

1 INTERNET-DRAFT
2 <draft-ietf-ipp-not-spec-065.txt>
3 [Target Category: standards track]
4

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
~~January 24, 2000~~ ~~August 30, 2000~~

16 Internet Printing Protocol (IPP):
17 **IPP Event Notification Specification**
18

19 Copyright (C) The Internet Society (20010). All Rights Reserved.

20 Status of this Memo

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

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

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

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

29 **Abstract**

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

37 defined for Subscription Objects: Get-Subscriptions-Attributes, Get-Subscriptions, Renew-Subscription, and
38 Cancel-Subscription.
39

39

40 The basic set of IPP documents includes:

41 Design Goals for an Internet Printing Protocol [RFC2567]

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

43 Internet Printing Protocol/1.1: Model and Semantics [~~IPP-MOD~~[RFC2911](#)]

44 Internet Printing Protocol/1.1: Encoding and Transport [~~IPP-PRO~~[RFC2910](#)]

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

46 Mapping between LPD and IPP Protocols [RFC2569]

47

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

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

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

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

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

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

74

74 **Table of Contents**

75	1	Introduction.....	8
76	1.1	Notification Overview	8
77	2	Models for Notification.....	10
78	2.1	Model for Notification (Simple Case).....	10
79	2.2	Model for Notification with Cascading Printers	11
80	2.3	Distributed Model for Notification.....	11
81	2.4	Extended Notification Recipient.....	11
82	3	Terminology	11
83	3.1	Conformance Terminology	12
84	3.2	Other Terminology.....	12
85	4	Object Relationships.....	14
86	4.1	Printer and Per-Printer Subscription Objects	14
87	4.2	Printer, Job and Per-Job Subscription Objects	14
88	5	Subscription Object.....	14
89	5.1	Rules for Support of Subscription Template Attributes	15
90	5.2	Rules for Processing Subscription Template Attributes	16
91	5.3	Subscription Template Attributes.....	19
92	5.3.1	notify-recipient-uri (uri)	19
93	5.3.2	notify-events (1setOf type2 keyword)	20
94	5.3.2.1	Standard Values for Subscribed Events	20
95	5.3.2.1.1	No Events	21
96	5.3.2.1.2	Subscribed Printer Events	21
97	5.3.2.1.3	Subscribed Job Events	22
98	5.3.2.2	Rules for Matching of Subscribed Events.....	23
99	5.3.2.2.1	Rules for Matching of Printer Events	23
100	5.3.2.2.2	Rules for Matching of Job Events	23
101	5.3.2.2.3	Special Cases for Matching Rules	24
102	5.3.3	notify-attributes (1setOf type2 keyword)	25
103	5.3.4	notify-user-data (octetString(63))	26
104	5.3.5	notify-charset (charset).....	26
105	5.3.6	notify-natural-language (naturalLanguage).....	26
106	5.3.7	notify-lease-duration (integer(0:67108863)).....	27
107	5.3.8	notify-time-interval (integer(0:MAX))	28
108	5.4	Subscription Description Attributes.....	29
109	5.4.1	notify-subscription-id (integer (1:MAX)).....	29
110	5.4.2	notify-sequence-number (integer (0:MAX)).....	29
111	5.4.3	notify-lease-expiration-time (integer(0:MAX))	30
112	5.4.4	notify-printer-up-time (integer(1:MAX)).....	31
113	5.4.5	notify-printer-uri (uri)	31
114	5.4.6	notify-job-id (integer(1:MAX)).....	31
115	5.4.7	notify-subscriber-user-name (name(MAX)).....	31
116	6	Printer Description Attributes Related to Notification.....	32
117	6.1	printer-state-change-time (integer(1:MAX)).....	32
118	6.2	printer-state-change-date-time (dateTime).....	32

119	7	New Values for Existing Printer Description Attributes	33
120	7.1	operations-supported (1setOf type2 enum).....	33
121	8	Attributes Only in Event Notifications	33
122	8.1	notify-subscribed-event (type2 keyword)	33
123	8.2	notify-text (text(MAX)).....	34
124	9	Event Notification Content	34
125	9.1	Content of Machine Consumable Event Notifications	35
126	9.1.1	Event Notification Content Common to All Events	36
127	9.1.2	Additional Event Notification Content for Job Events	37
128	9.1.3	Additional Event Notification Content for Printer Events.....	37
129	9.2	Content of Human Consumable Event Notification.....	37
130	9.2.1	Event Notification Content Common to All Events	38
131	9.2.2	Additional Event Notification Content for Job Events	39
132	9.2.3	Additional Event Notification Content for Printer Events.....	40
133	10	Delivery Methods.....	40
134	11	Operations for Notification	42
135	11.1	Subscription Creation Operations	42
136	11.1.1	Create-Job-Subscriptions Operation.....	42
137	11.1.1.1	Create-Job-Subscriptions Request	43
138	11.1.1.2	Create-Job-Subscriptions Response.....	43
139	11.1.2	Create-Printer-Subscriptions operation.....	45
140	11.1.2.1	Create-Printer-Subscriptions Request.....	45
141	11.1.2.2	Create-Printer-Subscriptions Response	45
142	11.1.3	Job Creation Operation – Extensions for Notification.....	45
143	11.1.3.1	Job Creation Request.....	46
144	11.1.3.2	Job Creation Response	46
145	11.2	Other Operations	47
146	11.2.1	Validate-Job Operation - Extensions for Notification	47
147	11.2.2	Get-Printer-Attributes - Extensions for Notification.....	48
148	11.2.3	Get-Subscription-Attributes operation.....	48
149	11.2.3.1	Get-Subscription-Attributes Request	48
150	11.2.3.2	Get-Subscription-Attributes Response.....	49
151	11.2.4	Get-Subscriptions operation.....	50
152	11.2.4.1	Get-Subscriptions Request.....	51
153	11.2.4.2	Get-Subscriptions Response	52
154	11.2.5	Renew-Subscription operation	53
155	11.2.5.1	Renew-Subscription Request	53
156	11.2.5.2	Renew-Subscription Response.....	54
157	11.2.6	Cancel-Subscription operation	55
158	11.2.6.1	Cancel-Subscription Request	55
159	11.2.6.2	Cancel-Subscription Response.....	56
160	12	Conformance Requirements.....	56
161	13	IANA Considerations.....	57
162	13.1	Attribute Registrations	57
163	13.2	Keyword Attribute Value Registrations	58
164	13.3	Operation Registrations	59
165	13.4	Status code Registrations	59
166	13.5	Attribute Group tag Registrations.....	60

167	13.6	Format for Event Notification Delivery Method Registration proposals	60
168	13.7	Format and Requirements for IPP Delivery Method Registration Proposals	61
169	14	Internationalization Considerations	61
170	15	Security Considerations	62
171	16	Status Codes.....	62
172	16.1	successful-ok-ignored-subscriptions (0x0003).....	62
173	16.2	client-error-ignored-all-subscriptions (0x0414).....	63
174	17	Status Codes in Subscription Attributes Groups	63
175	17.1	client-error-uri-scheme-not-supported (0x040C).....	63
176	17.2	client-error-too-many-subscriptions (0x0415)	63
177	17.3	successful-ok-too-many-events (0x0005).....	63
178	17.4	successful-ok-ignored-or-substituted-attributes (0x0001)	63
179	18	Encodings of Additional Attribute Tags	64
180	19	References	64
181	20	Author's Addresses.....	65
182	A.	Appendix - Model for Notification with Cascading Printers	67
183	B.	Appendix - Distributed Model for Notification.....	68
184	C.	Appendix - Extended Notification Recipient.....	69
185	D.	Appendix - Details about Conformance Terminology.....	70
186	E.	Appendix - Object Model for Notification.....	71
187	E.1	Appendix - Object relationships.....	73
188	E.2	Printer Object and Per-Printer Subscription Objects.....	73
189	E.3	Job Object and Per-Job Subscription Objects.....	73
190	F.	Appendix - Per-Job versus Per-Printer Subscription Objects	73
191	G.	Appendix: Full Copyright Statement.....	74
192			
193	Tables		
194	Table 1	– Subscription Template Attributes.....	19
195	Table 2	– Subscription Description Attributes.....	29
196	Table 3	– Printer Description Attributes Associated with Notification.....	32
197	Table 4	– Operation-id assignments	33
198	Table 5	– Attributes in Event Notification Content	36
199	Table 6	– Additional Event Notification Content for Job Events	37
200	Table 7	– Combinations of Events and Subscribed Events for “job-impressions-completed”	37
201	Table 8	– Additional Event Notification Content for Printer Events.....	37
202	Table 9	– Printer Name in Event Notification Content	39
203	Table 10	– Event Name in Event Notification Content	39
204	Table 11	– Event Time in Event Notification Content	39
205	Table 12	– Job Name in Event Notification Content	39

206 Table 13 – Job State in Event Notification Content 40
207 Table 14 – Printer State in Event Notification Content 40
208 Table 15 – Information about the Delivery Method..... 41
209 Table 16 – Conformance Requirements for Operations..... 57
210 **Figures**
211 Figure 1 – Model for Notification..... 10
212 Figure 2 – Model for Notification with Cascading Printers 68
213 Figure 3 – Opaque Use of a Notification Service Transparent to the Client..... 69
214 Figure 4 – Use of an Extended Notification Recipient transparent to the Printer 70
215 Figure 5 – Object Model for Notification 72
216
217

217 1 Introduction

218 This IPP notification specification is an extension to IPP/1.0 [RFC2568, RFC2569] and IPP/1.1 [~~ipp-~~
219 ~~mod~~RFC2911, ~~ipp-pro~~RFC2910]. This document in combination with the following documents is intended
220 to meet the notification requirements described in [ipp-not-req]:

221 Internet Printing Protocol (IPP): “Job Progress Attributes” [ipp-prog]
222 One or more Delivery Method Documents registered with IANA (see section 13).
223

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

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

227 1.1 Notification Overview

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

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

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

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

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

263 For each of the above operations:

- 264 • the Printer associates a Subscription Object with the Printer or a specific Job. When a Subscription
265 Object is associated with a Job Object, it is called a *Per-Job Subscription Object*. When a
266 Subscription Object is associated with a Printer Object, it is called a *Per-Printer Subscription*
267 *Object*.
- 268 • the response contains one Subscription Attributes Group for each Subscription Template Attributes
269 Group in the request and in the same order. When the Printer successfully creates a Subscription
270 Object, its corresponding Subscription Attributes Group contains the “notify-subscription-id”
271 attribute. This attribute uniquely identifies the Subscription Object and is analogous to a “job-id” for
272 a Job object. Some operations described below use the “notify-subscription-id” to identify the target
273 Subscription Object.

274 This document defines the following additional operations (see section 11.2 for further details):

- 275 • **Validate-Job operation:** When a client performs this operation, a client can include zero or more
276 Subscription Template Attributes Groups in the request. The Printer determines if it could create one
277 Subscription Object for each Subscription Template Attributes Group in the request. This document
278 extends this operation's definition in [~~ipp-mod~~[RFC2911](#)] by adding Subscription Template
279 Attributes Groups in the request and Subscription Attributes Groups in the response.
- 280 • **Get-Subscription-Attributes operation:** This operation allows a client to obtain the specified
281 attributes of a target Subscription Object.
- 282 • **Get-Subscriptions operation:** This operation allows a client to obtain the specified attributes of all
283 Subscription Objects associated with the Printer or a specified Job.

- 284 • **Renew-Subscription operation:** This operation renews the lease on the target Per-Printer
 285 Subscription Object before it expires. A newly created Per-Printer Subscription Object receives an
 286 initial lease. It is the duty of the client to use this operation frequently enough to preserve a Per-
 287 Printer Subscription Object. The Printer deletes a Per-Printer Subscription Object when its lease
 288 expires. A Per-Job Subscription Object last exactly as long as its associated Job Object and thus
 289 doesn't have a lease.
- 290 • **Cancel-Subscription operation:** This operation cancels the lease on the specified Per-Printer
 291 Subscription Object and thereby deletes the Subscription Object.

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

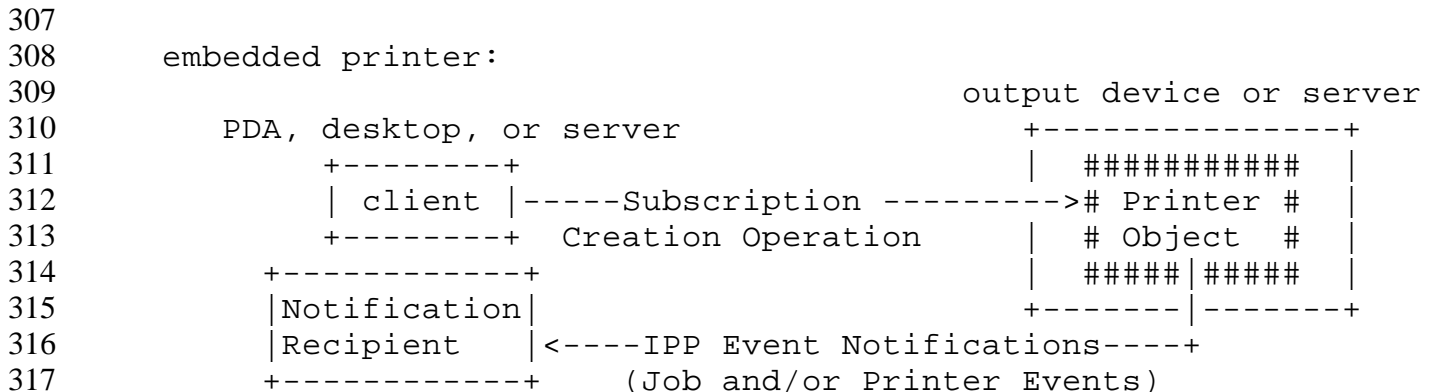
- 294 a) generates an Event Notification with information specified in section 9, AND
 295 b) either:
- 296 i) delivers the Event Notification using the Delivery Method and target address identified in the
 297 Subscription Object's "notify-recipient-uri" attribute if the Delivery Method is a "push", OR
 298 ii) saves Event Notification for a time period defined by the Delivery Method if the Delivery
 299 Method is a "pull", i.e., the Notification Recipient is expected to fetch the Event Notifications.

300 2 Models for Notification

301 2.1 Model for Notification (Simple Case)

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

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



318 **Figure 1 – Model for Notification**

319 2.2 Model for Notification with Cascading Printers

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

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

325 See Appendix A for more details.

326 2.3 Distributed Model for Notification

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

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

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

336 Appendix B describes this example in detail.

337 2.4 Extended Notification Recipient

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

342 Appendix C describes this example in detail.

343 3 Terminology

344 This section defines terminology used throughout this document. Other terminology is defined in [~~ipp-~~
345 ~~mod~~[RFC2911](#)].

346 3.1 Conformance Terminology

347 Capitalized terms, such as **MUST**, **MUST NOT**, **REQUIRED**, **SHOULD**, **SHOULD NOT**, **MAY**,
348 **NEED NOT**, and **OPTIONAL**, have special meaning relating to conformance to this specification. These
349 terms are defined in [\[ipp-modRFC2911\]](#) section 13.1 on conformance terminology, most of which is taken
350 from RFC 2119 [RFC2119]. See Appendix D for complete details.

351 Note: a feature that is **OPTIONAL** in this document becomes **REQUIRED** if the Printer implements a
352 Delivery Method that **REQUIRES** the feature

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

357 3.2 Other Terminology

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

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

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

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

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

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

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

373 **Subscribed Event** – an Event that the Subscribing Client expresses interest in by making it a value of the
374 “notify-events” attribute on a Subscription Object.

375 **Subscribed Job Event** – a Subscribed Event that is a Job Event.

376 **Subscribed Printer Event** – a Subscribed Event that is a Printer Event.

