

1 INTERNET-DRAFT 5 ISSUES are highlighted like this.
2 <draft-ietf-ipp-not-spec-01.doc>

S. Isaacson
Novell, Inc.
J. Martin
Underscore
R. deBry
Utah Valley State College
T. Hastings
Xerox Corporation
M. Shepherd
Xerox Corporation
R. Bergman
Dataproducts Corp.
~~September 10~~ October 14, 1999

15 Internet Printing Protocol/1.0 & 1.1: **IPP Event Notification Specification—Alternative**

16 Copyright (C) The Internet Society (1999). All Rights Reserved.

17 Status of this Memo

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

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

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

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

27 **Abstract**

28 This document describes an extension to the IPP/1.0 & IPP/1.1 model that allows a client to subscribe to
29 printing related events. Subscriptions include "Per-Job subscriptions" and "Per-Printer subscriptions". One
30 or more Per-Job Submission subscriptions are specified by the client when submitting a job. Additional
31 Per-Job and Per-Printer subscriptions are created by performing separate explicit ~~Create-Job-Subscription~~
32 Create-Printer-Subscription operations, respectively on the Printer. Subscriptions are modeled as
33 Subscription objects. Four other operations are defined for subscription objects: get attributes, get
34 subscriptions, renew a subscription, and cancel a subscription.

35 A subscription request and the Subscription object includes: the names of Job and/or Printer events, the
36 Notification Recipient URL, the ~~Notification content MIME type~~ text format if Human Consumable
37 notification is requested, possibly some opaque data, and the charset and natural language. In addition, the
38 subscription request includes: the requested lease time and persistency for the subscription. When the
39 event occurs, a notification is generated and delivered using the information specified in the subscription.

40 The full set of IPP documents includes:

41 Design Goals for an Internet Printing Protocol [IPP-REQ]

42 Rationale for the Structure and Model and Protocol for the Internet Printing Protocol [IPP-RAT]

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

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

45 Internet Printing Protocol/1.0: Implementer's Guide [IPP-IIG]

46 Mapping between LPD and IPP Protocols [IPP LPD]

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
50 in a printing protocol for the Internet. It identifies requirements for three types of users: end users,
51 operators, and administrators. It calls out a subset of end user requirements that are satisfied in IPP/1.0.
52 Operator and administrator requirements are out of scope for version 1.0. A few OPTIONAL operator
53 operations have 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
56 IPP 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
58 objects, their attributes, and their operations that are independent of encoding and transport. It introduces a
59 Printer 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
64 transporting over HTTP a message body whose Content-Type is "application/ipp". This document defines
65 a new scheme named 'ipp' for identifying IPP printers and jobs. Finally, this document defines
66 interoperability rules for supporting IPP/1.0 clients.

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

76	1	Introduction	6
77	1.1	Notification Overview	6
78	2	Model for Per-Job and Per-Printer Subscription and Event Notification.....	8
79	2.1	Model for Per-Job Subscription and Notification	8
80	2.2	Model for Per-Printer Subscription and Notification	10
81	2.3	Relationship between the Printer object and the Notification Delivery Service	11
82	2.3.1	Use of a Notification Service transparently to clients	11
83	2.3.2	Use of Notification Service transparently to the IPP Printer.....	12
84	3	Terminology	12
85	3.1	Conformance Terminology	12
86	3.2	Other terminology	13
87	4	Object Model for Notification.....	15
88	4.1	Object relationships	16
89	5	Subscription Object attributes	17
90	5.1	notify-recipient (uri)	19
91	5.2	notify-events (1setOf type2 keyword).....	19
92	5.3	notify-text-format (mimeMediaType).....	22
93	5.4	subscriber-user-data (octetString(63)).....	23
94	5.5	notify-charset (charset).....	23
95	5.6	notify-natural-language (naturalLanguage).....	23
96	5.7	request-id.....	23
97	5.8	subscription-id (integer (1:MAX))	24
98	5.9	notify-lease-expiration-time (integer(0:MAX))	24
99	5.10	printer-uri (uri).....	24
100	5.11	subscriber-user-name (name(MAX))	24
101	5.12	notify-printer-up-time (integer(1:MAX)).....	25
102	5.13	notify-persistence-granted (boolean).....	25
103	6	Printer Description Attributes related to Notification	25
104	6.1	notify-schemes-supported (1setOf uriScheme)	26
105	6.2	notify-events-default (1setOf type2 keyword)	26
106	6.3	notify-events-supported (1setOf type2 keyword).....	26
107	6.4	max-events-supported (integer(5:MAX)).....	27
108	6.5	notify-text-format-supported (1setOf mimeMediaType)	27
109	6.6	max-job-subscriptions-supported (integer(1:MAX))	27
110	6.7	max-printer-subscriptions-supported (integer(0:MAX)).....	28
111	6.8	notify-lease-time-supported (rangeOfInteger(0:MAX))	28
112	6.9	notify-lease-time-default (integer(0:MAX))	28
113	6.10	persistent-jobs-supported (boolean)	28
114	6.11	persistent-subscriptions-supported (boolean).....	28
115	6.12	printer-state-change-time (integer(1:MAX)).....	28
116	6.13	printer-state-change-date-time (dateTime).....	29

117	7	Notification Content	29
118	7.1	Notification content MIME media type formats	29
119	7.2	Machine Consumable form	29
120	7.3	Human Consumable form.....	30
121	7.4	Notification content attributes common to Job and Printer events.....	30
122	7.5	Additional Notification content attributes for Job events <i>only</i>	32
123	7.6	Additional Notification content attributes for Printer events <i>only</i>	32
124	8	Operations for notification	32
125	8.1	Operations for Per-Job Subscriptions <i>only</i>	33
126	8.1.1	Job Creation Operations (Create-Job, Print-Job, Print-URI) and Validate-Job	33
127	8.1.2	Create-Job-Subscription operation	35
128	8.2	Operations for Per-Printer Subscriptions <i>only</i>	37
129	8.2.1	Create-Printer-Subscription operation.....	38
130	8.3	Common Operations for Per-Job and Per-Printer Subscriptions.....	41
131	8.3.1	Get-Subscription-Attributes operation	41
132	8.3.2	Get-Subscriptions operation	41
133	8.3.3	Renew-Subscription operation	42
134	8.3.4	Cancel- Subscription operation	44
135	9	Comparison of Per-Job versus Per-Printer Subscriptions	44
136	10	Conformance Requirements	45
137	11	IANA Considerations	45
138	12	Internationalization Considerations.....	45
139	13	Security Considerations.....	46
140	14	Status Codes	46
141	14.1	'successful-ok-ignored-subscriptions' (0x0003)	46
142	14.2	client-error-uri-notification-scheme-not-supported (0x04??)	47
143	14.3	server-error-too-many-subscriptions (0x04??).....	47
144	14.4	server-error-too-many-events (0x04??).....	47
145	15	Additions to the IPP Encoding and Transport Document	47
146	16	References	48
147	17	Author's Addresses	49
148	A.	Appendix - Summary of Notification Attribute Usage	50
149	18	Appendix C: Full Copyright Statement.....	54
150			

151		
		Tables
152	Table 1 - Summary of Per-Job and Per-Printer Subscription operations	8
153	Table 2 - Subscription object attributes.....	18
154	Table 3 - Printer Description attributes associated with Notification	26
155	Table 4 - Common Job and Printer Notification content attributes.....	30
156	Table 5 - Additional Notification content attributes for Job events <i>only</i>	32
157	Table 6 - Additional Notification content attributes for Printer events <i>only</i>	32
158	Table 7 - Member attributes of the "job-notify" collection operation attribute.....	33
159	Table 8 - Conformance Requirements for Operations	45
160		Figures
161	Figure 1 - Client-Printer Per-Job Subscription and Notification Model	9
162	Figure 2 - Client-Server-Printer Per-Job Subscription and Notification Model.....	10
163	Figure 3 - Client-Printer Per-Printer Subscription and Notification Model	10
164	Figure 4 - Opaque Use of a Notification Service Transparent to the Client	11
165	Figure 5 - Use of a Notification Service transparent to the IPP Printer	12
166	Figure 6 - Object Model for Notification	16
167		

168

169 1 Introduction

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

173 Internet Printing Protocol/1.0 & 1.1: "Collection Attribute Syntax" [ipp-coll]

174 Internet Printing Protocol/1.0 & 1.1: "Job Progress Attributes" [ipp-prog]

175 Internet Printing Protocol/1.0 & 1.1: "Notification Change History" [ipp-not-hist]

176 In addition, each notification delivery method, whether REQUIRED or OPTIONAL, is described in
177 separate documents:

178 Internet Printing Protocol/1.0 & 1.1: "Notification Delivery Method xxx [TBD]

179 Internet Printing Protocol/1.0 & 1.1: "Notification Delivery Method yyy [TBD]

180

181 The rest of this document is laid out as follows:

- 182 - The rest of Section 1.1 is an overview of IPP Notification.
- 183 - Section 2 is the model for network entities that use IPP notification, including clients (desktop and
184 servers), IPP Printers (servers and devices), and Notification Recipients.
- 185 - Section 3 is the terminology used throughout the document.
- 186 - Section 4 is the object model for notification, including Job, Printer, and Subscription objects.
- 187 - Section 5 and 6 defines the notification attributes for each of the Subscription and Printer objects.
- 188 - Section 7 defines the content of Human Consumable and Machine Consumable Event Notification
189 contents.
- 190 - Sections 8 and 9 define the Per-Job and Per-Printer Subscription operations.
- 191 - Section 10 and 11 define the conformance requirements and IANA requirements, respectively.
- 192 - Section 12 - 14 cover Internationalization, Security, and Status codes.

