipp-mod].
1517

1518 Natural Language and Character Set:
1519 The “attributes-charset” and “attributes-natural-language” attributes as described in [ipp-mod]
1520 section 3.1.4.2. The “attributes-natural-language” MAY be the natural language of the Subscription
1521 Object, rather than the one requested.
1522

1523 Group 2: Unsupported Attributes

1524 See [ipp-mod] section 3.1.7 for details on returning Unsupported Attributes.
1525

1526 The response NEED NOT contain the “requested-attributes” operation attribute with any supplied
1527 values (attribute keywords) that were requested by the client but are not supported by the Printer. If

1528 the Printer does return unsupported attributes referenced in the “requested-attributes” operation
1529 attribute and that attribute included group names, such as ‘all’, the unsupported attributes MUST
1530 NOT include attributes described in the standard but not supported by the implementation.
1531

1532 Group 3: Subscription Attributes

1533 This group contains a set of attributes with their current values. Each attribute in this group:

- 1534 a) MUST be specified by the “requested-attributes” attribute in the request, AND
1535 b) MUST be present on the specified Subscription Object AND
1536 c) MUST NOT be restricted by the security policy in force. For example, a Printer MAY prohibit
1537 a client who is not the creator of a Subscription Object from seeing some or all of its attributes.
1538 See [ipp-mod] section 8.

1539 The Printer can return the attributes of the Subscription Object in any order. The client MUST
1540 accept the attributes in any order.

1541 11.2.4 Get-Subscriptions operation

1542 This operation allows a client to retrieve the values of attributes of all Subscription Objects belonging to a
1543 Job or Printer.

1544 A Printer MUST supported this operation.

1545 This operation is similar to the Get-Subscription-Attributes operation, except that this Get-Subscriptions
1546 operation returns attributes from possibly more than one object.

1547 This operation is similar to the Get-Jobs operation (see [ipp-mod] section 3.2.6), except that the operation
1548 returns Subscription Objects rather than Job objects.

1549 11.2.4.1 Get-Subscriptions Request

1550 The following groups of attributes are part of the Get-Subscriptions request:

1551 Group 1: Operation Attributes

1552 Natural Language and Character Set:

1553 The “attributes-charset” and “attributes-natural-language” attributes as described in [ipp-mod]
1554 section 3.1.4.1.

1555

1556 Target:

1557 The “printer-uri” attribute which defines the target for this operation as described in [ipp-mod]
1558 section 3.1.5.

1559

1560 Requesting User Name:

1561 The “requesting-user-name” attribute SHOULD be supplied by the client as described in [ipp-mod]
1562 section 8.3.

1563

1564 “notify-job-id” (integer(1:MAX)):

1565 If the client specifies this attribute, the Printer returns the specified attributes of all Per-Job
1566 Subscription Objects associated with the Job whose “job-id” attribute value equals the value of this
1567 attribute. If the client does not specify this attribute, the Printer returns the specified attributes of all
1568 Per-Printer Subscription Objects. Note: there is no way to get all Per-Job Subscriptions.

1569

1570 “limit” (integer(1:MAX)):

1571 The client OPTIONALLY supplies this attribute. The Printer MUST support this attribute. It is an
1572 integer value that determines the maximum number of Subscription Objects that a client will receive
1573 from the Printer even if the “my-subscriptions” attribute constrains which Subscription Objects are
1574 returned. The limit is a “stateless limit” in that if the value supplied by the client is ‘N’, then only
1575 the first ‘N’ Subscription Objects are returned in the Get-Subscriptions Response. There is no
1576 mechanism to allow for the next ‘M’ Subscription Objects after the first ‘N’ Subscription Objects.
1577 If the client does not supply this attribute, the Printer responds with all applicable Subscription
1578 Objects.

1579

1580 “requested-attributes” (1setOf type2 keyword):

1581 The client OPTIONALLY supplies this attribute. The Printer MUST support this attribute. This
1582 attribute specifies the attributes of the specified Subscription Objects that the Printer MUST return
1583 in the response. Each value of this attribute is either an attribute name (defined in sections 5.3 and
1584 5.4) or an attribute group name (defined in section 11.2.3.1). If the client omits this attribute, the
1585 Printer MUST respond as if the client had supplied this attribute with the one value: ‘notify-
1586 subscription-id’.

1587

1588 “my-subscriptions” (boolean):

1589 The client OPTIONALLY supplies this attribute. The Printer MUST support this attribute. If the
1590 value is ‘false’, the Printer MUST consider the Subscription Objects from all users as candidates. If
1591 the value is ‘true’, the Printer MUST return the Subscription Objects created by the requesting user
1592 of this request. If the client does not supply this attribute, the Printer MUST respond as if the client
1593 had supplied the attribute with a value of ‘false’. The means for authenticating the requesting user
1594 and matching the Subscription Objects is similar to that for Jobs which is described in [ipp-mod]
1595 section 8.

1596 11.2.4.2 Get-Subscriptions Response

1597 The Printer returns the following sets of attributes as part of the Get-Subscriptions Response:

1598 Group 1: Operation Attributes

1599 Status Message:

1600 Same as [ipp-mod].

1601

1602 Natural Language and Character Set:

1603 The “attributes-charset” and “attributes-natural-language” attributes as described in [ipp-mod]
1604 section 3.1.4.2.

1605

1606 Group 2: Unsupported Attributes

1607 Same as for Get-Subscription-Attributes.

1608

1609 Groups 3 to N: Subscription Attributes

1610 The Printer responds with one Subscription Attributes Group for each requested Subscription Object
1611 (see the “notify-job-id” attribute in the Operation Attributes Group of this operation). ~~The Printer
1612 ignores (does not respond with) any requested attribute which is not supported or which is restricted
1613 by the security policy in force, including whether the requesting user is the user that submitted the
1614 job (job originating user) or not (see [ipp-mod] section 8).~~

1615

1616 The Printer returns Subscription Objects in any order.

1617

1618 If the “limit” attribute is present in the Operation Attributes group of the request, the number of
1619 Subscription Attributes Groups in the response MUST NOT exceed the value of the “limit”
1620 attribute.

1621

1622 If there are no Subscription Objects associated with the specified Job or Printer, the Printer MUST
1623 return zero Subscription Attributes Groups and it MUST NOT treat this case as an error, i.e., the
1624 status-code MUST be ‘successful-ok’ unless something else causes the status code to have some
1625 other value.

1626

1627 See the Group 3 response (Subscription Attributes Group) of the Get-Subscription-Attributes
1628 operation (section 11.2.3.2) for the attributes that a Printer returns in this group.

1629

1630 11.2.5 Renew-Subscription operation

1631 This operation allows a client to request the Printer to extend the lease on a Per-Printer Subscription Object.

1632 The Printer MUST support this operation.

1633 The Printer MUST accept this request for a Per-Printer Subscription Object in any of the target Printer’s
1634 states, i.e., ‘idle’, ‘processing’, or ‘stopped’, but MUST NOT change the Printer’s “printer-state” attribute.

1635 The Printer MUST reject this request for a Per-Job Subscription Object because it has no lease (see section
1636 5.4.3). The status code returned MUST be ‘client-error-not-possible’.