- 377 **Event Notification** - the information about an Event that the Printer sends when an Event occurs.
- 378 **Notification Recipient** - the entity to which the Printer sends an Event Notification.
- 379 **Delivery Method** - the mechanism by which the Printer delivers the Event Notification, e.g., via email or via
380 SNMP.
- 381 **Delivery Method Document** - a document, separate from this document, that defines a Delivery Method.
- 382 **Compound Event Notification** - two or more Event Notifications that a Printer sends together as a single
383 entity. The Delivery Method Document specifies whether the Delivery Method supports Compound
384 Event Notifications.
- 385 **Subscription Object** - An object containing a set of attributes that indicate: the Notification Recipient, the
386 Delivery Method, the Subscribed Events that cause the Printer to send an Event Notification, and the
387 information to send in an Event Notification.
- 388 **Per-Job Subscription Object** - A Subscription Object that is associated with a single Job. The Create-Job-
389 Subscriptions operation and Job Creation operations create such an object.
- 390 **Per-Printer Subscription Object** - A Subscription Object that is associated with the Printer as a whole. The
391 Create-Printer-Subscriptions operation creates such an object.
- 392 **Subscribing Client** - The client that creates the Subscription Object.
- 393 **Subscription Creation Operation** - An operation that creates a Subscription Object: Job Creation
394 operations, Create-Job-Subscriptions operation, and Create-Printer-Subscriptions operation. In the
395 context of a Job Creation operation, a Subscription Creation Operation is the part of the Job Creation
396 operation that creates a Subscription object.
- 397 **Subscription Creation Request** – The request portion of a Subscription Creation Operation.
- 398 **Subscription Template Attributes** – Subscription Object attributes that a client can supply in a
399 Subscription Creation Operation and associated Printer Object attributes that specify supported and
400 default values for the Subscription Object attributes.
- 401 **Subscription Description Attributes** – Subscription Object attributes that a Printer supplies during a
402 Subscription Creation Operation.
- 403 **Subscription Template Attributes Group** – The attributes group in a request that contains Subscription
404 Object attributes that are Subscription Template Attributes.
- 405 **Subscription Attributes Group** – The attributes group in a response that contains Subscription Object
406 attributes.

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

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

411 **Printer** – the software that supports an output device or print server (see IPP/1.1 [~~ipp-mod~~[RFC2911](#)] which
412 uses the terms Printer and Printer object interchangeably). This document extends the IPP/1.1 Printer
413 definition to include the software that implements Subscription Creation Operations and the sending of
414 Event Notifications, even if the software for such a Printer would be distributed across a network (see
415 section 2.3).

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

418 **4 Object Relationships**

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

421 **4.1 Printer and Per-Printer Subscription Objects**

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

424 **4.2 Printer, Job and Per-Job Subscription Objects**

- 425 1. A Printer object is associated with zero or more Job objects.
- 426 2. Each Job object is associated with exactly one Printer object.
- 427 3. A Job object is associated with zero or more Per-Job Subscription Objects.
- 428 4. Each Per-Job Subscription Object is associated with exactly one Job object.

429 **5 Subscription Object**

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

434 Using the IPP Job Template attributes as a model (see [~~ipp-mod~~[RFC2911](#)] section 4.2), the attributes of a
435 Subscription Object are divided into two categories: Subscription Template Attributes and Subscription
436 Description Attributes.

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

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

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

443 5.1 Rules for Support of Subscription Template Attributes

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

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

- 448 1. Each attribute's name starts with the prefix string "notify-" and this document calls such attributes
449 "notify-xxx".
- 450 2. For each "notify-xxx" Subscription Object attribute defined in column 1 of Table 1 in section 5.3,
451 Table 1 specifies corresponding Printer attributes: "notify-xxx-default", "notify-xxx-supported", "yyy-
452 supported" and "notify-max-xxx-supported" defined in column 2 of Table 1. Note "xxx" stands for the
453 same string in each case and "yyy" stands for some other string.
- 454 3. If a Printer supports "notify-xxx" in column 1 of Table 1, then the Printer MUST support all associated
455 attributes specified in column 2 of Table 1. For example, Table 1 shows that if the Printer supports
456 "notify-events", it MUST support "notify-events-default", "notify-events-supported" and "notify-max-
457 events-supported".
- 458 4. If a Printer does not support "notify-xxx" in column 1 of Table 1, then the Printer MUST NOT support
459 any associated "notify-yyy" attributes specified in column 2 of Table 1. For example, Table 1 shows
460 that if the Printer doesn't support "notify-events", it MUST NOT support "notify-events-default",
461 "notify-events-supported" and "notify-max-events-supported". Note this rule does not apply to
462 attributes whose names do not start with the string "notify-" and are thus defined in another object and
463 used by other attributes.
- 464 5. Most "notify-xxx" attributes have a corresponding "yyy-supported" attribute that specifies the
465 supported values for "notify-xxx". Column 2 of Table 1 specifies the name of each "yyy-supported"

466 attribute. The naming rules of IPP/1.1 (see [~~ipp-mod~~[RFC2911](#)]) are used when “yyy-supported” is
467 “notify-xxx-supported”.

468 6. Some “notify-xxx” attributes have a corresponding “notify-xxx-default” attribute that specifies the value
469 for “notify-xxx” if the client does not supply it. Column 2 of Table 1 specifies the name of each “notify-
470 xxx-default” attribute. The naming rules of IPP/1.1 (see [~~ipp-mod~~[RFC2911](#)]) are used.

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

478 5.2 Rules for Processing Subscription Template Attributes

479 This section defines a detailed set of rules that a Printer follows when it processes Subscription Template
480 Attributes in a Subscription Creation Request. These rules for are similar to the rules for processing Operation
481 attributes in [~~ipp-mod~~[RFC2911](#)]. That is, the Printer may or may not support an attribute and a client may or
482 may not supply the attribute. Some combinations of these cases are OK. Others return warnings or errors,
483 and perhaps a list of unsupported attributes.

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

- 486 1. If a client supplies a “notify-xxx” attribute from column 1 of Table 1 and the Printer supports it and its
487 value, the Printer MUST populate the attribute on the created Subscription Object.
- 488 2. If a client supplies a “notify-xxx” attribute from column 1 of Table 1 and the Printer doesn’t support it or
489 its value, the Printer MUST NOT populate the attribute on the created Subscription Object with it. The
490 Printer MUST do one of the following:
 - 491 a) If the value of the “notify-xxx” attribute is unsupported, the Printer MUST return the attribute with its
492 value in the Subscription Attributes Group of the response.
 - 493 b) If “notify-xxx” is an unsupported attribute, the Printer MUST return the attribute in the Subscription
494 Attributes Group of the response with the ‘unsupported’ out-of-band value.

495 Note: The rules of this step are the same as for Unsupported Attributes [~~ipp-mod~~[RFC2911](#)] section
496 3.1.7. except that the unsupported attributes are returned in the Subscription Attributes Group rather than
497 the Unsupported Attributes Group because Subscription Creation Operations can create more than one
498 Subscription Object).

- 499 3. If a client is REQUIRED to supply a “notify-xxx” attribute from column 1 of Table 1 and the Printer
500 doesn’t support the supplied value, the Printer MUST NOT create a Subscription Object. The rules for
501 Unsupported Attributes in step #2 still apply.
- 502 4. If a client does not supply a “notify-xxx” attribute from column 1 of Table 1 and the attribute is
503 REQUIRED for the client to supply, the Printer MUST reject the Subscription Creation Operation
504 (including Job Creation operations) without creating a Subscription Object, and MUST return in the
505 response:
- 506 c) the status code ‘client-error-bad-request’ AND
- 507 d) no Subscription Attribute Groups.
- 508 5. If a client does not supply a “notify-xxx” attribute from column 1 of Table 1 that is OPTIONAL for the
509 client to supply, and column 2 of Table 1 either:
- 510 a) specifies a “notify-xxx-default” attribute, the Printer MUST behave as if the client had supplied the
511 “notify-xxx-default” attribute (see step #1) and populate the Subscription object with the value of the
512 “notify-xxx-default” attribute as part of the Subscription Creation operation (unlike Job Template
513 attributes where the Printer does not populate the Job object with defaults - see [~~ipp-mod~~RFC2911])
514 OR
- 515 b) does not specify a “notify-xxx-default” attribute, the Printer MUST populate the “notify-xxx” attribute
516 on the Subscription Object according to the definition of the “notify-xxx” attribute in a section 5.3. For
517 some attributes, the “notify-xxx” is populated with the value of some other attribute, and for others, the
518 “notify-xxx” is NOT populated on the Subscription object at all.
- 519 6. A Printer MUST create a Subscription Object for each Subscription Template Attributes group in a
520 request unless the Printer:
- 521 a) encounters some attributes in a Subscription Template Attributes Group that require the Printer not to
522 create the Subscription Object OR
- 523 b) would create a Per-Job Subscription Object when it doesn’t have space for another Per-Job
524 Subscription Object OR
- 525 c) would create a Per-Printer Subscription Object when it doesn’t have space for another Per-Printer
526 Subscription Object.
- 527 7. A response MUST contain one Subscription Attributes Group for each Subscription Template Attributes
528 Group in the request (and in the same order) whether the Printer creates a Subscription Object from the
529 Subscription Template Attributes Group or not. However, the attributes in each Subscription Attributes
530 Group can be in any order.
- 531 8. The Printer MUST populate each Subscription Attributes Group of the response such that each contains:

- 532 a) the “notify-subscription-id” attribute (see section 0), if and only if the Printer creates a Subscription
533 Object.
- 534 b) the “notify-lease-duration” attribute (see section 5.3.7), if and only if the Printer creates a Per-Printer
535 Subscription Object. The value of this attribute is the value of the Subscription Object’s “notify-lease-
536 duration” attribute. This value MAY be different from the client-supplied value (see section 5.3.7). If a
537 client supplies this attribute in the creation of a Per-Job Subscription Object, it MUST appear in this
538 group with the out-of-band value ‘unsupported’ to indicate that the Printer doesn’t support it in this
539 context.
- 540 c) all of the unsupported Subscription Template Attributes from step #2. Note, they are not returned in
541 the Unsupported Attributes Group in order to separate the unsupported attributes for each
542 Subscription Object.
- 543 d) the “notify-status-code” attribute if the Printer does not create the Subscription Object or if there are
544 unsupported attributes from step #2. The possible values of the “notify-status-code” attribute are
545 shown below (see section 17 for more details). The Printer returns the first value in the list below that
546 describes the status.
- 547 ‘client-error-uri-scheme-not-supported’: the Subscription Object was not created because the
548 scheme of the “notify-recipient-uri” attribute is not supported. See section 17.1 for more
549 details about this status code. See step #3 in this section for the case that causes this error,
550 and the resulting step #6a) that causes the Printer not to create the Subscription Object.
- 551 ‘client-error-too-many-subscriptions’: the Subscription Object was not created because the
552 Printer has no space for additional Subscription Objects. The client SHOULD try again later.
553 See section 17.2 for more details about this status code. See steps #6b) and #6c) in this
554 section for the cases that causes this error.
- 555 ‘successful-ok-too-many-events’: the Subscription Object was created without the “notify-
556 events” values included in this Subscription Attributes Group because the “notify-events”
557 attribute contains too many values. See section 17.3 for more details about this status code.
558 See step #2 in this section and section 5.3.2 for the cases that cause this status code.
- 559 ‘successful-ok-ignored-or-substituted-attributes’: the Subscription Object was created but some
560 supplied Subscription Template Attributes are unsupported. These unsupported attributes are
561 also in the Subscription Attributes Group. See section 17.4 for more details about this status
562 code. See step #2 in this section for the cases that cause this status code.
- 563 9. The Printer MUST validate all Subscription Template Attributes and MUST return all unsupported
564 attributes and values in the corresponding Subscription Attributes Group of the response (see step #2)
565 unless it determines that it could not create additional Subscription Objects because of condition #6b) or
566 condition #6c). Then, the Printer NEED NOT validate these additional Subscription Template Attributes
567 and the client MUST NOT expect to find unsupported attributes from step #2 in such additional
568 Subscription Attribute Groups.

5.3 Subscription Template Attributes

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

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

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

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

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-time-interval (integer(0:MAX))	

580

5.3.1 notify-recipient-uri (uri)

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

A Printer MUST support this attribute.

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

588 The “notify-schemes-supported (1setOf uriScheme)” attribute MUST specify the schemes supported for this
589 attribute. Note: According to [RFC1738] the “:” terminates the scheme and so is not part of the scheme.
590 Therefore, values of this attribute do not include the “:”.

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

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

595 **5.3.2 notify-events (1setOf type2 keyword)**

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

599 A Printer MUST support this attribute.

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

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

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

610 **5.3.2.1 Standard Values for Subscribed Events**

611 Each value of this attribute is a keyword and it specifies a Subscribed Event that represents certain changes.
612 Some keywords represent a subset of changes of another keyword, e.g., ‘job-completed’ is an Event value
613 which is a sub-value of ‘job-state-change’. See section 5.3.2.2 for the case where this attribute contains both
614 a value and a sub-value.

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

616 A Printer MUST support the Events indicated as “REQUIRED” and MAY support the Events indicated as
617 “OPTIONAL”.

618 5.3.2.1.1 No Events

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

620 **‘none’**: REQUIRED - no Event Notifications for any Events. As the sole value of “notify-events-supported”,
621 this value means that the Printer does not support the sending of Event Notifications. As the sole value of
622 “notify-events-default”, this value means that a client MUST specify the “notify-events” attribute in order
623 for a Subscription Creation Operation to succeed. If the Printer receives this value as the sole value of a
624 Subscription Creation Operation, it does not create a Subscription Object. If a Printer receives this value
625 with other values of a Subscription Creation Operation, the Printer MUST treat this value as an
626 unsupported value.

627 5.3.2.1.2 Subscribed Printer Events

628 The standard keyword values for Subscribed Printer Events are:

629 **‘printer-state-changed’**: REQUIRED - the Printer changed state from any state to any other state.
630 Specifically, the value of the Printer’s “printer-state”, “printer-state-reasons” or “printer-is-accepting-
631 jobs” attributes changed.
632

633 This Subscribed Event value has the following sub-values: ‘printer-restarted’ and ‘printer-shutdown’. A
634 client can listen for any of these sub-values if it doesn’t want to listen to all printer-state changes:

635 **‘printer-restarted’**: OPTIONAL - when the printer is powered up .

636 **‘printer-shutdown’**: OPTIONAL - when the device is being powered down .

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

639 **‘printer-config-changed’**: OPTIONAL - when the configuration of a Printer has changed, i.e., the value of
640 the “printer-message-from-operator” or any “configuration” Printer attribute has changed. A
641 “configuration” Printer attribute is an attribute which can change value because of some human interaction
642 either direct or indirect, and which is not covered by one of the other Events in this section. Examples of
643 “configuration” Printer attributes are any of the Job Template attributes, such as “xxx-supported”, “xxx-
644 ready” and “xxx-default”. Often, such a change is the result of a client performing a Set-Printer-Attributes
645 operation (see [ipp-set]) on the Printer. The client has to perform a Get-Printer-Attributes to find out the
646 new values of these changed attributes. This Event is useful for GUI clients and drivers to update the
647 available printer capabilities to the user.
648

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

652 **‘printer-media-changed’**: OPTIONAL - when the media loaded on a printer has been changed,
653 i.e., the “media-ready” attribute has changed. This Event includes two cases: an input tray that
654 goes empty and an input tray that receives additional media of the same type or of a different
655 type. The client must check the “media-ready” Printer attribute (see [[ipp-modRFC2911](#)]
656 section 4.2.11) separately to find out what changed.

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

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

665 5.3.2.1.3 Subscribed Job Events

666 The standard keyword values for Subscribed Job Events are:

667 **‘job-state-changed’**: REQUIRED - the job has changed from any state to any other state. Specifically, the
668 Printer sends this Event whenever the value of the “job-state” attribute or “job-state-reasons” attribute
669 changes. When a Job is removed from the Job History (see [[ipp-modRFC2911](#)] 4.3.7.1), no Event is
670 generated.

671
672 This Event value has the following sub-values: ‘job-created’, ‘job-completed’ and ‘job-stopped’. A
673 client can listen for any of these sub-values if it doesn’t want to listen to all ‘job-state changes’.

674 **‘job-created’**: REQUIRED - the Printer has accepted a Job Creation operation and the job’s
675 “time-at-creation” attribute value is set (see [[ipp-modRFC2911](#)] section 4.3.14.1). The
676 Printer puts the job in the ‘pending’, ‘pending-held’ or ‘processing’ states..

677 **‘job-completed’**: REQUIRED - the job has reached one of the completed states, i.e., the value of
678 the job’s “job-state” attribute has changed to: ‘completed’, ‘aborted’, or ‘canceled’. The
679 Job’s “time-at-completed” and “date-time-at-completed” (if supported) attributes are set (see
680 [[ipp-modRFC2911](#)] section 4.3.14).. The Printer also sends this Event when a Job is
681 removed with the Purge-Job operation. In this case, the Event Notification MUST report the
682 ‘job-state’ as ‘canceled’.