193 1.1 Notification Overview

194 A client can establish an event notification subscription so that when one of the specified events occurs, an
195 asynchronous Notification is sent to a specified Notification Recipient.

196 One or more Per-Job Submission subscriptions are specified by the client when submitting a job. One or
197 more Per-Job or Per-Printer subscriptions are created by performing separate explicit Create-Job-
198 Subscription or Create-Printer-Subscriptions operations, respectively on the Printer.

199 A Per-Job or Per-Printer subscription request includes:

- 200 1. the names of Job and/or Printer events that are of interest to the Notification Recipient
- 201 2. the delivery method and address to use to deliver the notification to one Notification Recipient
- 202 3. whether if Human Consumable or Machine Consumable notification content is to be sent, which
203 text format

- 204 4. some opaque data that the subscriber wants to be sent to the Notification Recipient in the
205 Notification, perhaps to identify either the subscriber or the ultimate recipient
- 206 5. the charset to use in the Notification, if it is to be different than the one used in the request that
207 created the subscription
- 208 6. the natural language to use in the Human Consumable Notification, if it is to be different than the
209 one used in the request that created the subscription
- 210 7. the requested lease time in seconds for the subscription
- 211 8. whether or not the subscription is requested to be persistent across power cycles.

212 For Per-Job subscriptions, a client requests job and printer event notification using the "job-notify"
213 operation attribute when creating a job with any of the Job Creation operation: Print-Job, Print-URI, and
214 Create-Job. The "job-notify" operation attributes may be submitted to the Validate-Job in order to be
215 validated. . The "job-notify" operation attribute contains one or more collection values, each consisting of
216 a number of member attributes that specify a subscription, so that a Job can have more than one Per-Job
217 subscription. The 'collection' is a new attribute syntax (see [ipp-coll]). The member attributes of each
218 collection value are copied to separate Subscription objects to populate the corresponding Subscription
219 Description attributes.

220

221 For Per-Printer subscriptions and Per-Job subscriptions created after the Job has been created, a client
222 requests job and printer event notification using new operations independent of any job. The Printer keeps
223 each subscription in a separate Subscription object. The Create-~~Job-Subscription~~ and ~~Create-Printer-~~
224 ~~Subscription~~ operations creates an instance of the Subscription object supplying these new operation
225 attributes and returns a subscription-id (analogous to a job-id for a Job object). These operation attributes
226 are copied to the Subscription object as Subscription Description attributes and so may be queried using the
227 Get-Subscription-Attributes and Get-Subscriptions operations. The subscriber requests a lease time for
228 each Per-Printer subscription which MAY be infinite. The Printer grants a lease time according to its
229 configured policy. A client MUST renew the Subscription before the granted lease time expires using the
230 Renew-Subscription operation.

231 Table 1 summarizes the Per-Job and Per-Printer Subscription operations and their salient input operation
232 attributes.

233

Table 1 - Summary of Per-Job and Per-Printer Subscription operations

Operation:	Per-Job	Per-Printer	salient inputs beside printer-uri:
Print-Job, Print-URI, Create-Job	yes	no	1setOf {recipient, [events,] [content-type text-format,] [user-data,] [charset,] [natural-language]}
Validate-Job	yes	no	1setOf {recipient, [events,] [content-type text-format,] [user-data,] [charset,] [natural-language]}
<u>Create-Printer-Subscription</u>	<u>no</u>	<u>yes</u>	<u>recipient, [events,] [text-format,] [user-data,] [charset,] [natural-language,] [lease-time-requested,] [persistence-requested]</u>
Create- <u>Job</u> -Subscription	yes	yes no	recipient, [job-id], [events,] [content-type text-format,] [user-data,] [charset,] [natural-language,] [lease-time-requested,] [persistence-requested]
Get-Subscription-Attributes	yes	yes	subscription-id, [requested-attributes]
Get-Subscriptions	yes	yes	[job-id], [my-subscriptions,] [requested-attributes]
Renew-Subscription	yes	yes	subscription-id, [lease time-requested]
Cancel-Subscription	yes	yes	subscription-id

234

235 There are two steps that IPP notification must take regarding each event – an internal event recording, and
 236 an external notification:

237 1) As an events occurs, the printer internally records in the job objects and the printer objects those
 238 events which are required to be supported by the system and those that are subscribed to by a
 239 notification recipient.

240 2) As an events occurs, the Printer searches the set of subscriptions for any interest in that event. As
 241 the Printer finds that some notification recipient is interested in that event (the notification recipient
 242 is subscribed to the event), the "request-id" sequence number for that event is incremented and a
 243 notification is generated and delivered using the methods and target addresses identified in the
 244 subscription. The "request-id" sequence number permits a Notification Recipient to detect duplicate
 245 notifications due either to duplicate subscriptions or retries and to detect dropped notifications.

246 2 Model for Per-Job and Per-Printer Subscription and Event Notification

247 2.1 Model for Per-Job Subscription and Notification

248 Per-Job subscriptions are created by a client (desktop or server acting as a client) as part of creation of the
 249 job in an IPP Printer (printing device or server). More than one subscription may be submitted with a job.
 250 Additional subscriptions may be associated with the job using the Create-Job-Subscription operation. The
 251 IPP Printer object delivers a Notifications to the Notification Recipient supplied by the Client in each
 252 subscription. A Notification Recipient can be the Job submitter or a third party.

253 Figure 1 shows the Per-Job subscription notification model for a simple Client - Printer relationship.


```

254
255 embedded printer:
256
257 desktop or server          output device or server
258 +-----+                  +-----+
259 | client |---IPP job submission---># Printer # |
260 +-----+ Per-Job subscription | # Object # |
261 +-----+                  | #####|##### |
262 |Notification|                  +-----+
263 |Recipient |<-----IPP Notifications-----+
264 +-----+ (Job and/or Printer events)

```

Figure 1 - Client-Printer Per-Job Subscription and Notification Model

266 Figure 2 shows a (spooling or non-spooling) Server that implements two Printer objects (1 and 2) that
 267 represent two devices. The devices A and B in turn each implement an IPP Printer object (3 and 4,
 268 respectively). The Server implementation has three choices for how to support Per-Job subscriptions to the
 269 client (and itself):

- 270 1. forward the Per-Job subscriptions to the down stream IPP Printer and let it perform the
 271 notification directly to the Notification Recipients supplied by the Client (Notifications(C)) and
 272 use Per-Printer Subscriptions for the Server's own purposes.
- 273 2. save the client-supplied Per-Job subscription on the Job object in the server and substitute its
 274 own Per-Job subscription with the Server as the Notification Recipient (Notifications(B)). Then
 275 the Server relays Notifications to the client-supplied Notification Recipients (Notifications(A)).
- 276 3. A combination of 1 and 2 in which the Server adds its own Per-Job subscriptions to those
 277 supplied by the client. Thus the IPP Job that goes to Printer object 4 has a combination of
 278 subscription information from both the Client and the Server. This latter approach is sometimes
 279 called "piggy-backing" because the Server is adding its Per-Job subscription information to that
 280 supplied by the client. Piggy-backing is especially useful, if device B also accepts (IPP or non-
 281 IPP) requests from other servers. Then when all the jobs from Server S have been completed by
 282 device B, there will be no more Job events sent to Server S. (Server S could still maintain a
 283 long term Per-Printer subscription with Printer D to that Server S can have Printer B's state track
 284 (shadow) that of Printer D or Server S could poll Printer D when queried about Printer B).