1637 *Access Rights:* The authenticated user (see [IPP-MOD] section 8.3) performing this operation MUST either
1638 be the owner of the Per-Printer Subscription Object or have Operator or Administrator access rights for the
1639 Printer (see [IPP-MOD] sections 1 and 8.5). Otherwise, the Printer MUST reject the operation and return:
1640 the ‘client-error-forbidden’, ‘client-error-not-authenticated’, or ‘client-error-not-authorized’ status code as
1641 appropriate.

1642 **11.2.5.1 Renew-Subscription Request**

1643 The following groups of attributes are part of the Renew-Subscription Request:

1644 Group 1: Operation Attributes

1645 Natural Language and Character Set:

1646 The “attributes-charset” and “attributes-natural-language” attributes as described in [ipp-mod]
1647 section 3.1.4.1.

1648 Target:

1649 The “printer-uri” attribute which defines the target for this operation as described in [ipp-mod]
1650 section 3.1.5.

1651
1652 “notify-subscription-id” (integer (1:MAX)):

1653 The client **MUST** supply this attribute. The Printer **MUST** support this attribute. This attribute
1654 specifies the Per-Printer Subscription Object whose lease the Printer **MUST** renew. If the client
1655 omits this attribute, the Printer **MUST** reject this request with the ‘client-error-bad-request’ status
1656 code.
1657

1658 Requesting User Name:

1659 The “requesting-user-name” (name(MAX)) attribute **SHOULD** be supplied by the client as
1660 described in [ipp-mod] section 8.3.
1661

1662

1663 Group 2: Subscription Template Attributes

1664
1665 “notify-lease-duration” (integer(0:MAX)):

1666 The client **MAY** supply this attribute. It indicates the number of seconds to renew the lease for the
1667 specified Subscription Object. A value of 0 requests an infinite lease (which **MAY** require Operator
1668 access rights). If the client omits this attribute, the Printer **MUST** use the value of the Printer’s
1669 “notify-lease-duration-default” attribute. See section 5.3.7 for more details.

1670 **11.2.5.2 Renew-Subscription Response**

1671 The Printer returns the following sets of attributes as part of the Renew-Subscription Response:

1672 Group 1: Operation Attributes

1673 Status Message:

1674 Same as [ipp-mod].

1675

1676 The following are some of the status codes returned:

1677

1678 **successful-ok:** The operation successfully renewed the lease on the Subscription Object for the
1679 requested duration..

1680 **successful-ok-ignored-or-substituted-attributes:** The operation successfully renewed the lease on
1681 the Subscription Object for some duration other than the amount requested.

1682 **client-error-not-possible:** The operation failed because the “notify-subscription-id” Operation
1683 attribute identified a Per-Job Subscription Object.

1684 **client-error-not-found:** The operation failed because the “notify-subscription-id” Operation
1685 attribute identified a non-existent Subscription Object.

1686

1687 Natural Language and Character Set:

1688 The “attributes-charset” and “attributes-natural-language” attributes as described in [ipp-mod]
1689 section 3.1.4.2. The “attributes-natural-language” MAY be the natural language of the Subscription
1690 Object, rather than the one requested.

1691

1692 Group 2: Unsupported Attributes

1693 See [ipp-mod] section 3.1.7 for details on returning Unsupported Attributes.

1694

1695 Group 3: Subscription Attributes

1696 The Printer MUST return the following Subscription Attribute:

1697 “notify-lease-duration” (integer(0:MAX)):

1698 The value of this attribute MUST be the number of seconds that the Printer has granted for the lease
1699 of the Subscription Object (see section 5.3.7 for details, such as the value of this attribute when the
1700 Printer doesn’t support the requested value).

1701

1702

1703 **11.2.6 Cancel-Subscription operation**

1704 This operation allows a client to delete a Subscription Object and stop the Printer from sending more Event
1705 Notifications. Once performed, there is no way to reference the Subscription Object.

1706 A Printer MUST supported this operation.

1707 The Printer MUST accept this request in any of the target Printer’s states, i.e., ‘idle’, ‘processing’, or
1708 ‘stopped’, but MUST NOT change the Printer’s “printer-state” attribute.

1709 If the specified Subscription Object is a Per-Job Subscription Object, the Printer MUST accept this request
1710 in any of the target Job’s states, but MUST NOT change the Job’s “job-state” attribute or affect the Job.

1711 *Access Rights:* The authenticated user (see [IPP-MOD] section 8.3) performing this operation MUST either
1712 be the owner of the Subscription Object or have Operator or Administrator access rights for the Printer (see
1713 [IPP-MOD] sections 1 and 8.5). Otherwise, the Printer MUST reject the operation and return: the ‘client-
1714 error-forbidden’, ‘client-error-not-authenticated’, or ‘client-error-not-authorized’ status code as appropriate.

1715 Note: There is no way to change any attributes on a Subscription Object, except the “notify-lease-
1716 duration” attribute (using the Renew-Subscription operation). In order to change other attributes, a client

1717 performs a Subscription Creation Operation and Cancel-Subscription operation on the old Subscription
1718 Object. If the client wants to avoid missing Event Notifications, it performs the Subscription Creation
1719 Operation first. If this order would create too many Subscription Objects on the Printer, the client reverses
1720 the order.

1721 11.2.6.1 Cancel-Subscription Request

1722 The following groups of attributes are part of the Cancel-Subscription Request:

1723 Group 1: Operation Attributes

1724 Natural Language and Character Set:

1725 The “attributes-charset” and “attributes-natural-language” attributes as described in [ipp-mod]
1726 section 3.1.4.1.

1727

1728 Target:

1729 The “printer-uri” attribute which defines the target for this operation as described in [ipp-mod]
1730 section 3.1.5.

1731

1732 “notify-subscription-id” (integer (1:MAX)):

1733 The client MUST supply this attribute. The Printer MUST support this attribute. This attribute
1734 specifies the Subscription Object that the Printer MUST cancel. If the client omits this attribute, the
1735 Printer MUST reject this request with the ‘client-error-bad-request’ status code.

1736

1737 Requesting User Name:

1738 The “requesting-user-name” attribute SHOULD be supplied by the client as described in [ipp-mod]
1739 section 8.3.

1740

1741 11.2.6.2 Cancel-Subscription Response

1742 The Printer returns the following sets of attributes as part of the Cancel-Subscription Response:

1743 Group 1: Operation Attributes

1744 Status Message:

1745 Same as [ipp-mod].

1746

1747 The following are some of the status codes returned:

1748

1749 **successful-ok:** The operation successfully canceled (deleted) the Subscription Object..

1750 **client-error-not-found:** The operation failed because the “notify-subscription-id” Operation
1751 attribute identified a non-existent Subscription Object.