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

685 **‘job-config-changed’**: OPTIONAL - when the configuration of a job has changed, i.e., the value of the
686 “job-message-from-operator” or any of the “configuration” Job attributes have changed. A

687 “configuration” Job attribute is an attribute that can change value because of some human interaction
688 either direct or indirect. Examples of “configuration” Job attributes are any of the job template attributes
689 and the “job-name” attribute. Often, such a change is the result of the user or the Operator performing a
690 Set-Job-Attributes operation (see [ipp-set]) on the Job object. The client performs a Get-Job-Attributes
691 to find out the new values of the changed attributes. This Event is useful for GUI clients and drivers to
692 update the job information to the user.

693 **‘job-progress’**: OPTIONAL – when the Printer has completed Printing a sheet. See the separate [ipp-prog]
694 specification for additional attributes that a Printer MAY send in an Event Notification caused by this
695 Event. The “notify-time-interval” attribute affects this Event by causing the Printer NOT to send an Event
696 Notification every time a ‘job-progress’ Events occurs. See section 5.3.8 for full details.

697 **5.3.2.2 Rules for Matching of Subscribed Events**

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

701 **5.3.2.2.1 Rules for Matching of Printer Events**

702 Suppose that the Printer causes Printer Event E to occur. For each Per-Job or Per-Printer Subscription S in
703 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
704 Printer MUST generate an Event Notification.

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

711 **5.3.2.2.2 Rules for Matching of Job Events**

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

- 713 1. For each Per-Printer Subscription S in the Printer, if E equals a value of this attribute in S or E is a sub-
714 value of a value of this attribute in S, the Printer MUST generate an Event Notification.
- 715 2. For each Per-Job Subscription S associated with Job J, if E equals a value of this attribute in S or E is
716 a sub-value of a value of this attribute in S, the Printer MUST generate an Event Notification.
- 717 3. For each Per-Job Subscription S that is NOT associated Job J, if E equals a value of this attribute in S
718 or E is a sub-value of a value of this attribute in, the Printer MUST NOT generate an Event
719 Notification from S.

720 Consider the example: There are three Subscription Objects listening for the Job Event ‘job-completed’.
721 Subscription Object A is a Per-Printer Subscription Object. Subscription Object B is a Per-Job Subscription
722 Object for Job 1, and Subscription Object C is a Per-Job Subscription Object for Job 2. In addition, Per-
723 Printer Subscription Object D is listening for the Job Event ‘job-state-changed’. When Job 1 completes, the
724 Printer sends an Event Notification to the Notification Recipient of Subscription Object A (because it is Per-
725 Printer) and Subscription Object B because it is a Per-Job Subscription Object associated with the Job
726 generating the Event. The Printer also sends an Event Notification to the Notification Recipient of
727 Subscription Object D because ‘job-completed’ is a sub-value of ‘job-state-changed’ – the value that
728 Subscription Object D is listening for. The Printer does not send an Event Notification to the Notification
729 Recipients of Subscription Object C because it is a Per-Job Subscription Object associated with some Job
730 other than the Job generating the Event.

731 **5.3.2.2.3 Special Cases for Matching Rules**

732 This section contains rule for special cases.

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

739 If an Event matches two values of this “notify-events” attribute in a single Subscription object (e.g., a value and
740 its sub-value), a Printer **MAY** send one Event Notification for each matched value in the Subscription Object
741 or it **MAY** send only one Event Notification per Subscription Object. The rules in sections 5.3.2.2.1 and
742 5.3.2.2.2 are purposefully ambiguous about the number of Event Notification sent when Event E matches two
743 or more values in a Subscription Object.

744 Consider the example: There are two Per-Printer Subscription Objects when a Job completes. Subscription
745 Object A has the Subscribed Job Event ‘job-state-changed’. Subscription Object B has the Subscribed Job
746 Events ‘job-state-changed’ and ‘job-completed’. The Printer sends an Event Notification to the Notification
747 Recipient of Subscription Object A with the value of ‘job-state-changed’ for the “notify-subscribing-event”
748 attribute. The Printer sends either one or two Event Notifications to the Notification Recipient of Subscription
749 Object B, depending on implementation. If it sends two Event Notifications, one has the value of ‘job-state-
750 changed’ for the “notify-subscribing-event” attribute, and the other has the value of ‘job-completed’ for the
751 “notify-subscribing-event” attribute. If it sends one Event Notification, it has the value of either ‘job-state-
752 changed’ or ‘job-completed’ for the “notify-subscribing-event” attribute, depending on implementation. The
753 algorithm for choosing such a value is implementation dependent.

754 **5.3.3 notify-attributes (1setOf type2 keyword)**

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

758 A Printer MAY support this attribute.

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

762 Each keyword value of this attribute on a Subscription Object MUST be a value of the “notify-attributes-
763 supported (1setOf type2 keyword)” attribute. The “notify-attributes-supported” MAY contain any Printer
764 attribute, Job attribute or Subscription Object attribute that the Printer supports in an Event Notification. It
765 MUST NOT contain any of the attributes in Section 9.1 that a Printer automatically puts in an Event
766 Notification; it would be redundant. If a client supplies an attribute in Section 9.1, the Printer MUST treat it as
767 an unsupported attribute value of the “notify-attributes” attribute.

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

- 769 a) a Subscription attribute, the Printer MUST use the attribute N in the Subscription Object that is being
770 used to generate the Event Notification.
- 771 b) a Job attribute and the Printer is generating an Event Notification from a Per-Job Subscription Object S,
772 the Printer MUST use the attribute N in the Job object associated with S.
- 773 c) a Job attribute and the Printer is generating an Event Notification from a Per-Printer Subscription Object
774 and the Event is:
- 775 • a Job Event, the Printer MUST use the attribute N in the Job object that caused the Event.
 - 776 • a Printer Event, the Printer MUST use the attribute N in the active Job.

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

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

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

782 **5.3.4 notify-user-data (octetString(63))**

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

- 785 • the identity of the Subscriber
- 786 • a path or index to some Subscriber information
- 787 • a key that identifies to the Notification Recipient the ultimate recipient of the Event Notification
- 788 • the id for a Notification Recipient that had previously registered with an Instant Messaging Service

789 A Printer MUST support this attribute.

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

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

795 **5.3.5 notify-charset (charset)**

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

798 A Printer MUST support this attribute.

799 A client MAY supply this attribute in a Subscription Creation Operation. If the client does not supply this
800 attribute in Subscription Creation Operation or supplies an unsupported value, the Printer MUST populate this
801 attribute in the Subscription Object with the value of the “attributes-charset” operation attribute, which is a
802 REQUIRED attribute in all IPP requests (see [[ipp-modRFC2911](#)]). If the value of the “attributes-charset”
803 attribute is unsupported, the Printer MUST populate this attribute in the Subscription Object with the value of
804 the Printer’s “charset-configured” attribute. There is no “notify-charset-default” attribute.

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

807 **5.3.6 notify-natural-language (naturalLanguage)**

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

811 A Printer MUST support this attribute.

812 A client MAY supply this attribute in a Subscription Creation Operation. If the client does not supply this
813 attribute in Subscription Creation Operation or supplies an unsupported value, the Printer MUST populate this
814 attribute in the Subscription Object with the value of the “attributes-natural-language” operation attribute,
815 which is a REQUIRED attribute in all IPP requests (see [[ipp-modRFC2911](#)]). If the value of the “attributes-
816 natural-language” attribute is unsupported, the Printer MUST populate this attribute in the Subscription Object
817 with the value of the Printer’s “natural-language-configured” attribute. There is no “notify-natural-language-
818 default” attribute.

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

821 **5.3.7 notify-lease-duration (integer(0:67108863))**

822 This attribute specifies the duration of the lease (in seconds) associated with the Per-Printer Subscription
823 Object at the time the Subscription Object was created or the lease was renewed. The duration of the lease is
824 infinite if the value is 0, i.e., the lease never expires.

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

828 A Printer MUST support this attribute.

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

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

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

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

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

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

848 **5.3.8 notify-time-interval (integer(0:MAX))**

849 The ‘job-progress’ Event occurs each time that a Printer completes a sheet. Some Notification Recipients do
850 not want to receive an Event Notification every time this Event occurs. This attribute allows a Subscribing
851 Client to request how often it wants to receive Event Notifications for ‘job-progress’ Events. The value of this
852 attribute MAY be any nonnegative integer (0,MAX) indicating the minimum number of seconds between ‘job-
853 progress’ Event Notifications.

854 The Printer MUST support this attribute if and only if the Printer supports the ‘job-progress’ Event.

855 A client MAY supply this attribute in a Subscription Creation Operation. If the client does not supply this
856 attribute, the Printer MUST not populate this attribute on the Subscription Object. There is no “notify-time-
857 interval-default” attribute.

858 There is no “notify-time-interval-supported” attribute.

859 If the ‘job-progress’ Event occurs and a Subscription Object contains the ‘job-progress’ Event as a value of
860 the ‘notify-events’ attribute, there are two cases to consider:

- 861 1. This attribute is not present on the Subscription Object or has the value of 0. The Printer MUST generate
862 and send an Event Notification (as is the case with other Events).
- 863 2. This attribute is present with a nonzero value of N:
 - 864 a) If the Printer has not sent an Event Notification for the ‘job-progress’ Event for the associated
865 Subscription Object within the past N seconds, the Printer MUST send an Event Notification for the
866 Event that just occurred. Note when the Printer completes the first page of a Job, this rule implies that
867 the Printer sends an Event Notification for a Per-Job Subscription Objects.
 - 868 b) Otherwise, the Printer MUST NOT generate or send an Event Notification for the associated
869 Subscription Object. The Printer MUST NOT increase the value of the “notify-sequence-number”
870 Subscription Object attribute (i.e., the sequence of values of the “notify-sequence-number” attribute
871 counts the Event Notifications that the Printer sent and not the Events that do not cause an Event
872 Notification to be sent).

873 It is RECOMMENDED that a Subscribing Client use this attribute when it subscribes to the ‘job-progress’
874 Event, and that the value be sufficiently large to limit the frequency with which the Printer sends Event
875 Notifications requests.

876 This attribute MUST NOT effect any Events other than ‘job-progress’.

877 5.4 Subscription Description Attributes

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

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

881 A client MUST NOT supply the attributes in Table 2 in a Subscription Template Attributes Group of a
882 Subscription Creation Operation. If the client supplies them, the Printer MUST NOT set them and MUST
883 treat them as unsupported attributes. There are no corresponding default or supported attributes.

884 **Table 2 – Subscription Description Attributes**

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

885

886 5.4.1 notify-subscription-id (integer (1:MAX))

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

889 A Printer MUST support this attribute.

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

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

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

898 5.4.2 notify-sequence-number (integer (0:MAX))

899 The value of this attribute indicates the number of times that the Printer has generated and attempted to send
900 an Event Notification. When an Event Notification contains this attribute, the Notification Recipient can

901 determine whether it missed some Event Notifications (i.e., numbers skipped) or received duplicates (i.e.,
902 same number twice).

903 A Printer MUST support this attribute.

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

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

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

915 **5.4.3 notify-lease-expiration-time (integer(0:MAX))**

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

918 A Printer MUST support this attribute.

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

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

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

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

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

934 **5.4.4 notify-printer-up-time (integer(1:MAX))**

935 This attribute is an alias for the Printer's "printer-up-time" attribute " (see [~~ipp-mod~~[RFC2911](#)] section
936 4.4.29).

937 A Printer MUST support this attribute.

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

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

945 **5.4.5 notify-printer-uri (uri)**

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

947 A Printer MUST support this attribute.

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

951 **5.4.6 notify-job-id (integer(1:MAX))**

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

954 A Printer MUST support this attribute.

955 If this attribute is not present, the Subscription Object MUST be a Per-Printer Subscription. If this attribute is
956 present, the Subscription Object MUST be a Per-Job Subscription Object and this attribute MUST identify
957 the Job with which the Subscription Object is associated.

958 Note: This attribute could be useful to a Notification Recipient that receives an Event Notification generated
959 from a Per-Job Subscription Object and caused by a Printer Event. The Event Notification gives access to the
960 Printer and the Subscription Object. The Event Notification gives access to the associated Job only via this
961 attribute.

962 **5.4.7 notify-subscriber-user-name (name(MAX))**

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

964 A Printer MUST support this attribute.

965 The Printer sets this attribute to the most authenticated printable name that it can obtain from the authentication
 966 service over which the Subscription Creation Operation was received. The Printer uses the same mechanism
 967 for determining the value of this attribute as it does for a Job's "job-originating-user-name" (see [~~ipp-~~
 968 ~~mod~~[RFC2911](#)] section 4.3.6).

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

972 6 Printer Description Attributes Related to Notification

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

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

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

977

978 6.1 printer-state-change-time (integer(1:MAX))

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

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

983 On power-up, the Printer MUST set the value of this attribute to be the value of its "printer-up-time" attribute,
 984 so that it always has a value. Whenever the 'printer-state-changed' Printer Event occurs, the Printer MUST
 985 set this attribute to the value of the Printer's "printer-up-time" attribute.

986 6.2 printer-state-change-date-time (dateTime)

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

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

991 On power-up, the Printer MUST set the value of this attribute to be the value of its “printer-current-time”
 992 attribute, so that it always has a value (see [~~ipp-mod~~RFC2911] section 4.4.30 on “printer-current-time”).
 993 Whenever the ‘printer-state-changed’ Printer Event occurs, the Printer MUST set this attribute to the value of
 994 the Printer’s “printer-current-time” attribute.

995 7 New Values for Existing Printer Description Attributes

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

997 7.1 operations-supported (1setOf type2 enum)

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

1000 **Table 4 – Operation-id assignments**

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

1001 8 Attributes Only in Event Notifications

1002 This section contains those attributes that exist only in Event Notifications and do not exist in any objects.

1003 8.1 notify-subscribed-event (type2 keyword)

1004 This attribute indicates the Subscribed Event that caused the Printer to send this Event Notification. This
 1005 attribute exists only in Event Notifications.

1006 This attribute MUST contain one of the values of the “notify-events” attribute in the Subscription Object, i.e.,
 1007 one of the Subscribed Event values. Its value is the Subscribed Event that “matches” the Event that caused the
 1008 Printer to send this Event Notification. This Subscribed Event value may be identical to the Event or the Event
 1009 may be a sub-value of the Subscribed Event. For example, the ‘job-completed’ Event (which is a sub-event
 1010 of the ‘job-state-changed’ event) would cause the Printer to send an Event Notification for either the ‘job-
 1011 completed’ or ‘job-state-changed’ Subscribed Events and to send the ‘job-completed’ or ‘job-state-
 1012 changed’ value for this attribute, respectively,. See section 5.3.2.2 for the “matching” rules of Subscribed
 1013 Events and for additional examples.

1014 The Delivery Method Document specifies whether the Printer includes the value of this attribute in an Event
 1015 Notification.

1016 8.2 notify-text (text(MAX))

1017 This attribute contains a Human Consumable text message (see section 0). This message describes the Event
1018 and is encoded as plain text, i.e., 'text/plain' with the charset specified by Subscription Object's "notify-
1019 charset" attribute.

1020 The Delivery Method Document specifies whether the Printer includes this attribute in an Event Notification.

1021 9 Event Notification Content

1022 This section defines the Event Notification content that the Printer sends when an Event occurs.

1023 When an Event occurs, the Printer MUST find each Subscription object whose "notify-events" attribute
1024 "matches" the Event. See section 5.3.2.2 for details on "matching". For each matched Subscription Object,
1025 the Printer MUST create an Event Notification with the content and format that the Delivery Method
1026 Document specifies. The content contains the value of attributes specified by the Delivery Method Document.
1027 The Printer obtains the values immediately after the Event occurs. For example, if the "printer-state" attribute
1028 changes from 'idle' to 'processing', the Event 'printer-state-changed' occurs and the Printer puts various
1029 attributes into the Event Notification, including "printer-up-time" and "printer-state" with the values that they
1030 have immediately after the Event occurs, i.e., the value of "printer-state" is 'processing'.

1031 If two different Events occur simultaneously, or nearly so (e.g., "printer-up-time" has the same value for both),
1032 the Printer MUST create a separate Event Notification for each Event, even if the associated Subscription
1033 Object is the same for both Events. However, the Printer MAY combine these distinct Event Notifications into
1034 a single Compound Event Notification if the Delivery Method supports Compound Event Notifications For
1035 example, suppose that two nearly-simultaneously Events represent two successive 'printer-state-changed'
1036 Events, one from 'idle' to 'processing' and another from 'processing' to 'stopped'. These two Events have
1037 the same name but are different instances of the Event. Then the Printer MUST create a separate Event
1038 Notification for each Event and SHOULD accurately report the "printer-state" of the first Event as
1039 'processing' and the second Event as 'stopped'.