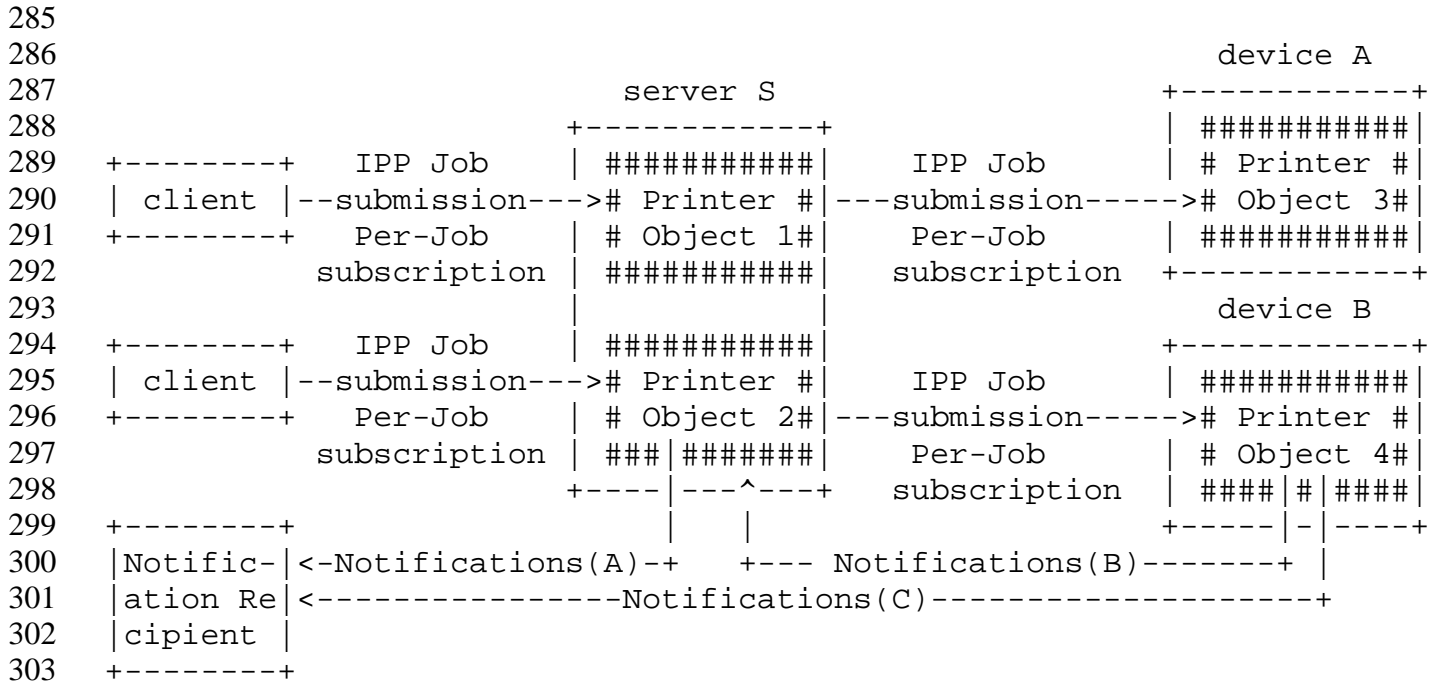


Figure 2 - Client-Server-Printer Per-Job Subscription and Notification Model

2.2 Model for Per-Printer Subscription and Notification

Per-Printer subscriptions are created by a client (an end user, an operator, or a server acting as a client) using a Create-Printer-Subscription operation that is independent of Job Submission. The Printer object (printing device or server) creates a Subscription object to hold the attributes supplied by the subscriber. The client creates separate Per-Printer subscriptions if more than one Notification Recipient is desired. The Printer delivers Notifications to the Notification Recipient specified by each Per-Printer subscription. A Notification Recipient may be the subscriber or a third party. Figure 3 shows the Per-Printer subscription notification model for the Client - Printer relationship where the client may be an end user, an operator, or a server acting as a client.

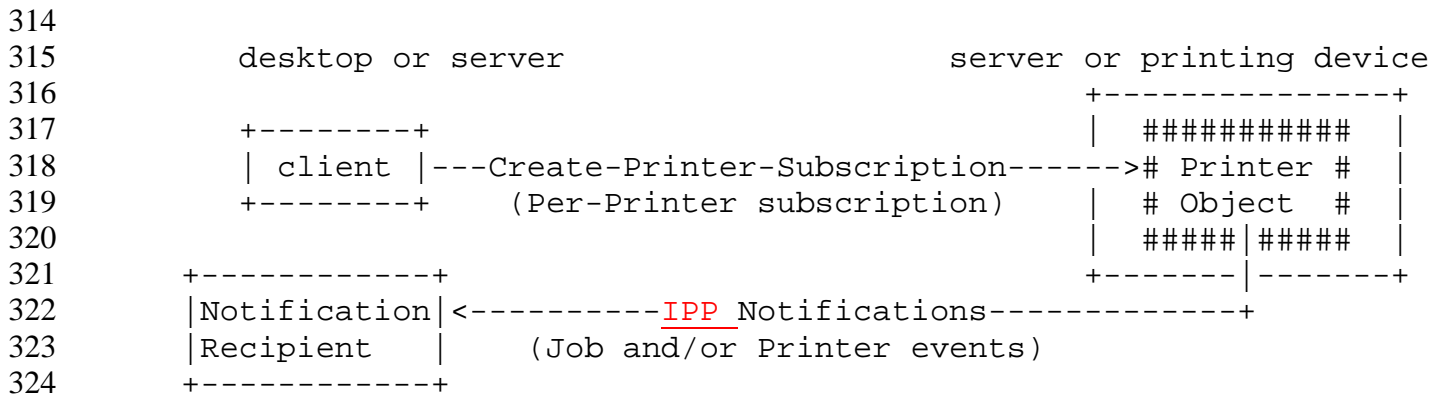


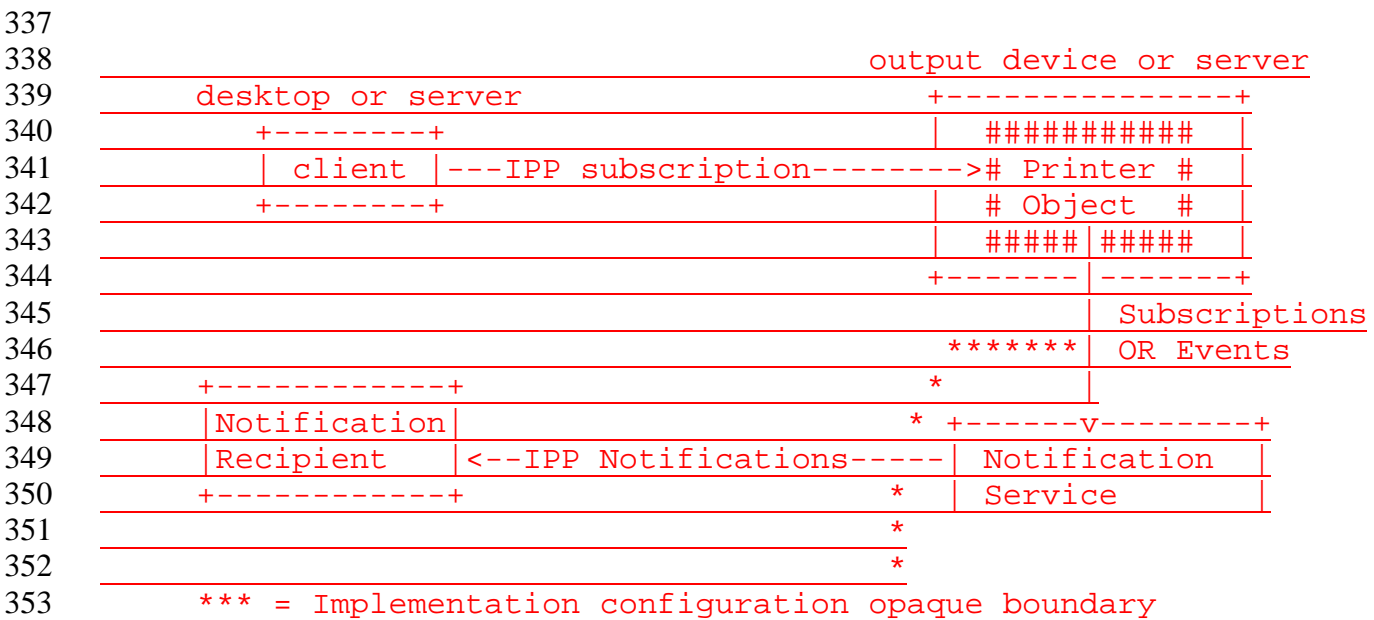
Figure 3 - Client-Printer Per-Printer Subscription and Notification Model

326 **2.3 Relationship between the Printer object and the Notification Delivery Service**

327 -The IPP Notification model does not mandate that the IPP Printer object implement the full semantics of
 328 subscription, report generation, and multiple delivery methods itself. This section describes two methods
 329 of using third party notification services. The first is transparent to the client and the second is transparent
 330 to the IPP Printer.

331 **2.3.1 Use of a Notification Service transparently to clients**

332 An implementation may be configured to use some other notification service to either (1) delivery the
 333 Notifications to the Notification Recipient(s) specified in the IPP Subscription or (2) keep the
 334 Subscriptions, accept events, possibly format the notification in the natural language of the Notification
 335 Recipient when a Human Consumable text format is used, and deliver the Notifications to the Notification
 336 Recipient(s) indicated in the IPP Subscription. Figure 4 shows this partitioning.



335 **Figure 4 - Opaque Use of a Notification Service Transparent to the Client**

336 ~~If an IPP Printer does use some other notification service, it MAY do so in either of two ways:~~

- 337 ~~1.The IPP Printer forwards the Subscriptions and the Events to the other notification service which~~
 338 ~~then sends the Notifications to the proper Recipients.~~
- 339 ~~2.The IPP Printer keeps the subscriptions and uses them to send Notifications to the other~~
 340 ~~notification service which forwards them to Recipients that have also subscribed directly with~~
 341 ~~the other notification service. Examples of this latter approach are paging and immediate~~
 342 ~~messaging services.~~

343 In any case, the interface between the IPP Printer and the other notification service is outside the scope of
 344 this document and is intended to be transparent to the client and this specification.

2.3.2 Use of Notification Service transparently to the IPP Printer

Another way that a Notification Service can be used is if the Notification Recipient indicated in the IPP Subscription is a notification service (transparent to the IPP Printer), which in turn forwards the Notification to the Ultimate Notification Recipient using additional parameters in the IPP Subscription (URI parameters or subscriber user data). In such cases, the Ultimate Notification Recipient has also subscribed directly with the other notification service (by means outside this document). As far as the IPP Printer is concerned, the IPP Subscription indicated that the IPP Printer is to delivery Notifications to the Notification Recipient (Notification Service) using the specified notification delivery method. The method that the Notification Recipient uses for delivering the notification to the Ultimate Notification Recipient is beyond the scope of this document and is transparent to the IPP Printer. However, the client does have to know how to pass additional information to the Notification Recipient in the IPP Subscription using either extra parameters in the URI or subscriber user data. Examples of this latter approach are paging, immediate messaging services, and NOS vendors infrastructure. Figure 5 shows this approach.