1752

1753 Natural Language and Character Set:
 1754 The “attributes-charset” and “attributes-natural-language” attributes as described in [ipp-mod]
 1755 section 3.1.4.2. The “attributes-natural-language” MAY be the natural language of the Subscription
 1756 Object, rather than the one requested.
 1757

1758 Group 2: Unsupported Attributes

1759 See [ipp-mod] section 3.1.7 for details on returning Unsupported Attributes.
 1760

1761 12 Conformance Requirements

1762 It is OPTIONAL to implement this Event Notification specification.

1763 If this Event Notification specification is implemented, Printers MUST:

- 1764 1. meet the Conformance Requirements detailed in section 5 of [ipp-mod].
- 1765 2. support all of the following attributes:
 - 1766 a. REQUIRED Subscription Object attributes in section 5.
 - 1767 b. REQUIRED Printer Description object attributes in section 6.
 - 1768 c. REQUIRED attributes in Event Notification content in section 8.
- 1769 3. send Event Notifications that conform to the requirements of the Delivery Method Document for each
 1770 supported Delivery Method (the conformance requirements for Delivery Method Documents is
 1771 specified in section 10).
- 1772 4. support all operations as described in Table 15:

1773 **Table 15 – Conformance Requirements for Operations**

Attribute	Conformance requirements
Subscription Attributes Group	REQUIRED
Create-Printer-Subscriptions (section 11.1.2)	REQUIRED
Create-Job-Subscriptions (section 11.1.1)	OPTIONAL
<u>Validate-Job - extensions (section 11.2.1)</u>	<u>REQUIRED</u>
<u>Get-Printer-Attributes - extensions (section 11.2.2)</u>	<u>REQUIRED</u>
Get-Subscription-Attributes (section 11.2.3)	REQUIRED
Get-Subscriptions (section 11.2.4)	REQUIRED
Renew-Subscription (section 11.2.5)	REQUIRED
Cancel-Subscription (section 11.2.6)	REQUIRED

1774

1775 **13 IANA Considerations**

1776 This section describes the procedures for registering Event Notification Delivery Method proposals with
1777 IANA to be used with this document. Such Delivery Method proposals can be IETF standards track
1778 documents or vendor-defined documents. In either case, they will be registered with IANA using
1779 procedures that extend those defined in [ipp-mod] section 6 and 11.

1780 These extension procedures are aligned with the guidelines as set forth by the IESG [IANA-CON]. Section
1781 13.1 defines the format and content for new registrations for consideration. IANA will reject registration
1782 proposals that leave out required information or do not follow the appropriate format described in Section
1783 13.1.

1784 Implementers can, at any time, define new Event Notification Delivery Methods by proposing the complete
1785 specification to IANA:

1786 iana@iana.org

1787 or by filling out the appropriate form on the IANA web pages (<http://www.iana.org>).

1788 IANA will forward the registration proposal to the IPP Designated Expert who will review the proposal
1789 with a mailing list that the Designated Expert keeps for this purpose. Initially, that list will be the mailing
1790 list used by the IPP WG:

1791 ipp@pwg.org

1792 even after the IPP WG is disbanded as permitted by [IANA-CON]. The IPP Designated Expert is appointed
1793 by the IESG Area Director responsible for IPP, according to [IANA-CON].

1794 When a Delivery Method Document is approved, the IPP Designated Expert becomes the point of contact
1795 for any future maintenance that might be required for that registration.

1796 **13.1 Format and Requirements for IPP Delivery Method Registration Proposals**

1797 This section defines the format and requirements for an IPP Event Notification Delivery Method
1798 Registration Proposal. A Delivery Method Registration Proposal:

1799 1. MUST contain the following information:

1800 Type of registration: IPP Event Notification Delivery Method

1801 Name of this delivery method:

1802 Proposed URL scheme name of this delivery method:

1803 Name of proposer:

1804 Address of proposer:

1805 Email address of proposer:

1806 Is this delivery method REQUIRED or OPTIONAL for conformance to the IPP Event Notification
1807 Specification document:

1808 Is this delivery method defining Machine Consumable and/or Human Consumable content:
1809 2. MUST meet the conformance requirements for Delivery Method Documents specified in section 10.

1810

1811 **14 Internationalization Considerations**

1812 This IPP Notification specification continues support for the internationalization of [ipp-mod] of attributes
1813 containing text strings and names. Allowing a Subscribing Client to specify a different natural language
1814 and charset for each Subscription Object increases the internationalization support.

1815 The Printer MUST be able to localize the content of Human Consumable Event Notifications and to
1816 localize the value of “notify-text” attribute in Machine Consumable Event Notifications that it sends to
1817 Notification Recipients. For localization, the Printer MUST use the value of the “notify-charset” attribute
1818 and the “notify-natural-language” attribute in the Subscription Object supplied by the Subscribing Client.

1819 **15 Security Considerations**

1820 By far the biggest security concern is the abuse of notification: sending unwanted Event Notifications to
1821 third parties (i.e., spam). The problem is made worse by notification addresses that may be redistributed to
1822 multiple parties (e.g., mailing lists). There exist scenarios where third party notification is required (see
1823 Scenario #2 and #3 in [ipp-not-req]). The fully secure solution would require active agreement of all
1824 recipients before sending out anything. However, requirement #9 in [ipp-req] (“There is no requirement for
1825 IPP Printer receiving the print request to validate the identity of an Event recipient”) argues against this.
1826 Certain systems may decide to disallow third party Event Notifications (a traditional fax model).

1827 Clients submitting Notification requests to the IPP Printer has the same security issues as submitting an
1828 IPP/1.1 print job request. The same mechanisms used by IPP/1.1 can therefore be used by the client
1829 Notification submission. Operations that require authentication can use the HTTP authentication.
1830 Operations that require privacy can use the HTTP/TLS privacy.

1831 The Notification access control model should be similar to the IPP access control model for Jobs. Creating
1832 a Per-Printer Subscription Object is associated with a user. Only the creator or an Operator can cancel the
1833 Subscription Object. The system may limit the listing of items to only those items owned by the user.
1834 Some Subscription Objects (e.g., those that have a lifetime longer than a job) can be done only by
1835 privileged users (users having Operator and/or Administrator access rights), if that is the authorization
1836 policy.

1837 The standard security concerns (delivery to the right user, privacy of content, tamper proof content) apply to
1838 the Delivery Method. IPP should use the security mechanism of the Delivery Method used. Some delivery
1839 mechanisms are more secure than others. Therefore, sensitive Event Notifications should use the Delivery
1840 Method that has the strongest security.

1841 **16 Status Codes**

1842 The following status codes are defined as extensions for Notification and are returned as the value of the
1843 “status-code” parameter in the Operation Attributes Group of a response (see [ipp-mod] section 3.1.6.1).
1844 Operations in this document can also return the status codes defined in section 13 of [ipp-mod]. The
1845 ‘successful-ok’ status code is an example of such a status code.

1846 **16.1 successful-ok-ignored-subscriptions (0x0003)**

1847 The Subscription Creation Operation was unable to create all requested Subscription Objects.~~If a returned~~
1848 ~~Subscription Attributes Group does not contain the “notify-subscription-id” attribute, the “notify-status-~~
1849 ~~code” attribute gives the reason why the corresponding Subscription Object was not created.~~

1850 For a Create-Job-Subscriptions or Create-Printer-Subscriptions operation, this status code means that the
1851 Printer created one or more Subscription Objects, but not all requested Subscription Objects.

1852 For a Job Creation operation, this status code means that the Printer created the Job along with zero or more
1853 Subscription Objects. The Printer returns this status code even if other job attributes are unsupported or in
1854 conflict. That is, if an IPP Printer finds a warning that would allow it to return ‘successful-ok-ignored-
1855 subscriptions’ and either ‘successful-ok-ignored-or-substituted-attributes’ and/or ‘successful-ok-conflicting-
1856 attributes’, it MUST return ‘successful-ok-ignored-subscriptions’.

1857 **16.2 client-error-ignored-all-subscriptions (0x0414)**

1858 This status code is the same as ‘successful-ok-ignored-subscriptions’ except that only the Create-Job-
1859 Subscriptions and Create-Printer-Subscriptions operation return it. They return this status code only when
1860 the Printer creates zero Subscription Objects.