1040 If a Subscription Object contains more than one Subscribed Event, and several Events occur in quick
1041 succession each matching a different Subscribed Event in the Subscription Object, the Printer MUST NOT
1042 generate a single Event Notification from several of these Events, but MAY combine distinct Event
1043 Notifications into a single Compound Event Notification if the Delivery Method supports Compound Event
1044 Notifications.

1045 After the Printer has created the Event Notification, the Printer delivers it via either a:

1046 Push Delivery Method: The Printer sends the Event Notification shortly after an Event occurs. For
1047 some Push Delivery Methods, the Notification Recipient MUST send a response; for others it MUST
1048 NOT send a response.

1049 Pull Delivery Method: The Printer saves Event Notifications for some event-lease time and expects the
1050 Notification Recipient to request Event Notifications. The Printer returns the Event Notifications in a
1051 response to such a request.

1052 If an error that meets the following conditions occurs, the Printer **MUST** cancel the Subscription Object.

- 1053 a) the error occurs during the sending of an Event Notification generated from Subscription Object S **AND**
1054 b) the error would continue to occur every time the Printer sends an Event Notification generated from
1055 Subscription Object S in the future.

1056 From example, if the address of the “notify-recipient-uri” of Subscription Object A references a non-existent
1057 target and the Printer determines that this fact, it **MUST** delete Subscription Object A.

1058 The next two sections describe the values that a Printer sends in the content of Machine Consumable and
1059 Human Consumable Event Notifications, respectively.

1060 The tables in the sub-sections of this section contain the following columns:

1061 a) **Source Value:** the name of the attribute that supplies the value for the Event Notification. Asterisks in
1062 this field refer to a note below the table.

1063 b) **Sends:** if the Printer supports the value (column 1) on the Source Object (column 3) the Delivery
1064 Method **MUST** specify:

1065 **MUST:** that the Printer **MUST** send the value.

1066 **SHOULD:** either that the Printer **MUST** send the value or that the value is incompatible with the
1067 Delivery Method.

1068 **MAY:** that the Printer **MUST**, **SHOULD**, **MAY**, **MUST NOT**, **SHOULD NOT**, or **NEED NOT**
1069 send the value. The Delivery Method specifies the level of conformance for the Printer.

1070 c) **Source Object:** the object from which the source value comes. If the object is “Event Notification”,
1071 the Printer fabricates the value when it sends the Event Notification. See section 8.

1072 9.1 Content of Machine Consumable Event Notifications

1073 This section defines the attributes that a Delivery Method **MUST** mention in a Delivery Method Document
1074 when specifying the Machine Consumable Event Notification’s contents.

1075 This document does not define the order of attributes in Event Notifications. However, Delivery Method
1076 Documents **MAY** define the order of some or all of the attributes.

1077 A Delivery Method Document **MUST** specify additional attributes (if any) that a Printer implementation sends
1078 in a Machine Consumable Event Notification.

1079 Notification Recipients **MUST** be able to accept Event Notifications containing attributes they do not
 1080 recognize. What a Notification Recipient does with an unrecognized attribute is implementation-dependent.
 1081 Notification Recipients **MAY** attempt to display unrecognized attributes anyway or **MAY** ignore them.

1082 The next three sections define the attributes in Event Notification Contents that are:

- 1083 1. for all Events
- 1084 2. for Job Events only
- 1085 3. for Printer Events only

1086 9.1.1 Event Notification Content Common to All Events

1087 This section lists the attributes that a Delivery Method Document **MUST** specify for all Events.

1088 Table 5 lists potential values in each Event Notification.

1089 **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

1090
 1091 *A Printer **MUST** send this value only if and only if it supports the Printer’s “printer-current-time” attribute.

1092 ** If the Subscription Object does not contain a “notify-user-data” attribute and the Delivery Method
 1093 document **REQUIRES** the Printer to send the “notify-user-data” source value in the Event Notification, the
 1094 Printer **MUST** send an octet-string of length 0.

1095 *** The last three rows represent additional attributes that a client **MAY** request via the “notify-attributes”
 1096 attribute. A Printer **MAY** support the “notify-attributes” attribute. The Delivery Method **MUST** say that the
 1097 Printer **MUST**, **SHOULD**, **MAY**, **MUST NOT**, **SHOULD NOT**, or **NEED NOT** support the “notify-
 1098 attributes” attribute and specific values of this attribute. The Delivery Method **MAY** say that support for the

1099 “notify-attributes” is conditioned on support of the attribute by the Printer or it MAY say that Printer MUST
 1100 support the “notify-attributes” attribute if the Printer supports the Delivery Method.

1101 9.1.2 Additional Event Notification Content for Job Events

1102 This section lists the additional attributes that a Delivery Method Document MUST specify for Job Events.
 1103 See Table 6.

1104 **Table 6 – Additional Event Notification Content for Job Events**

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

1105
 1106 * The Printer MUST send the “job-impressions-completed” attribute in an Event Notification only for the
 1107 combinations of Events and Subscribed Events shown in Table 7.

1108 **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’

1109

1110 9.1.3 Additional Event Notification Content for Printer Events

1111 This section lists the additional attributes that a Delivery Method Document MUST specify for Printer Events.
 1112 See Table 8.

1113 **Table 8 – Additional Event Notification Content for Printer Events**

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

1114

1115 9.2 Content of Human Consumable Event Notification

1116 This section defines the information that a Delivery Method MUST mention in a Delivery Method Document
 1117 when specifying the Human Consumable Event Notifications contents or the value of the “notify-text” attribute.

1118 Such a Delivery Method MUST specify the following information and a Printer SHOULD send it:

- 1119 a) the Printer name (see Table 9)
- 1120 b) the time of the Event (see Table 11)
- 1121 c) for Printer Events only:
 - 1122 i) the Event (see Table 10) and/or Printer state information (see Table 14)
- 1123 d) for Job Events only:
 - 1124 i) the job identity (see Table 12)
 - 1125 ii) the Event (see Table 10) and/or Job state information (see Table 13)

1126

1127 The subsections of this section specify the attributes that a Printer MUST use to obtain this information.

1128 A Delivery Method Document MUST specify additional information (if any) that a Printer implementation
1129 sends in a Human Consumable Event Notification or in the “notify-text” attribute.

1130 A client MUST NOT request additional attributes via the “notify-attributes” attribute because this attribute
1131 works only for Machine Consumable Event Notifications.

1132 Notification Recipients MUST NOT expect to be able to parse the Human Consumable Event Notification
1133 contents or the value of the “notify-text” attribute.

1134 The next three sections define the attributes in Event Notification Contents that are:

- 1135 a) for all Events
- 1136 b) for Job Events only
- 1137 c) for Printer Events only

1138

1139 **9.2.1 Event Notification Content Common to All Events**

1140 This section lists the source of the information that a Delivery Method MUST specify for all Events.

1141 There is a separate table for each piece of information. Each row in the table represents a source value for the
1142 information and the values are listed in order of preference, with the first one being the preferred one. An
1143 implementation SHOULD use the source value from the earliest row in each table. It MAY use the source
1144 value from another row instead, or it MAY combine the source values from several rows. An implementation
1145 is free to determine the best way to present this information.

1146 In all tables of this section, all rows contain a “MAY” in order to state that the Delivery Method specifies the
1147 conformance.

1148 Table 9 lists the source of the information for the Printer Name. The “printer-name” is more user-friendly
1149 unless the Notification Recipient is in a place where the Printer name is not meaningful. For example, an
1150 implementation could have the intelligence to send the value of the “printer-name” attribute to a Notification
1151 Recipient that can access the Printer via value of the “printer-name” attribute and otherwise send the value of
1152 the “notify-printer-uri” attribute.

1153

Table 9 – Printer Name in Event Notification Content

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

1154

1155

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

1156

1157

Table 10 – Event Name in Event Notification Content

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

1158

1159

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

1160

1161

1162

1163

Table 11 – Event Time in Event Notification Content

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

1164

1165 9.2.2 Additional Event Notification Content for Job Events

1166

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

1167

1168

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

1169

1170

Table 12 – Job Name in Event Notification Content

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

1171

1172

Table 13 lists the source of the information for the job state. If a Printer supports the "job-state-message" and "job-detailed-state-message" attributes, it SHOULD use those attributes for the job state information,

1173

1174 otherwise, it should fabricate such information from the “job-state” and “job-state-reasons”. For some Events,
1175 a Printer MAY combine this information with Event information.

1176 **Table 13 – Job State in Event Notification Content**

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

1177
1178 **9.2.3 Additional Event Notification Content for Printer Events**

1179 This section lists the source of the additional information that a Delivery Method MUST specify for Printer
1180 Events.

1181 Table 14 lists the source of the information for the printer state. If a Printer supports the “printer-state-
1182 message”, it SHOULD use that attribute for the job state information, otherwise it SHOULD fabricate such
1183 information from the “printer-state” and “printer-state-reasons”. For some Events, a Printer MAY combine
1184 this information with Event information.

1185 **Table 14 – Printer State in Event Notification Content**

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

1186 **10 Delivery Methods**

1187 A Delivery Method is the mechanism, i.e., protocol, by which the Printer delivers an Event Notification to a
1188 Notification Recipient. There are several potential Delivery Methods for Event Notifications, standardized, as
1189 well as proprietary. This document does not define any of these delivery mechanisms. Each Delivery Method
1190 MUST be defined in a Delivery Method Document that is separate from this document. New Delivery
1191 Methods will be created as needed using an extension to the registration procedures defined in [~~ipp-~~
1192 [modRFC2911](#)]. Such documents are registered with IANA (see section 13).

1193 The following sorts of Delivery Methods are expected:

- 1194 – The Notification Recipient polls for Event Notifications at intervals directed by the Printer
- 1195 – The Printer sends Event Notifications to the Notification Recipient using http as the transport.

11. What are the additional values or pieces of information that a Printer sends in an Event Notification content and the conformance requirements thereof?	
12. What are the additional Subscription Template and/or Subscription Description attributes and the conformance requirements thereof?	
13. What are the additional Printer Description attributes and the conformance requirements thereof?	

1204

1205 **11 Operations for Notification**

1206 This section defines all of the operations for Notification. Section 7.1 assigns the “operation-id” for each
 1207 operation. The following two sub-sections define Subscription Creation Operations, and other operations.

1208 **11.1 Subscription Creation Operations**

1209 This section defines the Subscription Creation Operations. The first section on Create-Job-Subscriptions gives
 1210 most of the information. The other Subscription Creation Operations refer to the section on Create-Job-
 1211 Subscriptions, even though the Create-Job-Subscriptions operation is the only OPTIONAL operation in this
 1212 document (see section 12).

1213 A Printer MUST support Create-Printer-Subscriptions and the Subscription Template Attributes Group in
 1214 Job Creation operations. It MAY support Create-Job-Subscriptions operations.

1215 **11.1.1 Create-Job-Subscriptions Operation**

1216 The operation creates one or more Per-Job Subscription Objects. The client supplies one or more
 1217 Subscription Template Attributes Groups each containing one or more of Subscription Template Attributes
 1218 (defined in section 5.3).

1219 Except for errors, the Printer MUST create exactly one Per-Job Subscription Object from each Subscription
 1220 Template Attributes Group in the request, even if the newly created Subscription Object would have identical
 1221 behavior to some existing Subscription Object. The Printer MUST associate each newly created Per-Job
 1222 Subscription Object with the target Job, which is specified by the “notify-job-id” operation attribute.

1223 The Printer MUST accept the request in any of the target job’s ‘not-completed’ states, i.e., ‘pending’,
 1224 ‘pending-held’, ‘processing’, or ‘processing-stopped’. The Printer MUST NOT change the job’s “job-state”
 1225 attribute because of this operation. If the target job is in any of the ‘completed’ states, i.e., ‘completed’,
 1226 ‘canceled’, or ‘aborted’, then the Printer MUST reject the request and return the ‘client-error-not-possible’
 1227 status code; the response MUST NOT contain any Subscription Attribute Groups.

1228 Access Rights: To create Per-Job Subscription Objects, the authenticated user (see [\[IPP-MODRFC2911\]](#))
 1229 section 8.3) performing this operation MUST either be the job owner or have Operator or Administrator

1230 access rights for this Printer (see [~~IPP-MOD~~[RFC2911](#)] sections 1 and 8.5). Otherwise the Printer MUST
1231 reject the operation and return: the ‘client-error-forbidden’, ‘client-error-not-authenticated’, or ‘client-error-
1232 not-authorized’ status code as appropriate.

1233 **11.1.1.1 Create-Job-Subscriptions Request**

1234 The following groups of attributes are part of the Create-Job-Subscriptions Request:

1235 Group 1: Operation Attributes

1236 Natural Language and Character Set:

1237 The “attributes-charset” and “attributes-natural-language” attributes as described in [~~ipp-~~
1238 [modRFC2911](#)] section 3.1.4.1.

1239

1240 Target:

1241 The “printer-uri” attribute which defines the target for this operation as described in [~~ipp-~~
1242 [modRFC2911](#)] section 3.1.5.

1243

1244 Requesting User Name:

1245 The “requesting-user-name” attribute SHOULD be supplied by the client as described in [~~ipp-~~
1246 [modRFC2911](#)] section 8.3.

1247

1248 notify-job-id (integer(1:MAX)):

1249 The client MUST supply this attribute and it MUST specify the Job object to associate the Per-Job
1250 Subscription with. The value of “notify-job-id” MUST be the value of the “job-id” of the associated
1251 Job object. If the client does not supply this attribute, the Printer MUST reject this request with a
1252 ‘client-error-bad-request’ status code.

1253

1254 Group 2-N: Subscription Template Attributes

1255 For each occurrence of this group:

1256

1257 The client MUST supply one or more Subscription Template Attributes in any order. See section 5.3
1258 for a description of each such attribute. See section 5.2 for details on processing these attributes.

1259 **11.1.1.2 Create-Job-Subscriptions Response**

1260 The Printer MUST return to the client the following sets of attributes as part of a Create-Job-Subscriptions
1261 response:

1262 Group 1: Operation Attributes

1263 Status Message:

1264 In addition to the REQUIRED status code returned in every response, the response OPTIONALLY
1265 includes a "status-message" (text(255)) and/or a "detailed-status-message" (text(MAX)) operation
1266 attribute as described in [RFC2911] sections 13 and 31.6. As defined in [ipp-mod].

1267

1268 In this group, the Printer can return any status codes defined in [ipp-modRFC2911] and section 16.
1269 The following is a description of the important status codes:

1270

1271 **successful-ok:** the Printer created all Subscription Objects requested.

1272 **successful-ok-ignored-subscriptions:** the Printer created some Subscription Objects requested but
1273 some failed. The Subscription Attributes Groups with a "notify-status-code" attribute are the ones
1274 that failed.

1275 **client-error-ignored-all-subscriptions:** the Printer created no Subscription Objects requested and all
1276 failed. The Subscription Attributes Groups with a "notify-status-code" attribute are the ones that
1277 failed

1278 **client-error-not-possible:** For this operation and other Per-Job Subscription operations, this error
1279 can occur because the specified Job has already completed.

1280

1281 Natural Language and Character Set:

1282 The "attributes-charset" and "attributes-natural-language" attributes as described in [ipp-
1283 modRFC2911] section 3.1.4.2.

1284

1285 Group 2: Unsupported Attributes

1286 See [ipp-modRFC2911] section 3.1.7 for details on returning Unsupported Attributes. This group
1287 does not contain any unsupported Subscription Template Attributes; they are returned in the
1288 Subscription Attributes Group (see below).

1289

1290 Group 3-N: Subscription Attributes

1291 These groups MUST be returned if and only if unless the Printer is unable to interpret the entire
1292 request, e.g., the "status-code" parameter returned in Group 1 has the values: 'client-error-bad-
1293 request', 'successful-ok', 'successful-ok-ignored-subscriptions', or 'client-error-ignored-all-
1294 subscriptions'.

1295

1296 "notify-status-code" (type2 enum):

1297 Indicates the status of this subscription (see section 17 for the status code definitions). Section 5.2
1298 defines when this attribute MUST be present in this group.

1299

1300 See section 5.2 for details on the contents of each occurrence of this group.

1301

1302 **11.1.2 Create-Printer-Subscriptions operation**

1303 The operation is identical to Create-Job-Subscriptions with exceptions noted in this section.

1304 The operation creates Per-Printer Subscription Objects instead of Per-Job Subscription Objects, and
1305 associates each newly created Per-Printer Subscription Object with the Printer specified by the operation
1306 target rather than with a specific Job.