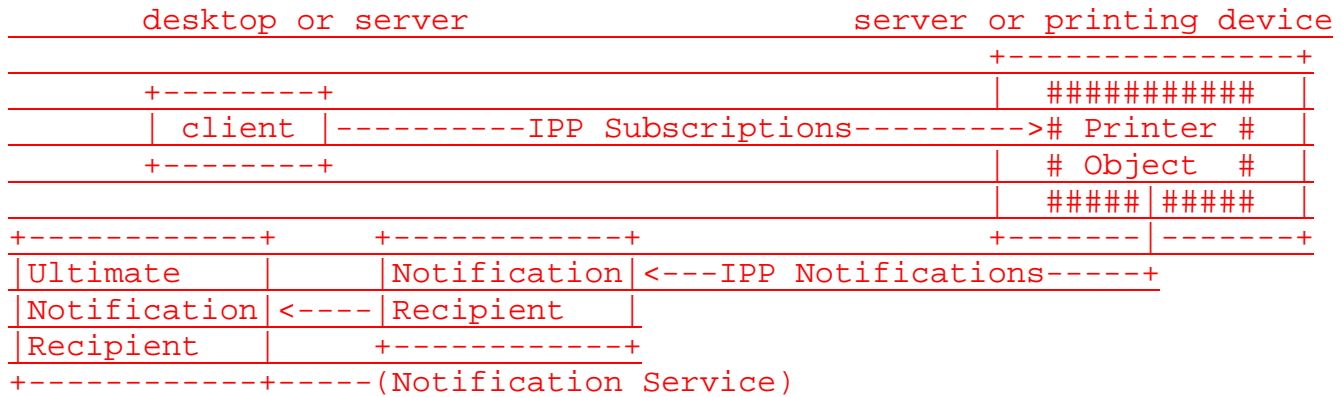


Figure 5 - Use of a Notification Service transparently to the IPP Printer

3 Terminology

This section defines terminology used throughout this document.

3.1 Conformance Terminology

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

*Since support of this entire notification specification is **OPTIONAL** for conformance to IPP/1.0 or IPP/1.1, the use of the term **REQUIRED (R)** in this document means "REQUIRED if this notification specification is implemented". Likewise, the term **RECOMMENDED** means "RECOMMENDED if this notification specification is implemented" and **OPTIONAL (O)** means **OPTIONAL if this notification specification is implemented**.*

READ-ONLY - indicates an attribute that **MUST NOT** be settable using the Set-Job-Attributes or Set-Printer-Attributes operations (see [ipp-set2]).

3.2 Other terminology

406
407 **Job Submitting End User** - A human end user who submits a print job to an IPP Printer. This person
408 may or may not be within the same security domain as the Printer. This person may or may not be
409 geographically near the printer.

410 **Administrator** - A human user who established policy for and configures the print system.

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

413 **Job Submitting Application** - An application (for example, a batch application), acting on behalf of a
414 Job Submitting End User, which submits a print job to an IPP Printer. The application may or may
415 not be within the same security domain as the Printer. This application may or may not be
416 geographically near the printer.

417 **Security Domain** - The set of network components which can communicate without going through a
418 proxy or firewall. A security domain may be geographically very large, for example - anyplace
419 within IBM.COM.

420 **IPP Client (or client)** - The software component (desktop or server) that sends an IPP operation
421 request to an IPP Printer object (server or printing device) and accepts the resulting operation
422 response from the IPP Printer object.

423 **Job Recipient** - A human who is the ultimate consumer of the print job. In many cases this will be the
424 same person as the Job Submitting End User, but need not be.

425 Example: If I use IPP to print a document on a printer in a business partner's office, I am the Job
426 Submitting End User, while the person I intend the document for in my business partner's office is
427 the Job Recipient. Since one of the goals of IPP is to be able to print near the Job Recipient of the
428 printed output, we would normally expect that person to be in the same security domain as, and
429 geographically near, the Printer. However, this may not always be the case. For example, I submit a
430 print job across the Internet to a Kinko's print shop. I am both the Submitting End User and the Job
431 Recipient, but I am neither near nor in the same security domain as the Printer.

432 **Job Recipient Proxy** - A human acting on behalf of the Job Recipient. In particular, the Job Recipient
433 Proxy physically picks up the printed document from the Printer, if the Job Recipient cannot
434 perform that function. The Proxy is **by definition** geographically near and in the same security
435 domain as the printer.

436 Example: I submit a print job from home to be printed on a printer at work. I'd like my secretary to
437 pick up the print job and put it on my desk. In this case, I am acting as both Job Submitting End
438 User and Job Recipient. My secretary is acting as a Job Recipient Proxy

439 **Notification Subscriber (or Subscriber)** - A client that requests the IPP Printer to send Event
440 Notifications to one or more Notification Recipients. A Notification Subscriber may be:

- 441 1. a Job Submitting End User or Job Submitting Application (desktop or server) that is submitting
442 a job or
- 443 2. an End User, an Operator, or an Administrator that is not submitting a job.

444 **Subscription** - A request by a Notification Subscriber to the IPP Printer to send Event Notifications to
445 a specified Notification Recipient when the event occur. A Subscription is represented as a set of
446 attributes that indicate the "what, where, who, and how" for notification. Notifications are
447 generated for certain events (what) and delivered using various delivery methods (how) to certain
448 addresses (where and who).

449 **Per-Job Subscription** - A Subscription that a client specifies as part of a create job operation (Print-
450 Job, Print-URI, Create-Job), a Validate-Job operation, or an explicit Create-Job-Subscription
451 operation with a Job object as the target.

452 **Per-Printer Subscription** - A Subscription that a client specifies using an explicit Create-Printer-
453 Subscription operation with a Printer object as the target.

454 **Notification Source** - The entity that sends Event Notifications. It MAY be the IPP Printer itself or the
455 IPP Printer MAY be configured to use a Notification Service to delivery Notifications transparently
456 to the subscribing clients (see Figure 4).

457 **Notification Recipient** - The entity identified as a recipient within a subscription that receives IPP
458 Notifications about Job and/or Printer events (see Figure 4 and Figure 5). A Notification Recipient
459 may be a: Job Submitting End User, -Job Submitting Application (desktop or server), Job Recipient,
460 Job Recipient Proxy, a Notification Service, an Operator, or Administrator, etc., and their
461 representative or log file or usage statistics gathering application or other active or passive entities.

462 **Ultimate Notification Recipient** - The entity to which the Notification Recipient (stores and) forwards
463 an IPP Notification when the Notification Recipient is a Notification Service (see Figure 5).

464 ~~**Notification Recipient Agent** - A program which receives Notifications on behalf of the Notification~~
465 ~~Recipient. The agent may take some action on behalf of the recipient, forward the notification to the~~
466 ~~recipient via some alternative means (for example, page the recipient), or queue the notification for~~
467 ~~later retrieval by the recipient.~~

468 **Event** - An event is some occurrence (either expected or unexpected) within the printing system of a
469 change of state, condition, or configuration of a Job or Printer object. A property of an event is that
470 it only occurs at one instant in time and does not span the time the physical event takes place. For
471 instance, jam-occurred and jam-cleared are two distinct events. The jam-occurred event is reported
472 only when the jam initially occurs and only if there is one or more event subscriptions outstanding
473 for that event.

474
475 Events can be classified along two dimensions:

- 476 - Either as Job Events or Printer Events, and
- 477 - Either as Errors, Warnings, or Reports

478
479 A Job event is some interesting state change in the Job object, and a Printer event is some
480 interesting change in the Printer object.

481
482 A report event is purely informational, such as 'job-completed' or 'accepting-jobs'. A warning is not
483 serious and processing continues. An error is serious and either the job is aborted or the printer
484 stops. These are typical uses of the terms report, warning, and error, although the actual usage is
485 implementation dependent.

486
487 An event occurs for a job or printer whether any entity has subscribed to be notified for that event or
488 not. A notification is only generated depending on the set of subscriptions outstanding.

489
490 **Notification** - When an event occurs, a Notification is generated that fully describes the event (what
491 the event was, where it occurred, when it occurred, etc.). Notifications are delivered to each
492 Notification Recipient that has a subscription that includes the event, if any. The Notification is
493 delivered to the address of the Notification Recipient using the notification delivery method defined
494 in the subscription. However, a Notification is sent ONLY if there is a corresponding subscription.

495 **Notification Delivery Method (or Delivery Method for short)** - Notifications are delivered using a
496 method, such as email, TCP/IP, etc.

497 **Immediate Notification** - Notifications that are delivered using a delivery method which is not store-
498 and-forward (e.g. TCP connection, UDP datagram). This can be on the order of several minutes
499 subject to network latency.

500 **Store and Forward Notification** - A Notification which are not necessarily delivered to Notification
501 Recipients immediately, but is queued for delivery by some intermediate network application, for
502 later retrieval. Email and Instant Messaging services are is-an-examples of a store and forward
503 notification delivery method.

504 **Human Consumable Notification** - Notifications that are intended to be consumed by human End
505 Users only. They are simple text that has been localized for the Notification Recipient as specified
506 in the subscription. Programs are not intended to parse Human Consumable Notification, since it is
507 localized and the content depends on implementation. There is no standardized format.