1861 **17 Status Codes in Subscription Attributes Groups**

1862 This section contains values of the “notify-status-code” attribute that the Printer returns in a Subscription
1863 Attributes Group in a response when the corresponding Subscription Object:

- 1864 1. is not created or
- 1865 2. is created and some of the client-supplied attributes are not supported.

1866 ~~This status code gives the most important reason why the Subscription Object was not created.~~The
1867 following sections are ordered in decreasing order of importance of the status-codes.

1868 **17.1 client-error-uri-scheme-not-supported (0x040C)**

1869 This status code is defined in [ipp-mod]. This document extends its meaning and allows it to be in a
1870 Subscription Attributes Group of a response.

1871 The scheme of the client-supplied URI in a “notify-recipient-uri” Subscription Template Attribute in a
 1872 Subscription Creation Operation is not supported. See section 5.3.1.

1873 **17.2 client-error-too-many-subscriptions (0x0415)**

1874 The number of Subscription Objects supported by the Printer would be exceeded if this Subscription Object
 1875 were created (see section 5.2).

1876 **17.3 successful-ok-too-many-events (0x0005)**

1877 The client supplied more Events in the “notify-events” operation attribute of a Subscription Creation
 1878 Operation than the Printer supports, as indicated in its “notify-max-events-supported” Printer attribute (see
 1879 section 5.3.2).

1880 **17.4 successful-ok-ignored-or-substituted-attributes (0x0001)**

1881 This status code is defined in [ipp-mod]. This document extends its meaning to include unsupported
 1882 Subscription Template Attributes and it can appear in a Subscription Attributes Group.

1883 **18 Encodings of Additional Attribute Tags**

1884 This section assigns values to two attributes tags as extensions to the encoding defined in [ipp-pro]).

1885 The “subscription-attributes-tag” delimits Subscription Template Attributes Groups in requests and
 1886 Subscription Attributes Groups in responses.

1887 The “event-notification-attributes-tag” delimits Event Notifications in Delivery Methods that use an IPP-
 1888 like encoding.

1889 The following table specifies the values for the delimiter tags:

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

1890 **19 References**

1891 [IANA-CON]

1892 Narte, T. and Alvestrand, H.T.: Guidelines for Writing an IANA Considerations Section in RFCs,
 1893 Work in Progress, draft-iesg-iana-considerations-04.txt, May 21, 1998.

1894 [ipp-mod]

1895 deBry, R., , Hastings, T., Herriot, R., Isaacson, S., Powell, P., “Internet Printing Protocol/1.1: Model
 1896 and Semantics”, <draft-ietf-ipp-model-v11-07.txt>, work in progress, May 22, 2000.

- 1897 [ipp-not-req]
1898 deBry, R., Lewis, H., Hastings, T., “Internet Printing Protocol/1.1: Requirements for IPP
1899 Notifications”, <draft-ietf-ipp-not-03.txt>, work in progress, August 11, 1999.
- 1900 [ipp-pro]
1901 Herriot, R., Butler, S., Moore, P., Tuner, R., “Internet Printing Protocol/1.1: Encoding and
1902 Transport”, <draft-ietf-ipp-protocol-v11-06.txt>, work in progress, May 30, 2000.
- 1903 [ipp-prog]
1904 Hastings, T., Bergman, R., Lewis, H., “Proposed Job Progress Attributes for IPP”, <draft-ietf-ipp-
1905 job-prog.txt> work in progress, February 2, 2000.
- 1906 [ipp-set]
1907 Kugler, C., , Hastings, T., Herriot, R., Lewis, H., “Internet Printing Protocol (IPP): Job and Printer
1908 Set Operations”, <draft-ietf-ipp-job-printer-set-ops-01.txt>, work in progress, March 8, 2000.
- 1909 [ipp-set2]
1910 Kugler, C., , Hastings, T., Lewis, H., “Internet Printing Protocol (IPP): Additional Operations, Set
1911 2”, <draft-ietf-ipp-ops-set2.txt>, work in progress, February 3, 2000.
- 1912 [RFC2026]
1913 S. Bradner, "The Internet Standards Process -- Revision 3", RFC 2026, October 1996.
- 1914 [RFC2119]
1915 S. Bradner, “Key words for use in RFCs to Indicate Requirement Levels”, RFC 2119 , March 1997
- 1916 [RFC2566]
1917 deBry, R., , Hastings, T., Herriot, R., Isaacson, S., Powell, P., “Internet Printing Protocol/1.0: Model
1918 and Semantics”, RFC 2566, April 1999.
- 1919 [RFC2567]
1920 Wright, D., “Design Goals for an Internet Printing Protocol”, RFC 2567, April 1999.
- 1921 [RFC2568]
1922 Zilles, S., “Rationale for the Structure and Model and Protocol for the Internet Printing Protocol”,
1923 RFC 2568, April 1999.
- 1924 [RFC2569]
1925 Herriot, R., Hastings, T., Jacobs, N., Martin, J., “Mapping between LPD and IPP Protocols”, RFC
1926 2569, April 1999.

1927 **20 Author’s Addresses**

- 1928 Scott A. Isaacson (Editor)
1929 Novell, Inc.
1930 122 E 1700 S

1931 Provo, UT 84606
1932
1933 Phone: 801-861-7366
1934 Fax: 801-861-2517
1935 e-mail: sisacson@novell.com
1936
1937 Tom Hastings
1938 Xerox Corporation
1939 737 Hawaii St. ESAE 231
1940 El Segundo, CA 90245
1941
1942 Phone: 310-333-6413
1943 Fax: 310-333-5514
1944 e-mail: hastings@cp10.es.xerox.com
1945
1946 Robert Herriot
1947 Xerox Corporation
1948 3400 Hillview Ave., Bldg #1
1949 Palo Alto, CA 94304
1950
1951 Phone: 650-813-7696
1952 Fax: 650-813-6860
1953 Email: robert.herriot@pahv.xerox.com
1954
1955 Roger deBry
1956 Utah Valley State College
1957 Orem, UT 84058
1958
1959 Phone: (801) 222-8000
1960 EMail: debryro@uvsc.edu
1961
1962 Jay Martin
1963 e-mail: jkm@underscore.com
1964
1965 Michael Shepherd
1966 Xerox Corporation
1967 800 Phillips Road MS 128-51E
1968 Webster, NY 14450
1969
1970 Phone: 716-422-2338
1971 Fax: 716-265-8871
1972 e-mail: mshepherd@crt.xerox.com
1973

1974 Ron Bergman (Editor)
1975 Hitachi Koki Imaging Solutions
1976 1757 Tapo Canyon Road
1977 Simi Valley, CA 93063-3394
1978
1979 Phone: 805-578-4421
1980 Fax: 805-578-4001
1981 Email: rbergma@hitachi-hkis.com

1982 **A. Appendix - Model for Notification with Cascading Printers**

1983 With this model (see Figure 2), there is an intervening Print server between the human user and the output-
1984 device. So the system effectively has two Printers. There are two cases to consider.