1307 The Printer MUST accept the request in any of its states, i.e., 'idle', 'processing', or 'stopped'. The Printer
1308 MUST NOT change its "printer-state" attribute because of this operation.

1309 Access Rights: To create Per-Printer Subscription Objects, the authenticated user (see [~~IPP-~~
1310 ~~MODRFC2911~~] section 8.3) performing this operation MUST have Operator or Administrator access rights
1311 for this Printer (see [~~IPP-~~~~MODRFC2911~~] sections 1 and 8.5). Otherwise, the Printer MUST reject the
1312 operation and return: the 'client-error-forbidden', 'client-error-not-authenticated', or 'client-error-not-
1313 authorized' status code as appropriate.

1314 **11.1.2.1 Create-Printer-Subscriptions Request**

1315 The groups are identical to the Create-Job-Subscriptions (see section 11.1.1.1) except that the Operation
1316 Attributes group MUST NOT contain the "notify-job-id" attribute. If the client does supply the "notify-job-
1317 id" attribute, then the Printer MUST treat it as any other unsupported Operation attribute and MUST return it
1318 in the Unsupported Attributes group.

1319 **11.1.2.2 Create-Printer-Subscriptions Response**

1320 The groups are identical to the Create-Job-Subscriptions (see section 11.1.1.2).

1321

1322 **11.1.3 Job Creation Operation – Extensions for Notification**

1323 This document extends the Job Creation operations to create Subscription Objects as a part of the operation.

1324 The operation is identical to Create-Job-Subscriptions with exceptions noted in this section.

1325 Unlike the Create-Job-Subscriptions operation, this operation associates the newly created Subscription
1326 Objects with the Job object created by this operation. The operation succeeds if and only if the Job creation
1327 succeeds. If the Printer does not create some or all of the requested Subscription Objects, the Printer MUST
1328 return a 'successful-ok-ignored-subscriptions' status-code instead of a 'successful-ok' status-code, but the
1329 Printer MUST NOT reject the operation because of a failure to create Subscription Objects.

1330 If the operation includes a Job Template group, the client MUST supply it after the Operation Attributes group
1331 and before the first Subscription Template Attributes Group.

1332 If a Printer does not support this Notification specification, then it MUST treat the Subscription Attributes
1333 Group like an unknown group and ignore it (see [~~ipp-mod~~RFC2911] section 5.2.2). Because the Printer
1334 ignores the Subscription Attributes Group, it doesn't return them in the response either, thus indicating to the
1335 client that the Printer doesn't support Notification.

1336 Access Rights: To create Per-Job Subscription Objects, the authenticated user (see [~~IPP-MOD~~RFC2911]
1337 section 8.3) performing this operation MUST either have permission to create Jobs on the Printer. Otherwise
1338 the Printer MUST reject the operation and return: the 'client-error-forbidden', 'client-error-not-
1339 authenticated', or 'client-error-not-authorized' status code as appropriate.

1340 11.1.3.1 Job Creation Request

1341 The groups for this operation are sufficiently different from the Create-Job-Subscriptions operation that they
1342 are all presented here. The following groups of attributes are supplied as part of a Job Creation Request:

1343 Group 1: Operation Attributes

1344 Same as defined in [~~ipp-mod~~RFC2911] for Print-Job, Print-URI, and Create-Job requests.

1345 Group 2: Job Template Attributes

1346 The client OPTIONALLY supplies a set of Job Template attributes as defined in [~~ipp-~~
1347 ~~mod~~RFC2911] section 4.2.

1348 Group 3 to N: Subscription Template Attributes

1349 The same as Group 2-N in Create-Job-Subscriptions. See section 11.1.1.1.

1350 Group N+1: Document Content (Print-Job only)

1351 The client MUST supply the document data to be processed.

1352

1353 11.1.3.2 Job Creation Response

1354 The Printer MUST return to the client the following sets of attributes as part of a Print-Job, Print-URI, and
1355 Create-Job Response:

1356 Group 1: Operation Attributes

1357

1358 Status Message:

1359

1360 As defined in [~~ipp-mod~~RFC2911] for Print-Job, Print-URI, and Create-Job requests.

1361

1362 In this group, the Printer can return any status codes defined in [~~ipp-mod~~RFC2911] and section 16.

1363 The following is a description of the important status codes:

1364

1365 **successful-ok:** the Printer created the Job and all Subscription Objects requested.

1366 **successful-ok-ignored-subscriptions:** the Printer created the Job and not all of the Subscription
1367 Objects requested. This status-code hides ‘successful-ok-xxx’ status-codes that could reveal
1368 problems in Job creation. The Printer **MUST** not return the ‘client-error-ignored-all-subscriptions’
1369 status code for Job Creation operations because the Printer returns an error status-code only when
1370 it fails to create a Job.

1371

1372 Natural Language and Character Set:

1373 The “attributes-charset” and “attributes-natural-language” attributes as described in [~~ipp-~~
1374 ~~mod~~[RFC2911](#)] section 3.1.4.2.

1375

1376 Group 2: Unsupported Attributes

1377 See [~~ipp-mod~~[RFC2911](#)] section 3.1.7 for details on returning Unsupported Attributes. This group
1378 does not contain any unsupported Subscription Template Attributes; they are returned in the
1379 Subscription Attributes Group (see below).

1380

1381 Group 3: Job Object Attributes

1382 As defined in [~~ipp-mod~~[RFC2911](#)] for Print-Job, Print-URI, and Create-Job requests.

1383

1384 Group 4 to N: Subscription Attributes

1385 These groups **MUST** be returned if and only if the client supplied Subscription Template Attributes
1386 and the operation was accepted.

1387

1388 See section 5.2 for details on the contents of each occurrence of this group.

1389

1390 **11.2 Other Operations**

1391 This section defines other operations on Subscription objects.

1392 **11.2.1 Validate-Job Operation - Extensions for Notification**

1393 A client can test whether one or more Subscription Objects could be created using the Validate-Job
1394 operation. The client supplies one or more Subscription Template Attributes Groups (defined in section 5.3),
1395 just as in a Job Creation request.

1396 A Printer **MUST** support this extension to this operation.

1397 The Printer **MUST** accept requests that are identical to the Job Creation request defined in section 11.1.3.1,
1398 except that the request **MUST** not contain document data.

1399 The Printer **MUST** return the same groups and attributes as the Print-Job operation (section 11.1.3.1) with
1400 the following exceptions. The Printer **MUST NOT** return a Job Object Attributes Group because no Job is

1401 created. The Printer MUST NOT return the “notify-subscription-id” attribute in any Subscription Attribute
1402 Group because no Subscription Object is created.

1403 If the Printer would succeed in creating a Subscription Object, the corresponding Subscription Attributes
1404 Group either has no ‘status-code’ attribute or a ‘status-code’ attribute with a value of ‘successful-ok-too-
1405 many-events’ or ‘successful-ok-ignored-or-substituted-attributes’ (see sections 5.2 and 17). The status-codes
1406 have the same meaning as in Job Creation except the results state what “would happen”.

1407 The Printer MUST validate Subscription Template Attributes Groups in the same manner as the Job Creation
1408 operations.

1409 11.2.2 Get-Printer-Attributes - Extensions for Notification

1410 This operation is extended so that it returns Printer attributes defined in this document.

1411 A Printer MUST support this extension to this operation.

1412 In addition to the requirements of [~~ipp-mod~~[RFC2911](#)] section 3.2.5, a Printer MUST support the following
1413 additional values for the “requested-attributes” Operation attribute in this operation and return such attributes
1414 in the Printer Object Attributes group of its response.

- 1415 1. **Subscription Template Attributes:** Each supported attribute in column 2 of Table 1.
- 1416 2. **New Printer Description Attributes:** Each supported attribute in section 6.
- 1417 3. **New Group Name:** The ‘subscription-template’ group name, which names all supported Subscription
1418 Template Attribute in column 2 of Table 1. This group name is also used in the Get-Subscription-
1419 Attributes and Get-Subscriptions operation with an analogous meaning.
- 1420 4. **Extended Group Name:** The ‘all’ group name, which names all Printer attributes according to [~~ipp-~~
1421 ~~mod~~[RFC2911](#)] section 3.2.5. In this extension ‘all’ names all attributes specified in [~~ipp-~~
1422 ~~mod~~[RFC2911](#)] plus those named in items 1 and 2 of this list.

1423 11.2.3 Get-Subscription-Attributes operation

1424 This operation allows a client to request the values of the attributes of a Subscription Object.

1425 A Printer MUST support this operation.

1426 This operation is almost identical to the Get-Job-Attributes operation (see [~~ipp-mod~~[RFC2911](#)] section 3.3.4).
1427 The only differences are that the operation is directed at a Subscription Object rather than a Job object, and
1428 the returned attribute group contains Subscription Object attributes rather than Job object attributes.

1429 11.2.3.1 Get-Subscription-Attributes Request

1430 The following groups of attributes are part of the Get-Subscription-Attributes request:

1431 Group 1: Operation Attributes

1432 Natural Language and Character Set:

1433 The “attributes-charset” and “attributes-natural-language” attributes as described in section [~~ipp-~~
1434 ~~mod~~[RFC2911](#)] 3.1.4.1.

1435

1436 Target:

1437 The “printer-uri” attribute which defines the target for this operation as described in [~~ipp-~~
1438 ~~mod~~[RFC2911](#)] section 3.1.5.

1439

1440 “notify-subscription-id” (integer (1:MAX)):

1441 The client MUST supply this attribute. The Printer MUST support this attribute. This attribute
1442 specifies the Subscription Object from which the client is requesting attributes. If the client omits this
1443 attribute, the Printer MUST reject this request with the ‘client-error-bad-request’ status code.

1444

1445 Requesting User Name:

1446 The “requesting-user-name” attribute SHOULD be supplied by the client as described in [~~ipp-~~
1447 ~~mod~~[RFC2911](#)] section 8.3.

1448

1449 “requested-attributes” (1setOf keyword):

1450 The client OPTIONALLY supplies this attribute. The Printer MUST support this attribute. This
1451 attribute specifies the attributes of the specified Subscription Object that the Printer MUST return in
1452 the response. Each value of this attribute is either an attribute name (defined in sections 5.3 and 5.4)
1453 or an attribute group name. The attribute group names are:

1454

- 1455 - ‘subscription-template’: all attributes that are both defined in section 5.3 and present on the specified
1456 Subscription Object (column 1 of Table 1).
- 1457 - ‘subscription-description’: all attributes that are both defined in section 5.4 and present on the
1458 specified Subscription Object (Table 2).
- 1459 - ‘all’: all attributes that are present on the specified Subscription Object.

1460

A Printer MUST support all these group names.

1461 If the client omits this attribute, the Printer MUST respond as if this attribute had been supplied with
1462 a value of ‘all’.

1463

1464 **11.2.3.2 Get-Subscription-Attributes Response**

1465 The Printer returns the following sets of attributes as part of the Get-Subscription-Attributes Response:

1466 Group 1: Operation Attributes

1467 Status Message:

1468 Same as [~~ipp-mod~~[RFC2911](#)].

1469

1470 Natural Language and Character Set:

1471 The “attributes-charset” and “attributes-natural-language” attributes as described in [~~ipp-~~
1472 ~~mod~~[RFC2911](#)] section 3.1.4.2. The “attributes-natural-language” MAY be the natural language of
1473 the Subscription Object, rather than the one requested.

1474

1475 Group 2: Unsupported Attributes

1476 See [~~ipp-mod~~[RFC2911](#)] section 3.1.7 for details on returning Unsupported Attributes.

1477

1478 The response NEED NOT contain the “requested-attributes” operation attribute with any supplied
1479 values (attribute keywords) that were requested by the client but are not supported by the Printer. If
1480 the Printer does return unsupported attributes referenced in the “requested-attributes” operation
1481 attribute and that attribute included group names, such as ‘all’, the unsupported attributes MUST
1482 NOT include attributes described in the standard but not supported by the implementation.

1483

1484 Group 3: Subscription Attributes

1485 This group contains a set of attributes with their current values. Each attribute in this group:

1486

1487 a) MUST be specified by the “requested-attributes” attribute in the request, AND

1488

b) MUST be present on the specified Subscription Object AND

1489

c) MUST NOT be restricted by the security policy in force. For example, a Printer MAY
1490 prohibit a client who is not the creator of a Subscription Object from seeing some or all of its
1491 attributes. See [~~ipp-mod~~[RFC2911](#)] section 8.

1492

The Printer can return the attributes of the Subscription Object in any order. The client MUST
1493 accept the attributes in any order.

1494

1495 **11.2.4 Get-Subscriptions operation**

1496 This operation allows a client to retrieve the values of attributes of all Subscription Objects belonging to a Job
1497 or Printer.

1498

A Printer MUST supported this operation.

1499

This operation is similar to the Get-Subscription-Attributes operation, except that this Get-Subscriptions
1500 operation returns attributes from possibly more than one object.

1501

This operation is similar to the Get-Jobs operation (see [~~ipp-mod~~[RFC2911](#)] section 3.2.6), except that the
1502 operation returns Subscription Objects rather than Job objects.

1503 11.2.4.1 Get-Subscriptions Request

1504 The following groups of attributes are part of the Get-Subscriptions request:

1505 Group 1: Operation Attributes

1506 Natural Language and Character Set:

1507 The “attributes-charset” and “attributes-natural-language” attributes as described in [~~ipp-~~
1508 [modRFC2911](#)] section 3.1.4.1.

1509

1510 Target:

1511 The “printer-uri” attribute which defines the target for this operation as described in [~~ipp-~~
1512 [modRFC2911](#)] section 3.1.5.

1513

1514 Requesting User Name:

1515 The “requesting-user-name” attribute SHOULD be supplied by the client as described in [~~ipp-~~
1516 [modRFC2911](#)] section 8.3.

1517

1518 “notify-job-id” (integer(1:MAX)):

1519 If the client specifies this attribute, the Printer returns the specified attributes of all Per-Job
1520 Subscription Objects associated with the Job whose “job-id” attribute value equals the value of this
1521 attribute. If the client does not specify this attribute, the Printer returns the specified attributes of all
1522 Per-Printer Subscription Objects. Note: there is no way to get all Per-Job Subscriptions.

1523

1524 “limit” (integer(1:MAX)):

1525 The client OPTIONALLY supplies this attribute. The Printer MUST support this attribute. It is an
1526 integer value that determines the maximum number of Subscription Objects that a client will receive
1527 from the Printer even if the “my-subscriptions” attribute constrains which Subscription Objects are
1528 returned. The limit is a “stateless limit” in that if the value supplied by the client is ‘N’, then only the
1529 first ‘N’ Subscription Objects are returned in the Get-Subscriptions Response. There is no
1530 mechanism to allow for the next ‘M’ Subscription Objects after the first ‘N’ Subscription Objects.
1531 If the client does not supply this attribute, the Printer responds with all applicable Subscription
1532 Objects.

1533

1534 “requested-attributes” (1setOf type2 keyword):

1535 The client OPTIONALLY supplies this attribute. The Printer MUST support this attribute. This
1536 attribute specifies the attributes of the specified Subscription Objects that the Printer MUST return in
1537 the response. Each value of this attribute is either an attribute name (defined in sections 5.3 and 5.4)
1538 or an attribute group name (defined in section 11.2.3.1). If the client omits this attribute, the Printer
1539 MUST respond as if the client had supplied this attribute with the one value: ‘notify-subscription-id’.