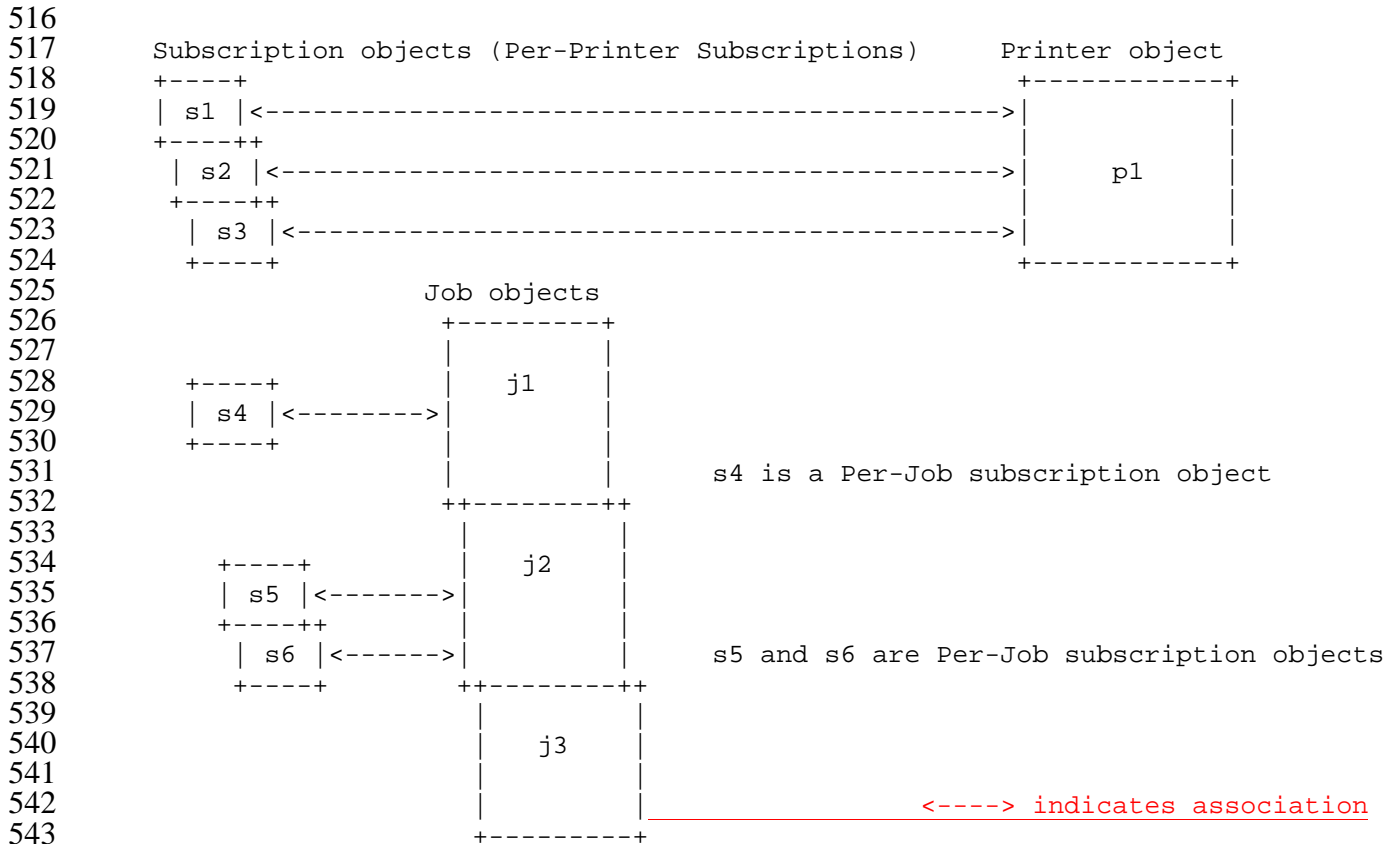
508 **Machine Consumable Notification** - Notifications that are intended for consumption by a program
509 only. They use the encoding of an IPP response. The Notification Recipient must localize the
510 contents, if displaying it to a human.

511 4 Object Model for Notification

512 This section describes the notification object model that adds a REQUIRED Subscription object which
513 together with the Job and Printer object provide the complete notification semantics.

514

515 The object relationships can be seen pictorially as:



544 **Figure 6 - Object Model for Notification**

545 s1, s2, and s3 are Per-Printer Subscription objects and can identify Printer and/or Job events.

546 s4, s5, and s6 are Per-Job subscription objects and can identify Printer and/or Job events.

547 4.1 Object relationships

548 The object relationships can be stated as follows:

- 549 1. The Printer object contains zero or more Per-Printer Subscription objects (p1 contains s1-s3 Per-Printer Subscription objects).
- 550
- 551 2. Each Per-Printer Subscription object (s1, s2, and s3) is contained in one Printer object (p1) and each
- 552 represents one Per-Printer subscription.
- 553 3. Each "Per-Printer" Subscription object identifies one or more Job and/or Printer events. Such Job
- 554 events are for *all* jobs on the Printer. Such Printer events are for *any* Printer event, no matter which
- 555 job is processing and when no jobs are processing.
- 556 4. A Job object is associated with zero or more Per-Job subscription objects. Job j1 is associated with
- 557 Per-Job subscription object s4, Job j2 is associated with Per-Job subscription objects s5 and s6, and
- 558 Job j3 ~~does~~ is not associated with any Per-Job subscription object. Note: the IPP notification
- 559 interface semantics are defined so that an implementation MAY associate Per-Job Subscription
- 560 objects with Job objects by having the Job objects actually contain its associated Per-Job

561 Subscription objects or MAY just use some internal bi-directional linking mechanism between the
562 Job and Subscription objects. Either implementation technique is transparent to the client.

563 5. Each "Per-Job" subscription object identifies one or more Job and/or Printer events. Such Job
564 events are only for *this* job (different than "per-Printer" Subscriptions). Such Printer events are for
565 *any* Printer event, no matter which job and when no jobs are processing (same as for "per-Printer"
566 Subscriptions).

567 6. A Per-Printer Subscription object cannot be contained in or associated with more than one Printer
568 object.

569 7. A Per-Job Subscription object cannot be contained in or associated with more than one Job object.

570 **5 Subscription Object attributes**

571 The following notification attributes are defined for the Subscription object. The definitions of the object
572 attributes are specified here so that they can be referred to from the subsequence definitions of the
573 operations that set them.

574

Table 2 - Subscription object attributes

Subscription object attributes:	Printer support	READ-ONLY, set by:
notify-recipient (uri)	REQUI RED	client - Job Creation, Create-Job-Subscription , and Create- Printer-Subscription
notify-events (1setOf type2 keyword)	REQUI RED	client - Job Creation, Create-Job-Subscription , and Create- Printer-Subscription
notify- content-type text-format (mimeMediaType)	REQUI RED	client - Job Creation, Create-Job-Subscription , and Create- Printer-Subscription
subscriber-user-data (octetString(63))	REQUI RED	client - Job Creation, Create-Job-Subscription , and Create- Printer-Subscription
notify-charset (charset)	OOPTI ONAL	client - Job Creation, Create-Job-Subscription , and Create- Printer-Subscription
notify-natural-languages (1setOf naturalLanguage)	OOPTI ONAL	client - Job Creation, Create-Job-Subscription , and Create- Printer-Subscription

575

request-id (integer(0:MAX))	REQUI RED	initialized to 0, incremented by Printer - beginning of each event
-----------------------------	----------------------	--

576

subscription-id (integer(1:MAX))	REQUI RED	Printer - Job Creation, Create-Job-Subscription , and Create- Printer-Subscription
notify-lease-expiration-time (integer(0:MAX))	REQUI RED	Printer - Job Creation (set to 0) and Create- Printer-Subscription , Renew-Subscription
printer-uri (uri)	REQUI RED	Printer - Job Creation, Create-Job-Subscription , and Create- Printer-Subscription
subscriber-user-name (name(MAX))	ROPTI ONAL	Printer - Job Creation, Create-Job-Subscription , and Create- Printer-Subscription

577

notify-printer-up-time (integer(1:MAX))	REQUI RED	Printer - returned by Create-Printer-Subscription , Get-Subscription-
---	----------------------	---

		Attributes, and Get-Subscriptions
<u>notify-persistence-granted (boolean)</u>	<u>REQUIRED</u>	<u>Printer - returned by Create-Job-Subscription and Create-Printer-Subscription</u>

578 Note: The Subscription object does not contain the "job-id" Subscription Description attribute. The Get-
579 Subscriptions operation has the "job-id" as an input operation attribute, so the "job-id" isn't returned in the
580 response. If an implementation needs such a link between Subscription objects and Job objects, then it
581 keeps such a link as in internal attribute. The intent is that whether Per-Job Subscription objects are
582 actually contained in a Job object or are just associated with them in some way is IMPLEMENTATION
583 DEPENDENT and is transparent to the client.

584 5.1 notify-recipient (uri)

585 This REQUIRED READ-ONLY Subscription object attribute describes both where (the address of the
586 Notification Recipient) and how (the delivery method) notifications are to be delivered to the Notification
587 Recipient when any of the events specified in the "notify-events" attribute occur.

588 There are potentially many different notification delivery methods for IPP notifications, standardized as
589 well as proprietary. This document does not define any of these delivery mechanisms; they will each be
590 described in separate complementary documents.

591 Each of the notification delivery method documents must provide at least the following information:

- 592 1) The URI scheme used.
- 593 2) The supported and default delivery format, and if not one of the specified types in Section 5.3,
594 description of the notification content.
- 595 3) Any content length restrictions imposed by the delivery protocol.
- 596 4) The latency of the delivery protocol used.
- 597 5) The reliability of the transport and delivery protocol used.
- 598 6) The security aspects of the transport and delivery protocol used, e.g. how it is handled in firewalls.
- 599 7) How the delivery protocol is initiated, e.g. does it have to be initiated by the receiving user (pull), or
600 is it initiated by the notification service (push).

601 **ISSUE 1** - Once a number of delivery solutions have been developed and evaluated, we may want to make
602 one or several of them REQUIRED for implementation to ensure a minimum set of interoperability. Which
603 one or ones should be REQUIRED?

604 5.2 notify-events (1setOf type2 keyword)

605 This REQUIRED READ-ONLY Subscription object attribute identifies the job and/or printer events that
606 are to be delivered to the Notification Recipient as Notifications as defined in section 7. If the client did
607 not supply this attribute when supplying the subscription, the Printer object populates this attribute with its
608 "notify-events-default" attribute value (see section 6.2).

609 There are both job events and printer events. Each job and printer event is assigned a keyword to use in
610 this attribute and in the Notification.