- 1985 1. When the Printer 1 (in the server) generates Events, the system behaves like the client and Printer in
1986 Figure 1. In this case, Printer 1 sends Event Notifications that are shown as Event Notifications (A)
1987 of Figure 2,.
- 1988 2. When the Printer 2 (in the output-device) generates Events, there are two possible system
1989 configurations:
 - 1990 a) Printer 1 forwards the client-supplied Subscription Creation Operations to the downstream
1991 Printer 2 and lets Printer 2 send the Event Notifications directly to the Notification Recipients
1992 supplied by the Client (Event Notifications(C) in the diagram).
 - 1993 b) Printer 1 performs the client-supplied Subscription Creation Operations and also forwards the
1994 Subscription Creation Operations to Printer 2 with the Notification Recipient changed to be the
1995 Printer 1. When an Event occurs in Printer 2, Printer 2 sends the Event Notification (B) to
1996 Notification Recipient of Printer 1, which relays the received Event Notification (B) to the
1997 client-supplied Notification Recipient (as Event Notifications(A) in the diagram). Note, when a
1998 client performs a Subscription Creation Operation, Printer 1 need not forward the Subscription
1999 Creation Operation to Printer 2 if it would create a duplicate Subscription Object on Printer 2.

2000 Note: when Printer 1 is forwarding Subscription Creation Operations to Printer 2, it may request Printer 2 to
2001 create additional Subscription Objects (called "piggy-backing"). Piggy-backing is useful when:

- 2002 • Device A is configured to accept (IPP or non-IPP) requests from other servers.
- 2003 • Server S wants to receive Job Events that the client didn't request and Server S wants these Events
2004 for jobs it submits and not for other jobs.

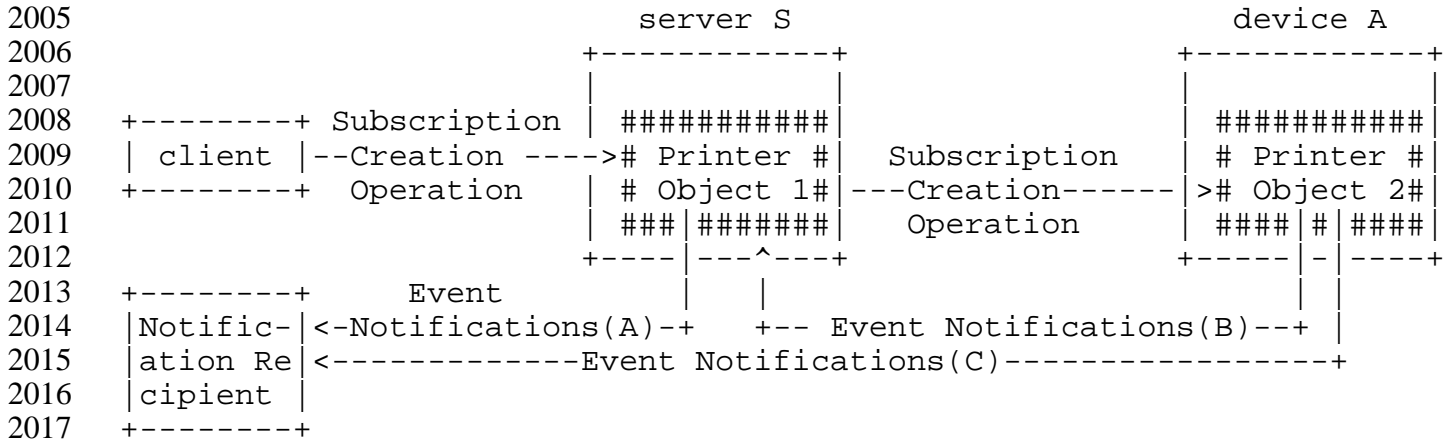


Figure 2 – Model for Notification with Cascading Printers

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.

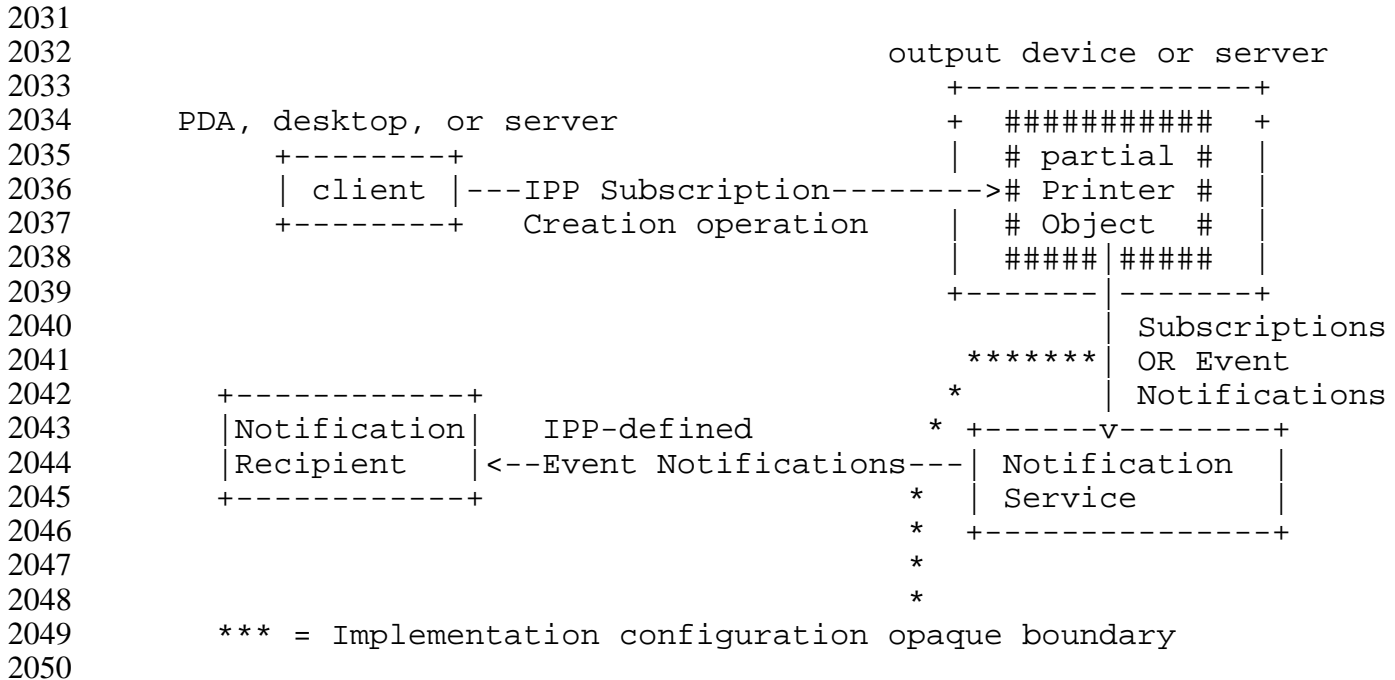


Figure 3 – Opaque Use of a Notification Service Transparent to the Client

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.