1540

1541 “my-subscriptions” (boolean):
1542 The client OPTIONALLY supplies this attribute. The Printer MUST support this attribute. If the
1543 value is ‘false’, the Printer MUST consider the Subscription Objects from all users as candidates. If
1544 the value is ‘true’, the Printer MUST return the Subscription Objects created by the requesting user
1545 of this request. If the client does not supply this attribute, the Printer MUST respond as if the client
1546 had supplied the attribute with a value of ‘false’. The means for authenticating the requesting user
1547 and matching the Subscription Objects is similar to that for Jobs which is described in [~~ipp-~~
1548 ~~mod~~RFC2911] section 8.
1549

1550 11.2.4.2 Get-Subscriptions Response

1551 The Printer returns the following sets of attributes as part of the Get-Subscriptions Response:

1552 Group 1: Operation Attributes

1553 Status Message:

1554 Same as [~~ipp-mod~~RFC2911].
1555

1556 Natural Language and Character Set:

1557 The “attributes-charset” and “attributes-natural-language” attributes as described in [~~ipp-~~
1558 ~~mod~~RFC2911] section 3.1.4.2.
1559

1560 Group 2: Unsupported Attributes

1561 Same as for Get-Subscription-Attributes.
1562

1563 Groups 3 to N: Subscription Attributes

1564 The Printer responds with one Subscription Attributes Group for each requested Subscription
1565 Object (see the “notify-job-id” attribute in the Operation Attributes Group of this operation).
1566

1567 The Printer returns Subscription Objects in any order.
1568

1569 If the “limit” attribute is present in the Operation Attributes group of the request, the number of
1570 Subscription Attributes Groups in the response MUST NOT exceed the value of the “limit” attribute.
1571

1572 If there are no Subscription Objects associated with the specified Job or Printer, the Printer MUST
1573 return zero Subscription Attributes Groups and it MUST NOT treat this case as an error, i.e., the
1574 status-code MUST be ‘successful-ok’ unless something else causes the status code to have some
1575 other value.
1576

1577 See the Group 3 response (Subscription Attributes Group) of the Get-Subscription-Attributes
1578 operation (section 11.2.3.2) for the attributes that a Printer returns in this group.
1579

1580 11.2.5 Renew-Subscription operation

1581 This operation allows a client to request the Printer to extend the lease on a Per-Printer Subscription Object.

1582 The Printer MUST support this operation.

1583 The Printer MUST accept this request for a Per-Printer Subscription Object in any of the target Printer's
1584 states, i.e., 'idle', 'processing', or 'stopped', but MUST NOT change the Printer's "printer-state" attribute.

1585 The Printer MUST reject this request for a Per-Job Subscription Object because it has no lease (see section
1586 5.4.3). The status code returned MUST be 'client-error-not-possible'.

1587 *Access Rights:* The authenticated user (see [~~IPP-MO~~RFC2911] section 8.3) performing this operation
1588 MUST either be the owner of the Per-Printer Subscription Object or have Operator or Administrator access
1589 rights for the Printer (see [~~IPP-MO~~RFC2911] sections 1 and 8.5). Otherwise, the Printer MUST reject the
1590 operation and return: the 'client-error-forbidden', 'client-error-not-authenticated', or 'client-error-not-
1591 authorized' status code as appropriate.

1592 11.2.5.1 Renew-Subscription Request

1593 The following groups of attributes are part of the Renew-Subscription Request:

1594 Group 1: Operation Attributes

1595 Natural Language and Character Set:

1596 The "attributes-charset" and "attributes-natural-language" attributes as described in [~~ipp-~~
1597 ~~mod~~RFC2911] section 3.1.4.1.

1598

1599 Target:

1600 The "printer-uri" attribute which defines the target for this operation as described in [~~ipp-~~
1601 ~~mod~~RFC2911] section 3.1.5.

1602

1603 "notify-subscription-id" (integer (1:MAX)):

1604 The client MUST supply this attribute. The Printer MUST support this attribute. This attribute
1605 specifies the Per-Printer Subscription Object whose lease the Printer MUST renew. If the client
1606 omits this attribute, the Printer MUST reject this request with the 'client-error-bad-request' status
1607 code.

1608

1609 Requesting User Name:

1610 The "requesting-user-name" (name(MAX)) attribute SHOULD be supplied by the client as
1611 described in [~~ipp-mod~~RFC2911] section 8.3.

1612

1613 Group 2: Subscription Template Attributes

1614

1615 “notify-lease-duration” (integer(0:MAX)):
1616 The client MAY supply this attribute. It indicates the number of seconds to renew the lease for the
1617 specified Subscription Object. A value of 0 requests an infinite lease (which MAY require Operator
1618 access rights). If the client omits this attribute, the Printer MUST use the value of the Printer’s
1619 “notify-lease-duration-default” attribute. See section 5.3.7 for more details.
1620

1621 11.2.5.2 Renew-Subscription Response

1622 The Printer returns the following sets of attributes as part of the Renew-Subscription Response:

1623 Group 1: Operation Attributes

1624 Status Message:

1625 Same as [~~ipp-mod~~[RFC2911](#)].
1626

1627 The following are some of the status codes returned:
1628

1629 **successful-ok:** The operation successfully renewed the lease on the Subscription Object for the requested
1630 duration..

1631 **successful-ok-ignored-or-substituted-attributes:** The operation successfully renewed the lease on the
1632 Subscription Object for some duration other than the amount requested.

1633 **client-error-not-possible:** The operation failed because the “notify-subscription-id” Operation attribute
1634 identified a Per-Job Subscription Object.

1635 **client-error-not-found:** The operation failed because the “notify-subscription-id” Operation attribute
1636 identified a non-existent Subscription Object.
1637

1638 Natural Language and Character Set:

1639 The “attributes-charset” and “attributes-natural-language” attributes as described in [~~ipp-~~
1640 ~~mod~~[RFC2911](#)] section 3.1.4.2. The “attributes-natural-language” MAY be the natural language of
1641 the Subscription Object, rather than the one requested.
1642

1643 Group 2: Unsupported Attributes

1644 See [~~ipp-mod~~[RFC2911](#)] section 3.1.7 for details on returning Unsupported Attributes.
1645

1646 Group 3: Subscription Attributes

1647 The Printer MUST return the following Subscription Attribute:

1648 “notify-lease-duration” (integer(0:MAX)):

1649 The value of this attribute MUST be the number of seconds that the Printer has granted for the lease
1650 of the Subscription Object (see section 5.3.7 for details, such as the value of this attribute when the
1651 Printer doesn’t support the requested value).
1652

1653

1654 **11.2.6 Cancel-Subscription operation**

1655 This operation allows a client to delete a Subscription Object and stop the Printer from sending more Event
1656 Notifications. Once performed, there is no way to reference the Subscription Object.

1657 A Printer MUST supported this operation.

1658 The Printer MUST accept this request in any of the target Printer's states, i.e., 'idle', 'processing', or
1659 'stopped', but MUST NOT change the Printer's "printer-state" attribute.

1660 If the specified Subscription Object is a Per-Job Subscription Object, the Printer MUST accept this request in
1661 any of the target Job's states, but MUST NOT change the Job's "job-state" attribute or affect the Job.

1662 *Access Rights:* The authenticated user (see [~~IPP-MOD~~RFC2911] section 8.3) performing this operation
1663 MUST either be the owner of the Subscription Object or have Operator or Administrator access rights for the
1664 Printer (see [~~IPP-MOD~~RFC2911] sections 1 and 8.5). Otherwise, the Printer MUST reject the operation
1665 and return: the 'client-error-forbidden', 'client-error-not-authenticated', or 'client-error-not-authorized' status
1666 code as appropriate.

1667 Note: There is no way to change any attributes on a Subscription Object, except the "notify-lease-duration"
1668 attribute (using the Renew-Subscription operation). In order to change other attributes, a client performs a
1669 Subscription Creation Operation and Cancel-Subscription operation on the old Subscription Object. If the
1670 client wants to avoid missing Event Notifications, it performs the Subscription Creation Operation first. If this
1671 order would create too many Subscription Objects on the Printer, the client reverses the order.

1672 **11.2.6.1 Cancel-Subscription Request**

1673 The following groups of attributes are part of the Cancel-Subscription Request:

1674 Group 1: Operation Attributes

1675 Natural Language and Character Set:

1676 The "attributes-charset" and "attributes-natural-language" attributes as described in [~~ipp-~~
1677 ~~mod~~RFC2911] section 3.1.4.1.

1678

1679 Target:

1680 The "printer-uri" attribute which defines the target for this operation as described in [~~ipp-~~
1681 ~~mod~~RFC2911] section 3.1.5.

1682

1683 "notify-subscription-id" (integer (1:MAX)):

1684 The client MUST supply this attribute. The Printer MUST support this attribute. This attribute
1685 specifies the Subscription Object that the Printer MUST cancel. If the client omits this attribute, the
1686 Printer MUST reject this request with the 'client-error-bad-request' status code.

1687

1688

Requesting User Name:

1689

The “requesting-user-name” attribute SHOULD be supplied by the client as described in [~~ipp-mod~~[RFC2911](#)] section 8.3.

1690

1691

1692

11.2.6.2 Cancel-Subscription Response

1693

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

1694

Group 1: Operation Attributes

1695

Status Message:

1696

Same as [~~ipp-mod~~[RFC2911](#)].

1697

1698

The following are some of the status codes returned:

1699

1700

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

1701

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

1702

1703

1704

Natural Language and Character Set:

1705

The “attributes-charset” and “attributes-natural-language” attributes as described in [~~ipp-mod~~[RFC2911](#)] section 3.1.4.2. The “attributes-natural-language” MAY be the natural language of the Subscription Object, rather than the one requested.

1706

1707

1708

1709

Group 2: Unsupported Attributes

1710

See [~~ipp-mod~~[RFC2911](#)] section 3.1.7 for details on returning Unsupported Attributes.

1711

1712

12 Conformance Requirements

1713

It is OPTIONAL to implement this Event Notification specification.

1714

If this Event Notification specification is implemented, Printers MUST:

1715

- meet the Conformance Requirements detailed in section 5 of [~~ipp-mod~~[RFC2911](#)].

1716

- support the Subscription Template Attributes Group in requests and the Subscription Attributes Group in responses.

1717

1718

- support all of the following attributes:

1719

a. REQUIRED Subscription Object attributes in section 5.

1720

b. REQUIRED Printer Description object attributes in section 6.

- 1721 c. REQUIRED attributes in Event Notification content in section 8.
- 1722 • send Event Notifications that conform to the requirements of the Delivery Method Document for
- 1723 each supported Delivery Method (the conformance requirements for Delivery Method Documents is
- 1724 specified in section 10).
- 1725 • support all operations as described in Table 16:

1726 **Table 16 – Conformance Requirements for Operations**

Operation	Conformance requirements
Create-Printer-Subscriptions (section 11.1.2)	REQUIRED
Create-Job-Subscriptions (section 11.1.1)	OPTIONAL
Get-Subscription-Attributes (section 11.2.2)	REQUIRED
Get-Subscriptions (section 11.2.4)	REQUIRED
Renew-Subscription (section 11.2.5)	REQUIRED
Cancel-Subscription (section 11.2.6)	REQUIRED

1727

1728 13 IANA Considerations

1729 This section contains the exact information for IANA to add to the IPP Registries according to the procedures

1730 defined in RFC 2911 [RFC2911] section 6.

1731 *Note to RFC Editors: Replace RFC NNNN below with the RFC number for this document, so that*

1732 *it accurately reflects the content of the information for the IANA Registry.*

1733 13.1 Attribute Registrations

1734 The attributes defined in this document will be published by IANA according to the procedures in RFC 2911

1735 [RFC2911] section 6.2 with the following path:

1736 <ftp.isi.edu/iana/assignments/ipp/attributes/>

1737 The registry entry will contain the following information:

<u>Subscription Template attributes:</u>	<u>Ref.</u>	
<u>Section:</u>		
<u>notify-recipient-uri (uri)</u>	<u>RFC NNNN</u>	<u>0</u>
<u>notify-events (1setOf type2 keyword)</u>	<u>RFC NNNN</u>	
<u>5.3.2</u>		
<u>notify-attributes (1setOf type2 keyword)</u>	<u>RFC NNNN</u>	
<u>5.3.3</u>		
<u>notify-user-data (octetString(63))</u>	<u>RFC NNNN</u>	
<u>5.3.4</u>		

1746

1747	<u>notify-charset (charset)</u>	RFC NNNN	
1748	5.3.5		
1749	<u>notify-natural-language (naturalLanguage)</u>	RFC NNNN	
1750	5.3.6		
1751	<u>notify-lease-duration (integer(0:67108863))</u>	RFC NNNN	
1752	5.3.7		
1753	<u>notify-time-interval (integer(0:MAX))</u>	RFC NNNN	
1754	5.3.8		
1755			
1756	<u>Subscription Description Attributes:</u>		
1757	<u>notify-subscription-id (integer (1:MAX))</u>	RFC NNNN	0
1758	<u>notify-sequence-number (integer (0:MAX))</u>	RFC NNNN	
1759	5.4.2		
1760	<u>notify-lease-expiration-time (integer(0:MAX))</u>	RFC NNNN	
1761	5.4.3		
1762	<u>notify-printer-up-time (integer(1:MAX))</u>	RFC NNNN	
1763	5.4.4		
1764	<u>notify-printer-uri (uri)</u>	RFC NNNN	
1765	5.4.5		
1766	<u>notify-job-id (integer(1:MAX))</u>	RFC NNNN	
1767	5.4.6		
1768	<u>notify-subscriber-user-name (name(MAX))</u>	RFC NNNN	
1769	5.4.7		
1770			
1771	<u>Printer Description Attributes:</u>		
1772	<u>printer-state-change-time (integer(1:MAX))</u>	RFC NNNN	0
1773	<u>printer-state-change-date-time (dateTime)</u>	RFC NNNN	
1774	6.2		
1775			
1776	<u>Attributes Only in Event Notifications</u>		
1777	<u>notify-subscribed-event (type2 keyword)</u>	RFC NNNN	
1778	8.1		
1779	<u>notify-text (text(MAX))</u>	RFC NNNN	
1780	8.2		
1781			

13.2 Keyword Attribute Value Registrations

1783 The keyword attribute values defined in this document will be published by IANA according to the
 1784 procedures in RFC 2911 [RFC2911] section 6.1 with the following path:

1785 <ftp://isi.edu/iana/assignments/ipp/attribute-values/>

1786 The registry entry will contain the following information:

1787 Keyword Attribute Values: Ref.
 1788 Section:
 1789 New Values for Existing Printer Description Attributes

1790 operations-supported (1setOf type2 enum) RFC NNNN
1791 7.1

1792

1793 **13.3 Operation Registrations**

1794 The operations defined in this document will be published by IANA according to the procedures in RFC 2911
1795 [RFC2911] section 6.4 with the following path:

1796 <ftp.isi.edu/iana/assignments/ipp/operations/>

1797 The registry entry will contain the following information:

1798	<u>Operations:</u>	<u>Ref.</u>
1799	<u>Section:</u>	
1800	<u>Create-Job-Subscriptions Operation</u>	<u>RFC NNNN</u>
1801	11.1.1	
1802	<u>Create-Printer-Subscriptions operation</u>	<u>RFC NNNN</u>
1803	11.1.2	
1804	<u>Job Creation Operations - Extensions</u>	<u>RFC NNNN</u>
1805	11.1.3	
1806	<u>Validate-Job Operation - Extensions</u>	<u>RFC NNNN</u>
1807	11.2.1	
1808	<u>Get-Printer-Attributes - Extensions</u>	<u>RFC NNNN</u>
1809	11.2.2	
1810	<u>Get-Subscription-Attributes operation</u>	<u>RFC NNNN</u>
1811	11.2.3	
1812	<u>Get-Subscriptions operation</u>	<u>RFC NNNN</u>
1813	11.2.4	
1814	<u>Renew-Subscription operation</u>	<u>RFC NNNN</u>
1815	11.2.5	
1816	<u>Cancel-Subscription operation</u>	<u>RFC NNNN</u>
1817	11.2.6	

1818

1819 **13.4 Status code Registrations**

1820 The status codes defined in this document will be published by IANA according to the procedures in RFC
1821 2911 [RFC2911] section 6.6 with the following path:

1822 <ftp.isi.edu/iana/assignments/ipp/status-codes/>

1823 The registry entry will contain the following information:

1824	<u>Status codes:</u>	<u>Ref.</u>
1825	<u>Section:</u>	
1826	<u>successful-ok-ignored-subscriptions (0x0003)</u>	<u>RFC NNNN</u>
1827	16.1	

1828	<u>client-error-ignored-all-subscriptions (0x0414)</u>	RFC NNNN
1829	16.2	
1830		
1831	<u>Status Codes in Subscription Attributes Groups:</u>	
1832	<u>client-error-uri-scheme-not-supported (0x040C)</u>	RFC NNNN
1833	17.1	
1834	<u>client-error-too-many-subscriptions (0x0415)</u>	RFC NNNN
1835	17.2	
1836	<u>successful-ok-too-many-events (0x0005)</u>	RFC NNNN
1837	17.3	
1838	<u>successful-ok-ignored-or-substituted-attributes (0x0001)</u>	
1839		RFC NNNN
1840	17.4	
1841		

1842 **13.5 Attribute Group tag Registrations**

1843 The attribute group tags defined in this document will be published by IANA according to the procedures in
 1844 RFC 2911 [RFC2911] section 6.5 with the following path:

1845 <ftp.isi.edu/iana/assignments/ipp/attribute-group-tags/>

1846 The registry entry will contain the following information:

1847	<u>Attribute Group Tags:</u>	Ref.
1848	<u>Section:</u>	
1849	<u>subscription-attributes-tag</u>	RFC NNNN
1850	18	
1851	<u>event-notification-attributes-tag</u>	RFC NNNN
1852	18	
1853		

1854 **13.6 Format for Event Notification Delivery Method Registration proposals**

1855 This section describes the procedures for registering Event Notification Delivery Method proposals with
 1856 IANA to be used with this document. Such Delivery Method proposals ~~can~~ that require a new URL scheme
 1857 MUST be IETF standards track documents ~~or vendor-defined documents~~ according to RFC 2717
 1858 [RFC2717]. In either case, they will be registered with IANA using procedures that extend those defined in
 1859 [ipp-mod] section 6 and 11.