611 The events are defined to be disjoint.

612 A Printer MUST support the events indicated as "REQUIRED".

613 The standard job event keyword values are:

614 **'none'**: REQUIRED - no notifications of any events (an IPP object can use this value to indicate that it
615 is configured not to support event notification; a client would not subscribe to this event).

616 **'job-created'**: REQUIRED - the Printer object has accepted a job creation operation (Print-Job, Print-
617 URI, or Create-Job) and the job's "time-at-creation" attribute value is set (see [ipp-mod] section
618 4.3.14.1). The Printer puts the job in the 'pending', 'pending-held' or 'processing' states.

619 Note: This event is separate from the 'job-state-changed' event so that it can be subscribed to
620 without having to get every job state change event for a Notification Recipient that is only
621 interested in when the job is first created.

622 **'job-completed'**: REQUIRED - the job has reached one of the completed states, i.e., the value of the
623 job's "job-state" attribute has changed to: 'completed', 'aborted', or 'canceled'. The Job's "time-at-
624 completed" and "date-time-at-completed" (if supported) attributes are set (see [ipp-mod] section
625 4.3.14).

626 Note: This event is separate from 'job-state-changed' so that it can be subscribed to without having
627 to get every job state change event for a Notification Recipient that is only interested in when the
628 job is completed.

629 **'job-state-changed'**: REQUIRED - the job has changed from any state to any other state and/or a
630 value has been added or removed from the job's "job-state-reasons" attribute, except when the job is
631 created or when the job moves to any of the "completed" job states ('completed', 'aborted', or
632 'canceled').

633
634 This event also indicates that one or more values have been added to or removed from the Job's
635 "job-state-reasons" attribute, such as 'job-queued' or 'job-printing', whether or not the job's state has
636 changed. If job state reasons are added when the job is created, only the 'job-created' event is
637 generated, in order to keep the events disjoint. If job state reasons are added or removed when the
638 job is completed, only the 'job-completed' event is generated, in order to keep the events disjoint.

639
640 A client that wants to subscribe to all job state changes, including creation and completion, includes
641 the 'job-created', 'job-state-changed', and 'job-completed' in the notification subscription. When a
642 job is finally removed from the Job History (see [ipp-mod] 4.3.7.1) no event is generated, i.e.,
643 neither a 'job-state-changed' event nor a 'job-purged' event is generated.

644 **'job-config-changed'**: OPTIONAL – when the configuration of a job has changed, i.e., the value of
645 the "job-message-from-operator" or any of the non-READ-ONLY Job attributes have changed, such
646 as any of the job template attributes or the "job-name" attribute. Typically, such a change is the
647 result of the user or the operator performing a Set-Job-Attributes operation (see [ipp-set2]) on the
648 Job object. The client performs a Get-Job-Attributes to find out the new values of the changed
649 attributes. This event is useful for GUI clients and drivers to update the job information to the user.

650 **'job-purged'**: OPTIONAL - when a 'not-completed' job (i.e., not 'completed', 'canceled', or 'aborted')
651 was purged from the printer using the Purge-Jobs operation. No event, including this event, is
652 generated when a job is aged out of the Job History or moved out explicitly with the Purge-Jobs
653 operation.

654 **'job-progress'** - a sheet or copy has completed. See separate [ipp-prog] spec.

655

656 The standard Printer event keywords values are:

657 **'none'**: REQUIRED - no notification of any events (an IPP object can use this value to indicate that it
658 is configured not to support event notification; a client would not subscribe to this event).

659 **'printer-restarted'**: OPTIONAL - when the printer is powered up or the Restart-Printer operation is
660 performed (see [ipp-set2]).

661 Note: This event is separate from the 'printer-state-changed' event so that it can be subscribed to
662 without having to get every printer state change event, for a Notification Recipient that is only
663 interested in when the Printer first comes up.

664 **'printer-shutdown'**: OPTIONAL - when the device is being powered down or the Shutdown-Printer
665 operation has been performed with either power-off or standby options (see [ipp-set2]).

666 Note: This event is separate from 'printer-state-changed' so that it can be subscribed to without
667 having to get every Printer state change event, for a Notification Recipient that is only interested in
668 when the Printer is powered down or shutdown.

669 **'printer-state-changed'**: REQUIRED - the Printer changed state, i.e., the value of the Printer's
670 "printer-state", "printer-state-reasons" (whether "printer-state" changed or not), and/or "printer-is-
671 accepting-jobs" attributes changed, except when the Printer starts up or is shutdown. If printer state
672 reasons are added when the Printer is started up, only the 'printer-restarted' event is generated, in
673 order to keep the events disjoint. If printer state reasons are added or removed when the printer is
674 powered-down or shutdown, only the 'printer-shutdown' event is generated, in order to keep the
675 events disjoint.

676
677 A client that wants to subscribe to all printer state changes, including restart and power-
678 down/shutdown, includes the 'printer-restarted', 'printer-state-changed', and 'printer-shutdown' in the
679 notification subscription.

680 **'printer-media-changed'**: OPTIONAL – when the media loaded on a printer has been changed, i.e.,
681 the "media-ready" attribute has changed. This event includes both an actual media change and
682 filling an empty input tray with the same or different media. The client must check the "media-
683 ready" Printer attribute (see [ipp-mod] section 4.2.11) separately to find out what new media was
684 loaded or filled.

685 **'printer-config-changed'**: OPTIONAL – when the configuration of a Printer has changed, i.e., the
686 value of the "printer-message-from-operator" or any non-READ-ONLY Printer attribute has
687 changed, except for "media-ready" (which has its own event), whether through the Set-Printer-
688 Attributes operation or by other means and whether initiated by a human or not. For example, any
689 "xxx-supported", "xxx-default", "printer-message-from-operator", etc. values have changed. The
690 client has to perform a Get-Printer-Attributes to find out the new values of these changed attributes.
691 This event is useful for GUI clients and drivers to update the available printer capabilities to the
692 user.

693 **'printer-queue-changed'**: OPTIONAL - the order of jobs in the Printer's queue has changed, so that an
694 application that is monitoring the queue can perform a Get-Jobs operation to determine the new
695 order. This event does not include when a job enters the queue (the 'job-created' event covers that)
696 and does not include when a job leaves the queue (the 'job-completed' event covers that).

697 **'printer-no-longer-full'**: OPTIONAL – when the Printer can now accept a Print-Job, Print-URI,
698 Create-Job, Send-Document, or Send-URI request. This event is used when there is more than one
699 client feeding a printer/server (fan-in), and the Printer may still be printing but has acquired more
700 buffer space to accept jobs. This event only occurs when the Printer did not have room to accept

701 jobs previously and rejected a Print-Job, Print-URI, Create-Job, Send-Document, or Send-URI
702 operation.

703 '**printer-almost-idle**': OPTIONAL – when the Printer needs another Job in order to stay busy. This
704 event is used when a spooler is feeding more than one printer/server (fan-out), and the spooler holds
705 jobs until a Printer requests them, rather than committing jobs to IPP Printers before it is necessary.
706 This event MAY be used by a Printer implementation to request a new job from any subscribers
707 sufficiently ahead of time so that the device does not run out of work between jobs.
708

709 **5.3 ~~notify-content-type~~text-format (mimeMediaType)**

710 This REQUIRED READ-ONLY Subscription object attribute ~~specifies~~ indicates the type of Human
711 Consumable format content that is to be sent in the notifications, instead of the Machine Consumable
712 format defined for the notification scheme, if any. ~~Thus the subscriber can control whether the event~~
713 ~~notification content is Human Consumable or Machine Consumable.~~ Most delivery methods are defined to
714 have support both forms a particular Machine Consumable forms of notification content type and to permit
715 Human Consumable forms as well and MUST define one of them as the default. An implementation MAY
716 support one or more Human Consumable formats, i.e., 'text' MIME media types, for those delivery methods
717 that permit the Human Consumable form. ~~If a delivery method only supports one form of content type,~~
718 ~~then that form is also the default.~~

719 If the 'text' MIME media type registration permits a charset parameter, than such a specification MUST be
720 used (instead of the "notify-~~attributes~~-charset" attribute) in order to indicate the charset to be used in the
721 notification content.

722 If the Subscriber did not supply this attribute when requesting the subscription, the Printer object populates
723 this Subscription object attribute ~~from the default value defined for the delivery method~~ with either the
724 'none' value or one of the values of the Printer's "notify-text-format-supported" attribute (see section 6.5),
725 depending on whether or not the delivery method specified in the "notify-recipient" attribute is defined to
726 have a Machine Consumable format (usual), respectively. In the latter case, the value selected depends on
727 the implementation.

728 Standard mimeMediaType values are:

729 ~~'application/ipp' the Machine Consumable notification content using the 'application/ipp' MIME~~
730 ~~media type [ipp-mod] as defined in section 7.~~

731 'none': Indicates that the notification content is not to be any of the text types, i.e., that the Machine
732 Consumable, not the Human Consumable, form is to be sent. If the client omits supplying this
733 attribute, the meaning is the same as if the client supplied the 'none' value.

734
735 'text/plain; charset=utf-8': A plain text document in ISO 10646 represented as UTF-8 [RFC-
736 2044~~2279~~] as defined in section 7.

737
738 'text/html': An HTML document [rfc????].