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

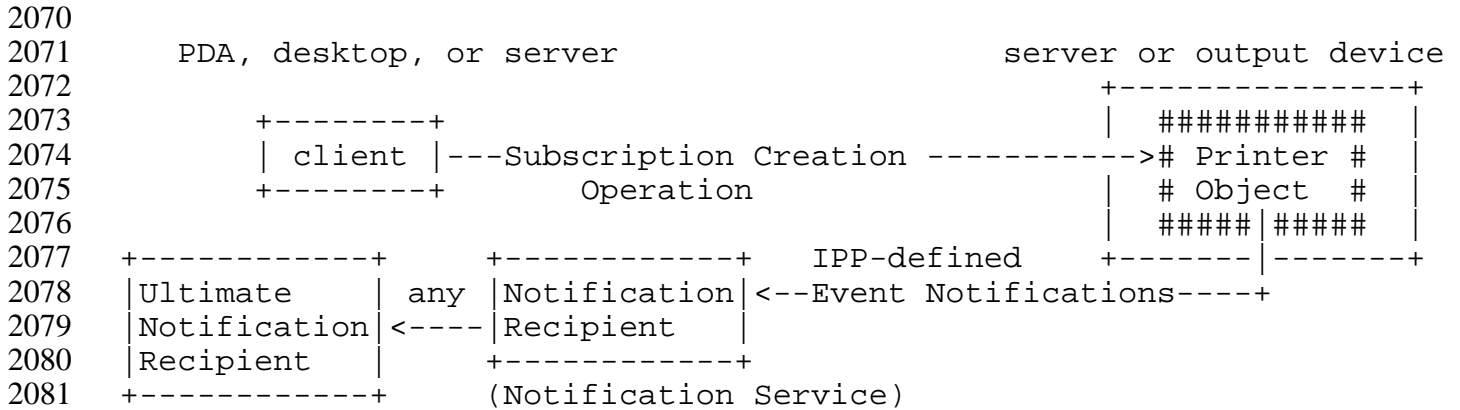


Figure 4 – Use of an Extended Notification Recipient transparent to the Printer

D. Appendix - Details about Conformance Terminology

The following paragraph provide more details about conformance terminology.

REQUIRED - an adjective used to indicate that a conforming IPP Printer implementation **MUST** support the indicated operation, object, attribute, attribute value, status code, or out-of-band value in requests and responses. See [ipp-mod] “Appendix A - Terminology for a definition of “support”. *Since support of this entire Notification specification is OPTIONAL for conformance to IPP/1.0 or IPP/1.1, the use of the term REQUIRED in this document means “REQUIRED if this OPTIONAL Notification specification is implemented”.*

RECOMMENDED - an adjective used to indicate that a conforming IPP Printer implementation is recommended to support the indicated operation, object, attribute, attribute value, status code, or out-of-band value in requests and responses. *Since support of this entire Notification specification is OPTIONAL for conformance to IPP/1.0 or IPP/1.1, the use of the term RECOMMENDED in this document means “RECOMMENDED if this OPTIONAL Notification specification is implemented”.*

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

E. Appendix - Object Model for Notification

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

2103 The object relationships can be seen pictorially as:

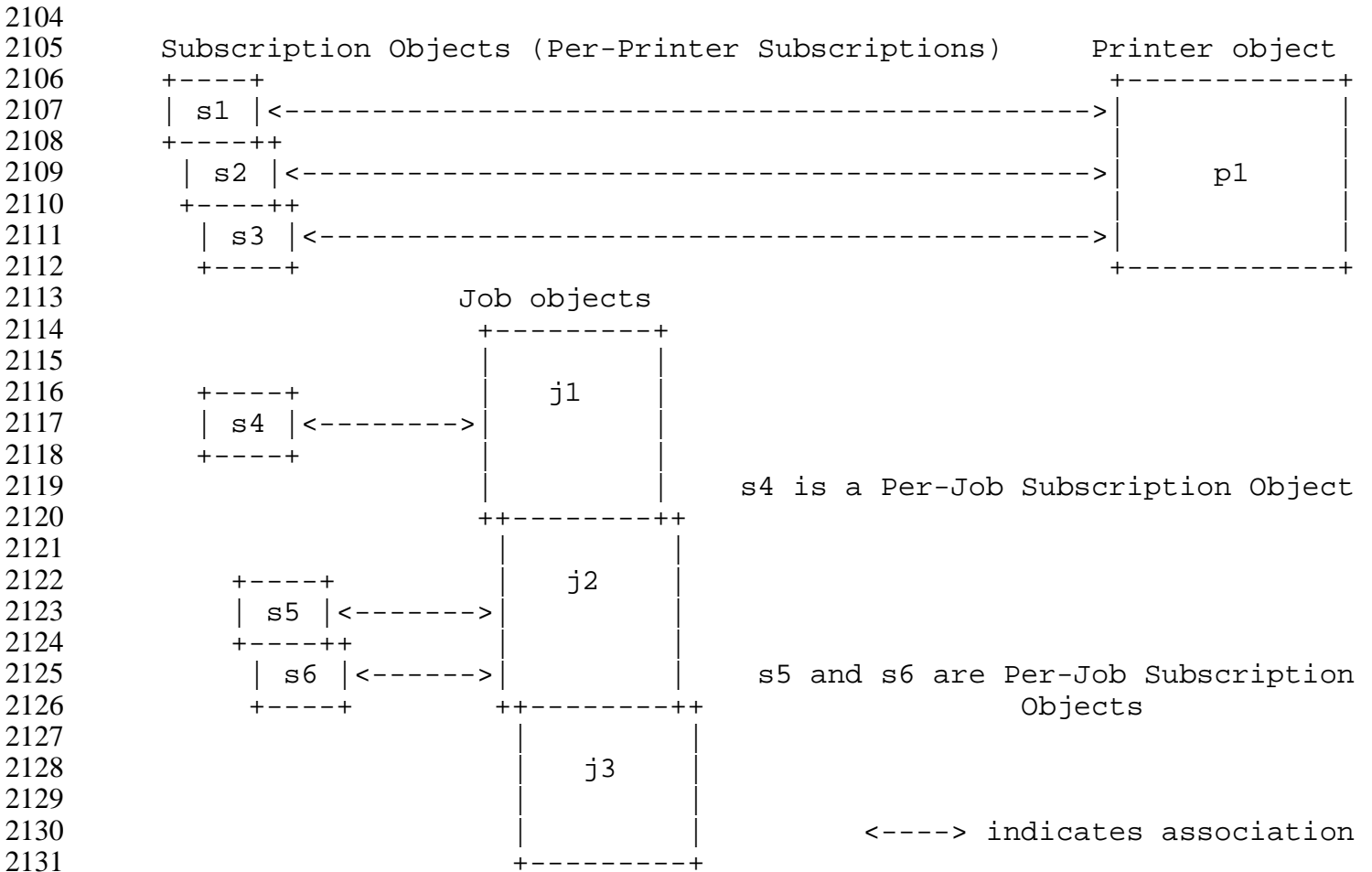


Figure 5 – Object Model for Notification

2133 s1, s2, and s3 are Per-Printer Subscription Objects and can identify Printer and/or Job Events.
2134 s4, s5, and s6 are Per-Job Subscription Objects and can identify Printer and/or Job Events.

2135 **E.1 Appendix - Object relationships**

2136 This sub-section defines the object relationships between the Printer, Job, and Subscription Objects by
2137 example. Whether Per-Printer Subscription Objects are actually contained in a Printer object or are just bi-
2138 directionally associated with them in some way is IMPLEMENTATION DEPENDENT and is transparent
2139 to the client. Similarly, whether Per-Job Subscription Objects are actually contained in a Job object or are
2140 just bi-directionally associated with them in some way is IMPLEMENTATION DEPENDENT and is
2141 transparent to the client. The object relationships are defined as follows:

2142 **E.2 Printer Object and Per-Printer Subscription Objects**

- 2143 1. The Printer object contains (is associated with) zero or more Per-Printer Subscription Objects (p1
2144 contains s1-s3 Per-Printer Subscription Objects).