1860 ~~These extension procedures are aligned with the guidelines as set forth by the IESG [IANA-CON]. Section~~
 1861 ~~13.1 defines the format and content for new registrations for consideration. IANA will reject registration~~
 1862 ~~proposals that leave out required information or do not follow the appropriate format described in Section~~
 1863 ~~13.1.~~

1864 ~~Implementers can, at any time, define new Event Notification Delivery Methods by proposing the complete~~
 1865 ~~specification to IANA:~~

1866 iana@iana.org

1867 ~~or by filling out the appropriate form on the IANA web pages (<http://www.iana.org>).~~

1868 ~~IANA will forward the registration proposal to the IPP Designated Expert who will review the proposal with a~~
1869 ~~mailing list that the Designated Expert keeps for this purpose. Initially, that list will be the mailing list used by~~
1870 ~~the IPP WG:~~

1871 ipp@pwg.org

1872 ~~even after the IPP WG is disbanded as permitted by [IANA-CON]. The IPP Designated Expert is appointed~~
1873 ~~by the IESG Area Director responsible for IPP, according to [IANA-CON].~~

1874 ~~When a Delivery Method Document is approved, the IPP Designated Expert becomes the point of contact for~~
1875 ~~any future maintenance that might be required for that registration.~~

1876 ~~13.1~~**13.7 Format and Requirements for IPP Delivery Method Registration Proposals**

1877 This section defines the format and requirements for an IPP Event Notification Delivery Method Registration
1878 Proposal. A Delivery Method Registration Proposal:

1879 1. MUST contain the following information:

1880 Type of registration: IPP Event Notification Delivery Method

1881 Name of this delivery method:

1882 Proposed URL scheme name of this delivery method:

1883 Name of proposer:

1884 Address of proposer:

1885 Email address of proposer:

1886 Is this delivery method REQUIRED or OPTIONAL for conformance to the IPP Event Notification
1887 Specification document:

1888 Is this delivery method defining Machine Consumable and/or Human Consumable content:

1889 2. MUST meet the conformance requirements for Delivery Method Documents specified in section 10.

1890 **14 Internationalization Considerations**

1891 This IPP Notification specification continues support for the internationalization of [~~ipp-mod~~[RFC2911](#)] of
1892 attributes containing text strings and names. Allowing a Subscribing Client to specify a different natural
1893 language and charset for each Subscription Object increases the internationalization support.

1894 The Printer MUST be able to localize the content of Human Consumable Event Notifications and to localize
1895 the value of “notify-text” attribute in Machine Consumable Event Notifications that it sends to Notification
1896 Recipients. For localization, the Printer MUST use the value of the “notify-charset” attribute and the “notify-
1897 natural-language” attribute in the Subscription Object supplied by the Subscribing Client.

1898 15 Security Considerations

1899 By far the biggest security concern is the abuse of notification: sending unwanted Event Notifications to third
1900 parties (i.e., spam). The problem is made worse by notification addresses that may be redistributed to
1901 multiple parties (e.g., mailing lists). There exist scenarios where third party notification is required (see
1902 Scenario #2 and #3 in [ipp-not-req]). The fully secure solution would require active agreement of all
1903 recipients before sending out anything. However, requirement #9 in [ipp-req] (“There is no requirement for
1904 IPP Printer receiving the print request to validate the identity of an Event recipient”) argues against this.
1905 Certain systems may decide to disallow third party Event Notifications (a traditional fax model).

1906 Clients submitting Notification requests to the IPP Printer has the same security issues as submitting an IPP/1.1
1907 print job request. The same mechanisms used by IPP/1.1 can therefore be used by the client Notification
1908 submission. Operations that require authentication can use the HTTP authentication. Operations that require
1909 privacy can use the HTTP/TLS privacy.

1910 The Notification access control model should be similar to the IPP access control model for Jobs. Creating a
1911 Per-Printer Subscription Object is associated with a user. Only the creator or an Operator can cancel the
1912 Subscription Object. The system may limit the listing of items to only those items owned by the user. Some
1913 Subscription Objects (e.g., those that have a lifetime longer than a job) can be done only by privileged users
1914 (users having Operator and/or Administrator access rights), if that is the authorization policy.

1915 The standard security concerns (delivery to the right user, privacy of content, tamper proof content) apply to
1916 the Delivery Method. IPP should use the security mechanism of the Delivery Method used. Some delivery
1917 mechanisms are more secure than others. Therefore, sensitive Event Notifications should use the Delivery
1918 Method that has the strongest security.

1919 16 Status Codes

1920 The following status codes are defined as extensions for Notification and are returned as the value of the
1921 “status-code” parameter in the Operation Attributes Group of a response (see [~~ipp-mod~~RFC2911] section
1922 3.1.6.1). Operations in this document can also return the status codes defined in section 13 of [~~ipp-~~
1923 ~~mod~~RFC2911]. The ‘successful-ok’ status code is an example of such a status code.

1924 16.1 successful-ok-ignored-subscriptions (0x0003)

1925 The Subscription Creation Operation was unable to create all requested Subscription Objects.

1926 For a Create-Job-Subscriptions or Create-Printer-Subscriptions operation, this status code means that the
1927 Printer created one or more Subscription Objects, but not all requested Subscription Objects.

1928 For a Job Creation operation, this status code means that the Printer created the Job along with zero or more
1929 Subscription Objects. The Printer returns this status code even if other job attributes are unsupported or in
1930 conflict. That is, if an IPP Printer finds a warning that would allow it to return ‘successful-ok-ignored-

1931 subscriptions' and either 'successful-ok-ignored-or-substituted-attributes' and/or 'successful-ok-conflicting-
1932 attributes', it MUST return 'successful-ok-ignored-subscriptions'.

1933 **16.2 client-error-ignored-all-subscriptions (0x0414)**

1934 This status code is the same as 'successful-ok-ignored-subscriptions' except that only the Create-Job-
1935 Subscriptions and Create-Printer-Subscriptions operation return it. They return this status code only when the
1936 Printer creates zero Subscription Objects.

1937 **17 Status Codes in Subscription Attributes Groups**

1938 This section contains values of the "notify-status-code" ([type2 enum](#)) attribute that the Printer returns in a
1939 Subscription Attributes Group in a response when the corresponding Subscription Object:

- 1940 1. is not created or
- 1941 2. is created and some of the client-supplied attributes are not supported.

1942 The following sections are ordered in decreasing order of importance of the status-codes.

1943 **17.1 client-error-uri-scheme-not-supported (0x040C)**

1944 This status code is defined in [~~ipp-mod~~[RFC2911](#)]. This document extends its meaning and allows it to be in a
1945 Subscription Attributes Group of a response.

1946 The scheme of the client-supplied URI in a "notify-recipient-uri" Subscription Template Attribute in a
1947 Subscription Creation Operation is not supported. See section 0.

1948 **17.2 client-error-too-many-subscriptions (0x0415)**

1949 The number of Subscription Objects supported by the Printer would be exceeded if this Subscription Object
1950 were created (see section 5.2).

1951 **17.3 successful-ok-too-many-events (0x0005)**

1952 The client supplied more Events in the "notify-events" operation attribute of a Subscription Creation Operation
1953 than the Printer supports, as indicated in its "notify-max-events-supported" Printer attribute (see section
1954 5.3.2).

1955 **17.4 successful-ok-ignored-or-substituted-attributes (0x0001)**

1956 This status code is defined in [~~ipp-mod~~[RFC2911](#)]. This document extends its meaning to include unsupported
1957 Subscription Template Attributes and it can appear in a Subscription Attributes Group.

18 Encodings of Additional Attribute Tags

This section assigns values to two attributes tags as extensions to the encoding defined in [~~ipp-pro~~RFC2910]).

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

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

The following table specifies the values for the delimiter tags:

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

19 References

[IANA-CON]

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

~~[ipp-mod]~~

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

[ipp-not-req]

deBry, R., Lewis, H., Hastings, T., “Internet Printing Protocol/1.1: Requirements for IPP Notifications”, <draft-ietf-ipp-not-054.txt>, work in progress, ~~July 6, 2000~~January 23, 2001.

~~[ipp-pro]~~

~~Herriot, R., Butler, S., Moore, P., Tuner, R., “Internet Printing Protocol/1.1: Encoding and Transport”, <draft-ietf-ipp-protocol-v11-06.txt>, work in progress, May 30, 2000.~~

[ipp-prog]

Hastings, T., Bergman, R., Lewis, H., “IPP: Job Progress Attributes”, <draft-ietf-ipp-job-prog-030.txt> work in progress, ~~July 6, 2000~~January 23, 2001.

[ipp-set]

Kugler, C., Hastings, T., Herriot, R., Lewis, H., “Internet Printing Protocol (IPP): Job and Printer Set Operations”, <draft-ietf-ipp-job-printer-set-ops-032.txt>, work in progress, ~~March 23, 2000~~January 22, 2001.

- 1985 [RFC2026]
1986 S. Bradner, "The Internet Standards Process -- Revision 3", RFC 2026, October 1996.
- 1987 [RFC2119]
1988 S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, March 1997
- 1989 [RFC2566]
1990 deBry, R., , Hastings, T., Herriot, R., Isaacson, S., Powell, P., "Internet Printing Protocol/1.0: Model and
1991 Semantics", RFC 2566, April 1999.
- 1992 [RFC2567]
1993 Wright, D., "Design Goals for an Internet Printing Protocol", RFC 2567, April 1999.
- 1994 [RFC2568]
1995 Zilles, S., "Rationale for the Structure and Model and Protocol for the Internet Printing Protocol", RFC
1996 2568, April 1999.
- 1997 [RFC2569]
1998 Herriot, R., Hastings, T., Jacobs, N., Martin, J., "Mapping between LPD and IPP Protocols", RFC 2569,
1999 April 1999.
- 2000 [\[RFC2717\]](#)
2001 R. Petke and I. King, "Registration Procedures for URL Scheme Names", RFC 2717, November 1999.
- 2002 [\[RFC2910\]](#)
2003 Herriot, R., Butler, S., Moore, P., Turner, R., "Internet Printing Protocol/1.1: Encoding and Transport",
2004 RFC 2910, September 2000.
- 2005 [\[RFC2911\]](#)
2006 deBry, R., , Hastings, T., Herriot, R., Isaacson, S., Powell, P., "Internet Printing Protocol/1.1: Model and
2007 Semantics", RFC 2911, September 2000.

2008 **20 Author's Addresses**

- 2009 Robert Herriot
2010 Xerox Corporation
2011 3400 Hillview Ave., Bldg #1
2012 Palo Alto, CA 94304
2013
2014 Phone: 650-813-7696
2015 Fax: 650-813-6860
2016 Email: robert.herriot@pahv.xerox.com
2017
2018 Tom Hastings

2019 Xerox Corporation
2020 737 Hawaii St. ESAE 231
2021 El Segundo, CA 90245
2022
2023 Phone: 310-333-6413
2024 Fax: 310-333-5514
2025 e-mail: hastings@cp10.es.xerox.com
2026
2027 Scott A. Isaacson
2028 Novell, Inc.
2029 122 E 1700 S
2030 Provo, UT 84606
2031
2032 Phone: 801-861-7366
2033 Fax: 801-861-2517
2034 e-mail: sisaacson@novell.com
2035
2036 Roger deBry
2037 Utah Valley State College
2038 Orem, UT 84058
2039
2040 Phone: (801) 222-8000
2041 EMail: debryro@uvsc.edu
2042
2043 Jay Martin
2044 Underscore Inc.
2045 9 Jacqueline St.
2046 Hudson, NH 03051-5308
2047 603-889-7000
2048 fax: 775-414-0245
2049 e-mail: jkm@underscore.com
2050
2051 Michael Shepherd
2052 Xerox Corporation
2053 800 Phillips Road MS 128-51E
2054 Webster, NY 14450
2055
2056 Phone: 716-422-2338
2057 Fax: 716-265-8871
2058 e-mail: mshepherd@crt.xerox.com
2059

2060 Ron Bergman
2061 Hitachi Koki Imaging Solutions
2062 1757 Tapo Canyon Road
2063 Simi Valley, CA 93063-3394
2064
2065 Phone: 805-578-4421
2066 Fax: 805-578-4001
2067 Email: rbergma@hitachi-hkis.com

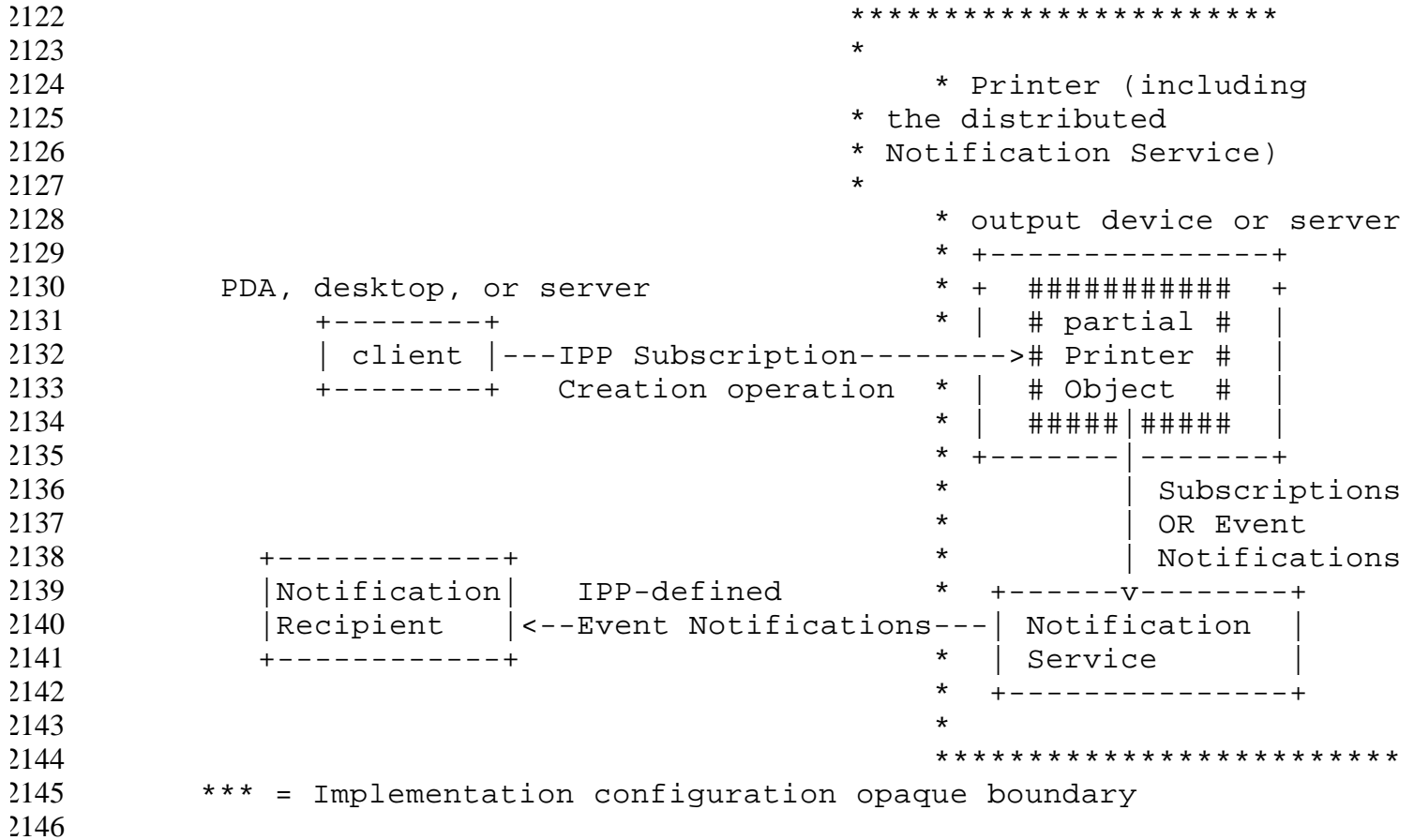
2068 **A. Appendix - Model for Notification with Cascading Printers**

2069 With this model (see Figure 2), there is an intervening Print server between the human user and the output-
2070 device. So the system effectively has two Printers. There are two cases to consider.