739 ~~ISSUE 2 Should notify-content-type (mimeMediaType) be changed to notify-content-type (type2~~
740 ~~keyword), with values: 'human-consumable' and 'machine-readable'? Then content types that are neither~~
741 ~~'application/ipp' or 'text/plain; charset=utf-8' could be handled without having to be registered. For~~
742 ~~example, the 'snmp-ipp' is a trap MIB Machine Consumable format wouldn't not need to be registered (or~~

743 ~~have some special rule since the 'snmp-ipp' value doesn't have a registered MIME type and doesn't fit either~~
744 ~~'application/ipp' or 'text/plain'. Another advantage is that the charset would always be indicated in the~~
745 ~~"notify-attributes-charset" operation attribute, even for plain text.~~

746 ~~ISSUE 3—There is no good way for a client to discover which "notify-content-type" values are supported~~
747 ~~for each delivery method "notify-schemes-supported". Adding "notify-content-type-supported" doesn't~~
748 ~~work, since the delivery methods have to be paired with the content types. Instead, how about only having~~
749 ~~each scheme support one or the other content type? Then we don't need "notify-content-type" as all. For~~
750 ~~the 'mailto' scheme, we specify it is human consumable, i.e., text, and then register a 'mailto-ipp' scheme~~
751 ~~which is the 'application/ipp' scheme.~~

752 **5.4 subscriber-user-data (octetString(63))**

753 This REQUIRED READ-ONLY Subscription object attribute holds opaque information being sent from
754 the Subscriber to the Notification Recipient, such as the identify of the Subscriber or a path or index to
755 some Subscriber information. Or it MAY contain a key that the Notification Recipient needs in order to
756 process the Notification, such as the ultimate recipient, if the Notification Recipient is a general application
757 that in turn forwards notifications and the ultimate recipient isn't included in the value of the "notify-
758 recipient" attribute. An Instant Messaging Service is an example of such a general application where the
759 "subscriber-user-data" might be the user's id for that messaging service and the "notification-recipient" is
760 the URL of the messaging service.

761 **5.5 notify-attributes-charset (charset)**

762 This OPTIONAL READ-ONLY Subscription object attribute specifies the charset to be used in the
763 Notification content sent to the Notification Recipient, whether the notification content is Machine
764 Consumable or Human Consumable. This attribute MUST NOT be used when the "notify-~~content-~~
765 ~~typetext-format~~" attribute value specifies the charset parameter in its MIME media type value, e.g.,
766 'text/plain; charset=utf-8'.

767 ~~ISSUE 4—Should the Subscription object attributes "notify-attributes-charset (charset)" and "notify-~~
768 ~~attributes-natural-languages (naturalLanguage)" be renamed to simply "attributes-charset (charset)" and~~
769 ~~"attributes-natural-languages (naturalLanguage)" so that all IPP requests and objects have the same names?~~
770 ~~Then the "notify-attributes-charset (charset)" and "notify-attributes-natural-languages (naturalLanguage)"~~
771 ~~would only be operation attributes that set or reflect the "attributes-charset (charset)" and "attributes-~~
772 ~~natural-languages (naturalLanguage)" Subscription object attributes.~~

773 **5.6 notify-attributes-natural-languages (naturalLanguage)**

774 This OPTIONAL READ-ONLY Subscription object attribute specifies the natural language for the IPP
775 object to use in the Notification content that is sent to the Notification Recipient, whether the notification
776 content is Machine Consumable or Human Consumable.

777 **5.7 request-id**

778 This REQUIRED READ-ONLY Subscription object attribute holds the most recent request-id sequence
779 number delivered in a Notification content to the Notification Recipient. A value of 0 indicates that no
780 Notifications have been sent for this subscription. The first request-id sent for a subscription MUST be 1.

781 Each Notification Recipient has its own monotonically increasing series of request-ids, i.e., no gaps, in
782 order to be able to detect a missing notification.

783 **5.8 subscription-id (integer (1:MAX))**

784 This REQUIRED READ-ONLY Subscription object attribute uniquely identifies this Subscription object
785 instance on this Printer object or this Job object. The Printer object, on acceptance of a Create-Job-
786 Subscription or Create-Printer-Subscription request, generates an ID which identifies the new Subscription
787 object on that Printer or Job. The Printer returns the value of the "subscription-id" attribute as part of the
788 response to a Create-Job-Subscription or Create-Printer-Subscription request. The 0 value is not included
789 to allow for compatibility with "job-id" and with SNMP index values which also cannot be 0.

790 It is RECOMMENDED that Per-Printer Subscription objects be persistent. Then the Subscription objects
791 including the subscription-id remains unique across power-cycles. Even if an implementation does not
792 make Per-Printer subscription objects persist, the implementation SHOULD make every attempt not to re-
793 use subscription ids that subscribers might still think are valid. In other words, the Printer SHOULD at
794 least keep the next subscription-id to be assigned in non-volatile memory. Note: it is assumed that Per-Job
795 subscriptions are persistent if Jobs are persistent, in order to be consistent with the persistency of Job
796 objects. The [ipp-mod] RECOMMENDS that Job objects be persistent.

797 **5.9 notify-lease-expiration-time (integer(0:MAX))**

798 This REQUIRED READ-ONLY Subscription object attribute specifies the time in the future when the
799 subscription lease will expire, i.e., the "printer-up-time" value at which the lease will expire. When the
800 Printer object creates a Per-Printer Subscription object, it populates this attribute with the appropriate value.
801 When the indicated time arrives, the Printer MUST delete the Per-Printer Subscription object. Per-Job
802 Subscription objects always return a value of 0 since Per-Job Subscriptions don't have a lease, but exist for
803 the life-time of the Job instead.

804 A client is able to extend a lease of a Per-Printer subscription using the Renew-Subscription operation (see
805 section 8.3.3). A value of 0 indicates an infinite time, if such a policy is supported as indicated in the
806 "notify-lease-time-supported" (integer(0:MAX)) Printer Description attribute (see section 6.8) and the
807 subscriber is authorized to request an infinite lease. A Per-Job subscription cannot be renewed.

808 Note: In order to compute the number of seconds remaining in a Per-Printer Subscription lease, a client can
809 subtract the "notify-printer-up-time" Subscription object attribute (see section 5.12) from the "notify-lease-
810 expiration-time" Subscription object attribute.

811 **5.10 printer-uri (uri)**

812 This REQUIRED READ-ONLY Subscription object attribute identifies the Printer object that created this
813 Subscription object. When a Printer object creates a Subscription object, it populates this attribute with the
814 Printer object URI that was used in the create request. This attribute permits a client to identify the Printer
815 object URI that was used to create this Subscription object, i.e., what security scheme was used.

816 **5.11 subscriber-user-name (name(MAX))**

817 This ~~REQUIRED~~ OPTIONAL READ-ONLY Subscription object attribute contains the name of the user
818 that created the Subscription object. The Printer object sets this attribute to the most authenticated printable

819 name that it can obtain from the authentication service over which the IPP operation was received. This
820 attribute is intended to help a human user determine for which Per-Printer Subscriptions they are the
821 Subscriber. Only if such is not available, does the Printer object use the value supplied by the client in the
822 "requesting-user-name" operation attribute of the create operation (see [IPP-MOD] Sections 4.4.2, 4.4.3,
823 and 8). For Per-Job subscriptions created as part of the Job creation operation, the value of the "subscriber-
824 user-name" is the same as the "job-originating-user-name" Job attribute (see [ipp-mod] section 4.3.6).

825 The value of the "subscriber-user-name" is implementation dependent when a server accepts a request and
826 forwards it to a downstream IPP Printer (see Figure 2 and the [ipp-iig]).

827 Note: The Printer object needs to keep an internal originating user id of some form, typically as a
828 credential of a principal, with the Subscription object. Since such an internal attribute is implementation-
829 dependent and not of interest to clients, it is not specified as a Subscription Description attribute. This
830 originating user id is used for authorization checks (if any) on all subsequent operations.

831 **5.12 notify-printer-up-time (integer(1:MAX))**

832 This REQUIRED READ-ONLY Subscription object attribute indicates the amount of time (in seconds)
833 that the Printer implementation has been up and running. This attribute is an alias for the Printer's "printer-
834 up-time" attribute" (see [ipp-mod] section 4.4.29), in an analogous way that the Job's "job-printer-up-time"
835 is an alias for "printer-up-time" (see [ipp-mod] section 4.3.13.4).

836 Note: A client can request this attribute in a Get-Subscription-Attributes or Get-Subscriptions request and
837 use the value returned in combination with the "notify-lease-expiration-time" (see section 5.9) in order to
838 display wall clock time equivalent to the user. The difference between this attribute and the 'integer' value
839 of the "notify-lease-expiration-time" attribute is the number of seconds in the future that the subscription
840 will expire. A client can compute the wall-clock time at which the subscription will expire by adding this
841 difference to the client's wall-clock time.

842 **5.13 notify-persistence-granted (boolean)**

843 This REQUIRED Subscription object attribute whether or not the Per-Job or Per-Printer Subscription is
844 persistent, i.e., saved across power cycles in an implementation-define manner.

845 **6 Printer Description Attributes related to Notification**

846 This section defines the Printer Description attributes that are related to Notification. Table 3 lists the
847 Printer Description attributes and indicates the Printer support required for conformance: "R" indicates
848 REQUIRED, "O" indicates OPTIONAL, and "CR" indicates CONDITIONALLY REQUIRED, i.e.,
849 required if Human Consumable notification formats are supported.