- 2145 2. Each Per-Printer Subscription Object (s1, s2, and s3) is contained in (or is associated with) exactly
2146 one Printer object (p1).

2147 **E.3 Job Object and Per-Job Subscription Objects**

- 2148 1. A Job object (j1, j2, j3) is associated with zero or more Per-Job Subscription Objects (s4-s6). Job j1
2149 is associated with Per-Job Subscription Object s4, Job j2 is associated with Per-Job Subscription
2150 Objects s5 and s6, and Job j3 is not associated with any Per-Job Subscription Object.
- 2151 2. Each Per-Job Subscription Object is associated with exactly one Job object.

2152 **F. Appendix - Per-Job versus Per-Printer Subscription Objects**

2153 Per-Job and Per-Printer Subscription Objects are quite similar. Either type of Subscription Object can
2154 subscribe to Job Events, Printer Events, or both. Both types of Subscription Objects can be queried using
2155 the Get-Subscriptions and Get-Subscription-Attributes operations and canceled using the Cancel-
2156 Subscription operation. Both types of Subscription Objects create Subscription Objects which have the
2157 same Subscription Object attributes defined. However, there are some semantic differences between Per-
2158 Job Subscription Objects and Per-Printer Subscription Objects. A Per-Job Subscription Object is
2159 established by the client when submitting a job and after creating the job using the Create-Job-
2160 Subscriptions operation by specifying the “job-id” of the Job with the “notify-job-id” attribute. A Per-
2161 Printer Subscription Object is established between a client and a Printer using the Create-Printer-
2162 Subscriptions operation. Some specific differences are:

- 2163 1. A client usually creates one or more Per-Job Subscription Objects as part of the Job Creation operations
2164 (Create-Job, Print-Job, and Print-URI), rather than using the OPTIONAL Create-Job-Subscriptions
2165 operation, especially since Printer implementations NEED NOT support the Create-Job-Subscriptions
2166 operation, since it is OPTIONAL.
- 2167 2. For Per-Job Subscription Objects, the Subscription Object is only valid while the job is “not-complete”
2168 (see sections 5.4.3) while for the Per-Printer Subscription Objects, the Subscription Object is valid until
2169 the time (in seconds) that the Printer returned in the “notify-lease-expiration-time” operation attribute.
- 2170 3. Job Events in a Per-Job Subscription Object apply only to “one job” (the Job created by the Job
2171 Creation operation or references by the Create-Job-Subscriptions operation) while Job Events in a Per-
2172 Printer Subscription Object apply to ALL jobs contained in the IPP Printer.

2173 **G. Appendix: Change History (to be removed for Internet-Draft)**

2174 This section summarizes the changes to the document. Each sub-section is in *reverse* chronological order.
2175 Adding or removing ISSUES that don't change the document are not listed here.

2176 **G.1 Changes to the May 10, 2000 version to create the June [29,30](#), 2000 version**

2177 The following changes were made to the May 10, 2000 version to create the June [29,30](#), 2000 version based
2178 on the agreements reached at the May IPP WG meetings and subsequent teleconferences:

- 2179 1. Editorially reorganized and revised the document so that information is stated only once. Moved
2180 supplementary material to appendices.
- 2181 2. Cleaned up the terminology so that it is used consistently throughout the document; capitalized such
2182 terms. Simplified the descriptions of each term.
- 2183 3. Recast the Subscription attributes to be Subscription Template and Subscription Description attributes
2184 following the IPP/1.1 model for Jobs. Therefore, a few attribute names were changed to make them
2185 consistent.
- 2186 4. Reworked the operation descriptions to align with the style in [ipp-mod].
- 2187 5. Made the validation and processing of Subscription Template attributes be the same for Job Creation
2188 Operations, Create-Job-Subscriptions, and Create-Printer-Subscriptions operations (and defined in one
2189 place) and as similar to validation of jobs as possible (though there are some differences since one
2190 request can generate multiple Subscription objects.
- 2191 6. Clarified the error handling for all operations.
- 2192 7. Removed the “notify-text-format” and “notify-additional-formats” Subscription Template attributes and
2193 added the “notify-job-id” Subscription Description attribute.
- 2194 8. The client can supply one or more Subscription Template Attribute Groups in all Subscription Creation
2195 requests and the printer returns Subscription Object Attributes groups for each Subscription object
2196 created. Consequently, an “s” was added to Create-Job-Subscriptions and Create-Printer-Subscriptions
2197 operations.
- 2198 9. Reorganized the Events, so that some of the Events represent a group of events and the rest are sub-
2199 events. This reduces the number of Subscribed Events that a Printer needs to support in one
2200 Subscription from 5 to 2. It also means that the event that is delivered is one of the Subscribed events,
2201 not necessarily the trigger event, so “notify-trigger-event” was renamed to “notify-subscribed-event” in
2202 the Event Notification.
- 2203 10. Added the ‘printer-full’ and ‘printer-not-almost-idle’ Events to go along with the ‘printer-no-longer-
2204 full’ and ‘printer-almost-idle’ Events. Renamed the ‘printer-queue-changed’ Event to ‘printer-queue-
2205 order-changed’.
- 2206 11. Clarified what MUST be in a Delivery Method Document.
- 2207 12. Removed “persistent-jobs-supported” Printer Description attribute, since it has nothing to do with
2208 Notifications and is not needed to describe Subscription object persistence.
- 2209 13. Changed notify-max-printer-subscriptions-supported (integer(0:MAX)) and notify-max-job-
2210 subscriptions-supported (integer(0:MAX)) so that MAX means no limit and 0 means no subscriptions
2211 are (currently) allowed, so as to give a way to turn off accepting new subscriptions.

2212 **G.2 Changes to the March 8, 2000 version to create the May 10, 2000 version**

2213 The following changes were made to the March 8, 2000 version to create the May 10, 2000 version based
2214 on the agreements reached at the April IPP WG meetings and subsequent teleconferences:

- 2215 1. Change “notify-format” to “notify-text-format” and made it apply only to the format of the “notify-
2216 text” (formerly called “human-readable-report”) and Human Consumable form. A new attribute “notify-
2217 additional-formats” specifies the formats for the Machine Consumable contents of Delivery Methods
2218 that support multiple formats.
- 2219 2. Change the “job-notify” collection attribute in Job Creation operations to be multiple “notify-xxx”
2220 attributes. This change eliminates the need for collection values. It also means that a Job Creation
2221 operation can create only one Subscription Object.
- 2222 3. Change the Machine Consumable form to be transport independent.
- 2223 4. Reduce the set of REQUIRED attributes in the Machine Consumable form and add the OPTIONAL
2224 “notify-attributes” attribute that allows a client to request additional attributes.
- 2225 5. Specify the information that SHOULD be in the Human Consumable form

2226 **G.3 Changes to the March 6, 2000 version to create the March 8, 2000 version**

2227 The following changes were made to the March 6, 2000 version to create the March 8, 2000 version based
2228 on the agreements reached on the mailing list:

- 2229 1. Changed the name of the SNMP Delivery Method from ‘snmp’ to ‘snmpnotify’, since the Notification
2230 Recipient isn’t an SNMP agent.
- 2231 2. Clarified that an implementation with only a single value for persistent-jobs-supported (boolean) or
2232 persistent-subscriptions-supported (boolean) MAY make it settable to the single value or make it not-
2233 settable.