- 2071 1. When the Printer 1 (in the server) generates Events, the system behaves like the client and Printer in Figure
2072 1. In this case, Printer 1 sends Event Notifications that are shown as Event Notifications (A) of Figure 2,.
- 2073 2. When the Printer 2 (in the output-device) generates Events, there are two possible system configurations:
 - 2074 a) Printer 1 forwards the client-supplied Subscription Creation Operations to the downstream Printer 2
2075 and lets Printer 2 send the Event Notifications directly to the Notification Recipients supplied by the
2076 Client (Event Notifications(C) in the diagram).
 - 2077 b) Printer 1 performs the client-supplied Subscription Creation Operations and also forwards the
2078 Subscription Creation Operations to Printer 2 with the Notification Recipient changed to be the Printer
2079 1. When an Event occurs in Printer 2, Printer 2 sends the Event Notification (B) to Notification
2080 Recipient of Printer 1, which relays the received Event Notification (B) to the client-supplied
2081 Notification Recipient (as Event Notifications(A) in the diagram). Note, when a client performs a
2082 Subscription Creation Operation, Printer 1 need not forward the Subscription Creation Operation to
2083 Printer 2 if it would create a duplicate Subscription Object on Printer 2.

2084 Note: when Printer 1 is forwarding Subscription Creation Operations to Printer 2, it may request Printer 2 to
2085 create additional Subscription Objects (called “piggy-backing”). Piggy-backing is useful when:

- 2086 • Device A is configured to accept (IPP or non-IPP) requests from other servers.
- 2087 • Server S wants to receive Job Events that the client didn't request and Server S wants these Events
2088 for jobs it submits and not for other jobs.



2147 **Figure 3 – Opaque Use of a Notification Service Transparent to the Client**

2148 **C. Appendix - Extended Notification Recipient**

2149 The model allows for an extended Notification Recipient that is itself a notification service that forwards each
2150 Event Notification to another recipient (called the Ultimate Notification Recipient in this section). The Delivery
2151 Method to the Ultimate Recipient is probably different from the Delivery Method used by the Printer to the
2152 extended Notification Recipient.

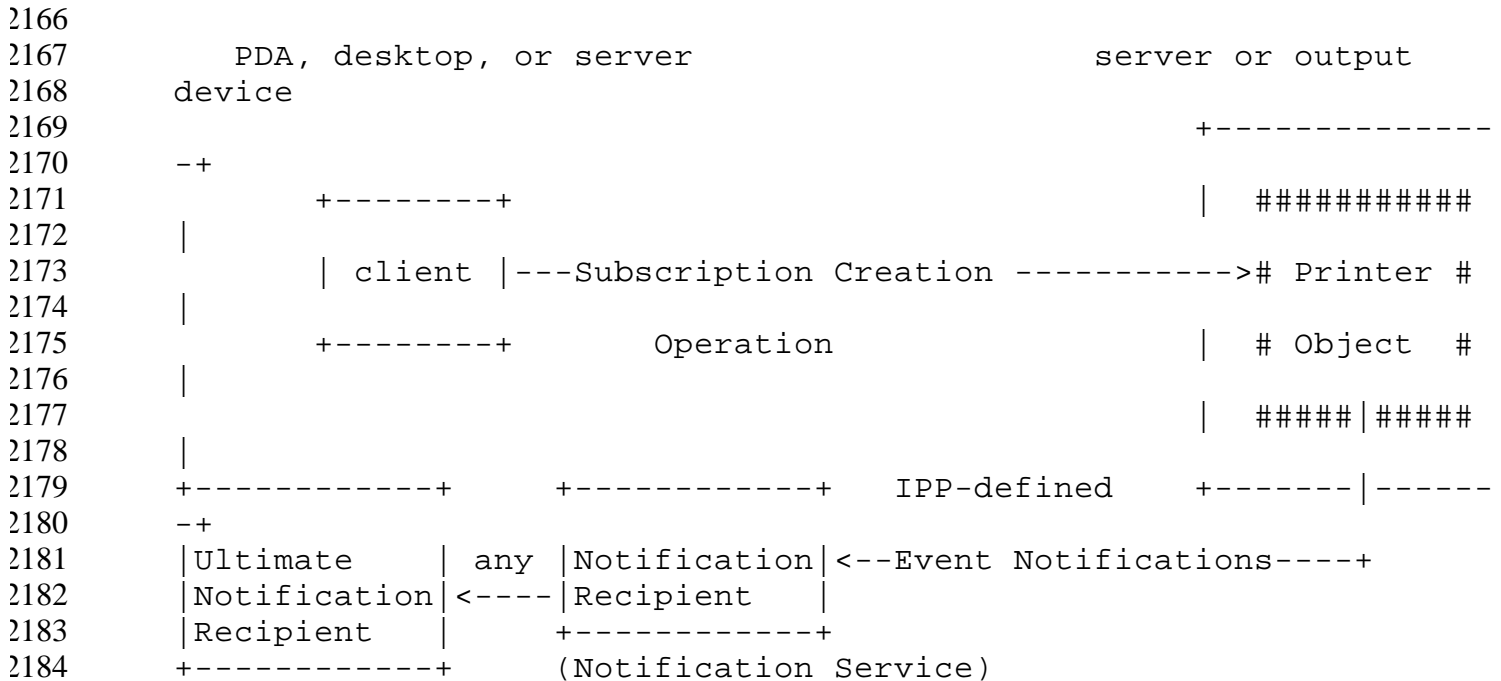
2153 This extended Notification Recipient is transparent to the Printer but not to the client.

2154 When a client performs a Subscription Creation Operation, it specifies the extended Notification Recipient as
2155 it would any Notification Recipient. In addition, the client specifies the Ultimate Notification Recipient in the
2156 Subscription Creation Operation in a manner specified by the extended Notification Recipient. Typically, it is
2157 either some bytes in the value of “notify-user-data” or some additional parameter in the value of “notify-
2158 recipient-uri”. The client also subscribes directly with the extended Notification Recipient (by means outside
2159 this document), since it is a notification service in its own right.

2160 The IPP Printer treats the extended Notification Recipient like any other Notification Recipient and the IPP
2161 Printer is not aware of the forwarding. The Delivery Method that the extended Notification Recipient uses for

2162 delivering the Event Notification to the Ultimate Notification Recipient is beyond the scope of this document
 2163 and is transparent to the IPP Printer.

2164 Examples of this extended Notification Recipient are paging, immediate messaging services, general
 2165 notification services, and NOS vendors' infrastructure. Figure 4 shows this approach.



2185 **Figure 4 – Use of an Extended Notification Recipient transparent to the Printer**

2186 D. Appendix - Details about Conformance Terminology

2187 The following paragraphs provide more details about conformance terminology.

2188 **REQUIRED** - an adjective used to indicate that a conforming IPP Printer implementation **MUST** support the
 2189 indicated operation, object, attribute, attribute value, status code, or out-of-band value in requests and
 2190 responses. See [[ipp-modRFC2911](#)] "Appendix A - Terminology for a definition of "support". *Since*
 2191 *support of this entire Notification specification is OPTIONAL for conformance to IPP/1.0 or*
 2192 *IPP/1.1, the use of the term REQUIRED in this document means "REQUIRED if this*
 2193 *OPTIONAL Notification specification is implemented"*.

2194 **RECOMMENDED** - an adjective used to indicate that a conforming IPP Printer implementation is
 2195 recommended to support the indicated operation, object, attribute, attribute value, status code, or out-of-
 2196 band value in requests and responses. *Since support of this entire Notification specification is*
 2197 *OPTIONAL for conformance to IPP/1.0 or IPP/1.1, the use of the term RECOMMENDED in*
 2198 *this document means "RECOMMENDED if this OPTIONAL Notification specification is*
 2199 *implemented"*.

2200 **OPTIONAL** - an adjective used to indicate that a conforming IPP Printer implementation MAY, but is NOT
2201 REQUIRED to, support the indicated operation, object, attribute, attribute value, status code, or out-of-
2202 band value in requests and responses.

2203 **E. Appendix - Object Model for Notification**

2204 This section describes the Notification object model that adds a Subscription Object which together with the
2205 Job and Printer object provide the complete Notification semantics.

2206 The object relationships can be seen pictorially as:

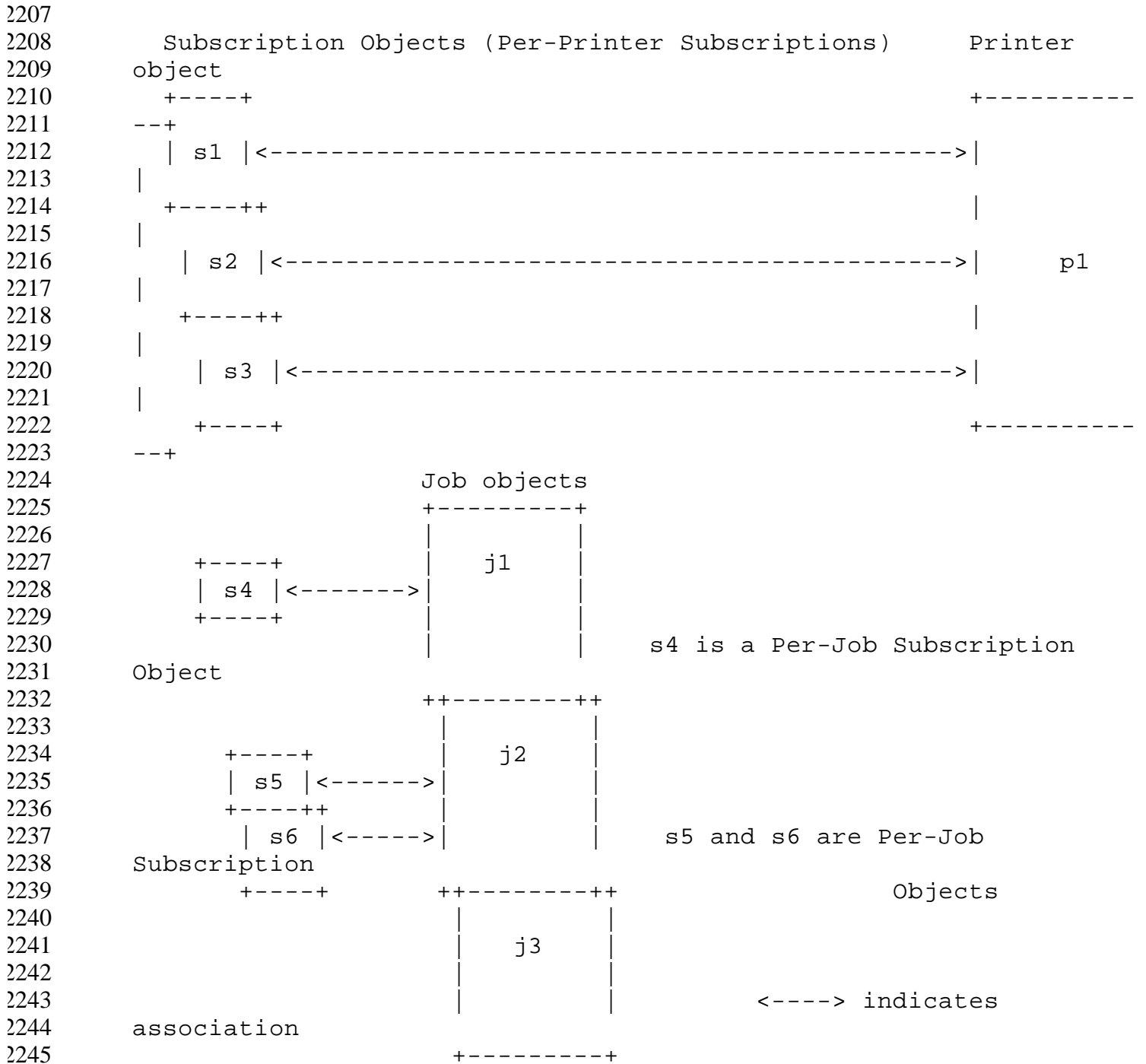


Figure 5 – Object Model for Notification

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

2249 **E.1 Appendix - Object relationships**

2250 This sub-section defines the object relationships between the Printer, Job, and Subscription Objects by
2251 example. Whether Per-Printer Subscription Objects are actually contained in a Printer object or are just bi-
2252 directionally associated with them in some way is IMPLEMENTATION DEPENDENT and is transparent to
2253 the client. Similarly, whether Per-Job Subscription Objects are actually contained in a Job object or are just
2254 bi-directionally associated with them in some way is IMPLEMENTATION DEPENDENT and is transparent
2255 to the client. The object relationships are defined as follows:

2256 **E.2 Printer Object and Per-Printer Subscription Objects**

- 2257 1. The Printer object contains (is associated with) zero or more Per-Printer Subscription Objects (p1
2258 contains s1-s3 Per-Printer Subscription Objects).
- 2259 2. Each Per-Printer Subscription Object (s1, s2, and s3) is contained in (or is associated with) exactly
2260 one Printer object (p1).

2261 **E.3 Job Object and Per-Job Subscription Objects**

- 2262 1. A Job object (j1, j2, j3) is associated with zero or more Per-Job Subscription Objects (s4-s6). Job j1
2263 is associated with Per-Job Subscription Object s4, Job j2 is associated with Per-Job Subscription
2264 Objects s5 and s6, and Job j3 is not associated with any Per-Job Subscription Object.
- 2265 2. Each Per-Job Subscription Object is associated with exactly one Job object.

2266 **F. Appendix - Per-Job versus Per-Printer Subscription Objects**

2267 Per-Job and Per-Printer Subscription Objects are quite similar. Either type of Subscription Object can
2268 subscribe to Job Events, Printer Events, or both. Both types of Subscription Objects can be queried using the
2269 Get-Subscriptions and Get-Subscription-Attributes operations and canceled using the Cancel-Subscription
2270 operation. Both types of Subscription Objects create Subscription Objects which have the same Subscription
2271 Object attributes defined. However, there are some semantic differences between Per-Job Subscription
2272 Objects and Per-Printer Subscription Objects. A Per-Job Subscription Object is established by the client
2273 when submitting a job and after creating the job using the Create-Job-Subscriptions operation by specifying
2274 the “job-id” of the Job with the “notify-job-id” attribute. A Per-Printer Subscription Object is established
2275 between a client and a Printer using the Create-Printer-Subscriptions operation. Some specific differences
2276 are:

- 2277 1. A client usually creates one or more Per-Job Subscription Objects as part of the Job Creation
2278 operations (Create-Job, Print-Job, and Print-URI), rather than using the OPTIONAL Create-Job-
2279 Subscriptions operation, especially since Printer implementations NEED NOT support the Create-Job-
2280 Subscriptions operation, since it is OPTIONAL.

- 2281 2. For Per-Job Subscription Objects, the Subscription Object is only valid while the job is “not-
2282 complete” (see sections 5.4.3) while for the Per-Printer Subscription Objects, the Subscription Object
2283 is valid until the time (in seconds) that the Printer returned in the “notify-lease-expiration-time”
2284 operation attribute.
- 2285 3. Job Events in a Per-Job Subscription Object apply only to “one job” (the Job created by the Job
2286 Creation operation or references by the Create-Job-Subscriptions operation) while Job Events in a
2287 Per-Printer Subscription Object apply to ALL jobs contained in the IPP Printer.

2288 **G. Appendix: Full Copyright Statement**

2289 Copyright (C) The Internet Society (1998,1999,2000,[2001](#)). All Rights Reserved

2290 This document and translations of it may be copied and furnished to others, and derivative works that
2291 comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and
2292 distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and
2293 this paragraph are included on all such copies and derivative works. However, this document itself may not
2294 be modified in any way, such as by removing the copyright notice or references to the Internet Society or
2295 other Internet organizations, except as needed for the purpose of developing Internet standards in which case
2296 the procedures for copyrights defined in the Internet Standards process must be followed, or as required to
2297 translate it into languages other than English.

2298 The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its
2299 successors or assigns.

2300 This document and the information contained herein is provided on an “AS IS” basis and THE INTERNET
2301 SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES,
2302 EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE
2303 OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED
2304 WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.