2234 **G.4 Changes to the February 2, 2000 version to create the March 6, 2000 version**

2235 The following changes were made to the February 2, 2000 version to create the March 6, 2000 version
2236 based on the agreements reached on the mailing list, at the February IPP WG meetings, and reflected in the
2237 minutes:

- 2238 1. Clarified that this extension is intended as an extension to IPP/1.0, IPP/1.1, and future versions.
- 2239 2. Allocated the operation-id 0x0016 to 0x001B values for the Notification operations defined in the
2240 document.
- 2241 3. Pre-pended the word “subscription-” on the front of the “request-id” Subscription Object attribute to
2242 distinguish it from the “request-id” parameter that is sent in every request and response.
- 2243 4. Added the term “settable” for describing attributes that are not READ-ONLY.

- 2244 5. Added the term “Subscription Creation Operation” to stand for any operation that can create a
2245 Subscription Object: Job Creation operations (Create-Job, Print-Job, and Print-URI), Create-Job-
2246 Subscriptions, and Create-Printer-Subscriptions.
- 2247 6. Changed the “subscriber-user-name” (name(MAX)) Subscription Object attribute from OPTIONAL to
2248 REQUIRED.
- 2249 7. Changed the name and semantics of “notify-printer-up-time(integer(1:MAX)) to notify-server-up-time
2250 so that it can be either the Printer’s uptime or a Notification Delivery Service uptime.
- 2251 8. Added the ‘ipp:’, ‘indp:’, ‘mailto:’, and ‘snmp:’ notification delivery schemes to the definition of the
2252 “notify-recipients” to indicate possible schemes.
- 2253 9. Changed the name and semantics of “notify-text-format” (mimeMediaType) to “notify-format” so that it
2254 can be used to specify either Human Consumable or Machine Consumable formats where the
2255 implementation supports both. Clarified that this attribute controls whatever variable Notification
2256 Content that the implementation supports, which may be an attachment to the fixed content format or
2257 the contents of the “human-readable-report” (text(MAX)) attribute. Clarified that an implementation
2258 NEED NOT support all of its supported Notification Content formats with all of its supported Delivery
2259 Methods.
- 2260 10. Added ‘text/xml’, ‘application/ipp’, ‘application/postscript’, and ‘image/tiff’ and additional example
2261 MIME media types for “notify-format” (mimeMediaType).
- 2262 11. Clarified that the recommend way for a client to determine whether or not a Printer supports Per-Job
2263 Subscriptions is to query the Printer’s “notify-max-job-subscriptions-supported” attribute, since Create-
2264 Job-Subscriptions is an OPTIONAL operation.
- 2265 12. Clarified that the recommend way for a client to determine whether or not a Printer supports Per-Printer
2266 Subscriptions is to query the Printer’s “operations-supported” attribute to see if the Create-Printer-
2267 Subscriptions operations is supported, since this is the usual way to determine a Printer’s capabilities.
- 2268 13. Clarified that if “persistent-jobs-supported” (boolean) and “persistent-subscriptions-supported”
2269 (boolean) are settable, then setting them must affect whether or not jobs and subscriptions are persistent.
- 2270 14. Allowed Delivery Methods to send operations with or without a response, depending on the definition
2271 of the Delivery Method.
- 2272 15. Indicated that a deliver method definition is free to REQUIRE that the client supply the “notify-user-
2273 data” attribute.
- 2274 16. Required that the Printer support the “job-uri” operation attribute as a target, in addition to “printer-uri”
2275 & “job-id”, i.e., keep consistent with all Job operations.
- 2276 17. Changed the ‘none’ out-of-band value to be a reference to the collection document [ipp-coll], since the
2277 use for it in this document is with the ‘collection’ attribute syntax.

- 2278 18. Clarified that a conforming implementation MUST support the 'collection' attribute syntax, since that is
2279 required in Job Creation operations.
- 2280 19. Allocated the values to the new status codes defined in this document.
- 2281 20. Allocated the [ipp-pro] subscription-attributes-tag and notification-attributes-tag delimiter tags to
2282 delimit Subscription attributes and Notification Content attributes in requests and responses.
- 2283 21. Changed the 'server-error-too-many-subscriptions' and 'server-error-too-many-events' to be client
2284 errors, i.e., 'client-error-too-many-subscriptions' and 'client-error-too-many-events', since other errors
2285 of this type are client errors.

2286 **G.5 Changes to the October 14, 1999 version to create the February 2, 2000 version**

2287 The following changes were made to the October 14, 1999 version to create the February 2, 2000 version
2288 based on the agreements reached at the October and December IPP WG meetings and reflected in the
2289 minutes:

- 2290 1. Added a Java Listener as an example of a Notification Recipient.
- 2291 2. Clarified the object relationships.
- 2292 3. Clarified how job Events differ for Per-Job versus Per-Printer Subscriptions.
- 2293 4. Added the ability for the Machine Consumable form to contain a Human Readable "human-readable-
2294 report" (text) attribute so that both forms could be sent in the same Notification.
- 2295 5. Clarified that the 'none' value for notify-text-format (mimeType) has to be out-of-band, not the
2296 text string 'none' as a mimeType.
- 2297 6. Clarified that 'none' means send the Machine Consumable form without the "human-readable-report"
2298 (text) attribute, if it is defined.
- 2299 7. Clarified that Notification Recipients MUST be able to accept unrecognized attributes.
- 2300 8. Allowed the notification Delivery Method definition to be modeled as (1) a request with an operation
2301 code without a response, (2) a request with a operation code with a response or (3) a response with a
2302 status code.
- 2303 9. Added "notify-text-format" (mimeType) and "human-readable-report" (text(MAX)) to be able to
2304 be sent in a Notification content, if the notification Delivery Method Document permits it.
- 2305 10. Added "job-k-octets" (integer(0:MAX)), "job-impressions" (integer(0:MAX)), and "job-media-sheets"
2306 (integer(0:MAX)) as OPTIONAL for Notification content for use in job-progress Events to show the
2307 target values so that the Notification Recipient can show a thermometer.
- 2308 11. Added a Subscription Attributes Group (and subscription-attributes tag) the Create-Job-Subscriptions
2309 and Create-Printer-Subscriptions requests and responses.

- 2310 12. Added the 'none' out-of-band value for use with "notify-text-format" (mimeType) attribute.
- 2311 13. Changed the job progress attributes from using -2 to mean 'unknown' as in the PWG Job Monitoring
- 2312 MIB, to use the 'unknown' out-of-band value.

2313

2314 **H. Appendix: Full Copyright Statement**

2315 Copyright (C) The Internet Society (1998,1999,2000). All Rights Reserved

2316 This document and translations of it may be copied and furnished to others, and derivative works that
2317 comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and
2318 distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and
2319 this paragraph are included on all such copies and derivative works. However, this document itself may not
2320 be modified in any way, such as by removing the copyright notice or references to the Internet Society or
2321 other Internet organizations, except as needed for the purpose of developing Internet standards in which
2322 case the procedures for copyrights defined in the Internet Standards process must be followed, or as
2323 required to translate it into languages other than English.

2324 The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its
2325 successors or assigns.

2326 This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET
2327 SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES,
2328 EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE
2329 OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED
2330 WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

2331