850

Table 3 - Printer Description attributes associated with Notification

Printer object attributes:	Printer support	READ-ONLY, set by:
notify-schemes-supported (1setOf uriScheme)	R	Administrator/implementation
notify-events-default (1setOf type2 keyword)	R	Administrator/implementation
notify-events-supported (1setOf type2 keyword)	R	Administrator/implementation
max-events-supported (integer(5:MAX))	R	Administrator/implementation
<u>notify-text-format-supported (1setOf mimeType)</u>	<u>CR</u>	<u>Administrator/implementation</u>
max-job-subscriptions-supported (integer(1:MAX))	R	Administrator/implementation
max-printer-subscriptions-supported (integer(0:MAX))	R	Administrator/implementation
notify-lease-time-supported (rangeOfInteger(0:MAX))	R	Administrator/implementation
notify-lease-time-default (integer(0:MAX))	R	Administrator/implementation
persistent-jobs-supported (boolean)	O	Administrator/implementation
persistent-subscriptions-supported (boolean)	O	Administrator/implementation
printer-state-change-time (integer(1:MAX))	O	Administrator/implementation
printer-state-change-date-time (dateTime)	O	Administrator/implementation

851 ~~ISSUE 5—Shouldn't we add a "number of printer subscriptions" Printer Description attribute, since the~~
852 ~~client is NOT assured that it can determine the number of outstanding Per Printer Subscriptions by doing~~
853 ~~Get Subscriptions because of access control?~~

854 ~~ISSUE 5a—Shouldn't we add a "number of job subscriptions" Job Description attribute, since the client is~~
855 ~~NOT assured that it can determine the number of outstanding Per Job Subscriptions by doing Get~~
856 ~~Subscriptions because of access control and since there are no Job Descriptions attributes associated with~~
857 ~~notification.~~

858 **6.1 notify-schemes-supported (1setOf uriScheme)**

859 This REQUIRED Printer attribute describes the notification delivery methods supported by this Printer
860 object. Standard values are defined in Section 5.1).

861 **6.2 notify-events-default (1setOf type2 keyword)**

862 This REQUIRED Printer attribute identifies the event values if the client does not supply the "notify-
863 events" operation attribute in either a Job creation request, Create-Job-Subscription, or the Create-Printer-
864 Subscription request. Any value in this attribute MUST also appear in the notify-events-supported
865 attribute, i.e., be a supported event.

866 **6.3 notify-events-supported (1setOf type2 keyword)**

867 This REQUIRED Printer attribute identifies the events supported by this Printer object for both Per-Job and
868 Per-Printer subscriptions which MUST be the same. Standard values are defined in Section 5.2.

869 **6.4 max-events-supported (integer(5:MAX))**

870 This REQUIRED Printer attribute specifies the maximum number of events that are supported in a single
871 Per-Job or Per-Printer subscription which must be the same. A Printer MUST support at least 5 events per
872 subscription, so that clients can depend on at least 5 events in a single subscription. If the number of events
873 supplied by a client in a subscription exceed this number, the Printer rejects the request and returns the
874 'server-error-too-many-events' (see section 14.4). If notification is not supported, this attribute MUST NOT
875 be supported.

876 **6.5 notify-text-format-supported (1setOf mimeType)**

877 This CONDITIONALLY REQUIRED Printer attribute identifies the MIME media types supported for
878 Human Consumable notification content for those notification delivery methods that support Human
879 Consumable format (most do). If an implementation supports any Human Consumable formats, it MUST
880 support this attribute and MUST support those Human Consumable formats for all notification delivery
881 methods that permit a Human Consumable format. If the implementation does not support any Human
882 Consumable MIME media types, then the 'none' MUST be the only value of this attribute or this attribute
883 MUST NOT be supported at all.

884 A client can determine what formats are supported for a notify-scheme as follows. The implementation
885 contains a hard coding of the Machine Consumable format (obtained from the specification). If the hard
886 coding (obtained from the specification) prohibits a Human Consumable format for the notify-scheme, or
887 the printer doesn't support the "notify-text-format-supported" attribute, there are no Human Consumable
888 formats for any notify-scheme. Otherwise, the attribute "notify-text-formats-supported" specifies the
889 supported mime types for those delivery schemes that are supported and permit a Human Consumable
890 format.

891 Note: The rationale for the "notify-text-format-supported" attribute is that the Machine Consumable format
892 seems easy to pick for each notify-scheme and thus easy to document. It is easy for a printer to support
893 most Machine Consumable formats. It is much harder to support a Human Consumable format because of
894 localization issues. Once the code is written to support a particular Human Consumable format, it is easy
895 to transmit it on any of the supported notify-schemes. Thus, if a vendor decides to support a notify-scheme,
896 it has already committed to implement the Machine Consumable format. This may be simple if existing
897 code can be reused, e.g. application/ipp and or more difficult if new code must be written, e.g. it is SNMP.

898 **6.6 max-job-subscriptions-supported (integer(1:MAX))**

899 This REQUIRED Printer attribute specifies the maximum number of Per-Job subscriptions that are
900 supported in a single Job Creation request for a job, i.e., the maximum number of collection values for the
901 "job-notify" operation attribute, and/or the maximum number of subsequent Create-~~Job~~-Subscription
902 operation requests in combination for a job-adding Per Job subscriptions to the job object. A value of 0
903 indicates no effective maximum. A Printer MUST support at least 1 Per-Job subscription. If the number of
904 Per-Job subscriptions supplied by a client in a Job Creation request exceeds the value of this attribute or
905 would exceed some implementation-defined total number of Per-Job Subscriptions for the Printer, the
906 Printer MUST accept the Job Creation and ignores the excess subscriptions. If a subsequent Create-~~Job~~-
907 Subscription request would exceed this number, the Printer rejects the request and returns the 'server-error-
908 too-many-subscriptions' (see section 14.3).

909 **6.7 max-printer-subscriptions-supported (integer(0:MAX))**

910 This REQUIRED Printer attribute specifies the maximum number of Per-Printer subscriptions that are
911 supported by multiple Create-Printer-Subscription requests, i.e., the maximum number of un-expired Per-
912 Printer Subscription objects that the Printer supports at a time. A value of 0 indicates no effective
913 maximum. A Printer MUST support at least 1 Per-Printer subscription. If the number of Per-Printer
914 subscriptions exceeds the value of this attribute or would exceed some implementation-defined total
915 number of Per-Printer Subscriptions for the Printer (if any)~~would exceed this number~~, the Printer rejects the
916 Create-Printer-Subscription request and returns the 'server-error-too-many-subscriptions' (see section 14.3).

917 **6.8 notify-lease-time-supported (rangeOfInteger(0:MAX))**

918 This REQUIRED Printer attribute specifies the range of values in seconds that are supported for the
919 "notify-lease-time-requested" operation attribute in a Create-Printer-Subscription or Renew-Subscription
920 request for a Per-Printer subscription. When the lease time expires for a Per-Printer Subscription without
921 renewing, the Printer MUST delete the Subscription object. If the client requests a value outside this range,
922 the Printer MUST grant a value that is in this range (see section 5.9). A value of 0 indicates an infinite
923 lease, i.e., one that does not expire.

924 **6.9 notify-lease-time-default (integer(0:MAX))**

925 This REQUIRED Printer attribute specifies the value of the lease time that the Printer object has been
926 configured to assume if the client does not supply a "notify-lease-time-requested" operation attribute in the
927 Create-Printer-Subscription or Renew-Subscription requests.

928 **6.10 persistent-jobs-supported (boolean)**

929 This OPTIONAL Printer attribute indicates whether or not the Printer supports persistent Jobs, i.e., Jobs
930 object that are preserved across power cycles. If Jobs are persistent, then Per-Job Subscriptions MUST also
931 be persistent, since they are part of the Job object. It is RECOMMENDED that Jobs (and Per-Job
932 Subscriptions) be persistent.

933 **6.11 persistent-subscriptions-supported (boolean)**

934 This OPTIONAL Printer attribute indicates whether or not the Printer supports persistent Per-Printer
935 Subscriptions, i.e., Subscription objects that are preserved across power cycles. When this value is 'true'
936 the implementation MAY support some that are persistent and some that are not. If the value is 'false' or
937 the attribute is not supported, Per-Printer Subscriptions MUST NOT be persistent. It is RECOMMENDED
938 that Per-Printer subscriptions be persistent.

939 **6.12 printer-state-change-time (integer(1:MAX))**

940 This OPTIONAL Printer attribute records the time, i.e., copy of the Printer's "printer-up-time" attribute,
941 that the Printer's "printer-state" attribute was last changed. On power-up, the Printer populates the "printer-
942 state-change-time" from its "printer-up-time" attribute, so that it always has a value.

943 6.13 printer-state-change-date-time (dateTime)

944 This OPTIONAL Printer attribute records the date and time, i.e., copy of the Printer's "printer-current-time"
945 attribute, that the Printer's "printer-state" attribute was last changed. On power-up, the Printer populates the
946 "printer-state-change-date-time" from its "printer-current-time" attribute, so that it always has a value.

947 7 Notification Content

948 This section defines the Notification content that is sent to a Notification Recipient when an event occurs.
949 The Notification MAY be sent by the IPP Printer or a third party Notification Service (see section 2.3).

950 There are two notification content types: Machine Consumable and Human Consumable, i.e., 'text' MIME
951 media type. For most notification delivery methods both content types are defined. Each Notification
952 Content type will either define one specific Machine Consumable form (usual) or indicate that no Machine
953 Consumable form is defined. In addition, each Notification Content type will indicate whether (usual) or
954 not Human Consumable forms are permitted. But the definition will not define the Human Consumable
955 forms. For those Human Consumable forms that a Printer implementation supports as indicated in the
956 Printer's "notify-text-format-supported" attribute, it MUST support for all Notification Content formats
957 supported that permit Human Consumable form. Some of the notification delivery methods are defined to
958 support only one content type. A Printer MUST support all content types defined for a notification delivery
959 method, if it supports that notification delivery method.

960 7.1 Notification content MIME media type formats

961 This section defines the Notification content that the Notification Source sends asynchronously to each
962 Notification Recipient based on the subscription information stored with the subscription. The Notification
963 is either in a Machine Consumable or Human Consumable form.

964 7.2 Machine Consumable form

965 If the notification delivery method is defined to have a Machine Consumable form and that form is defined
966 to be the 'application/ipp' MIME media type [ipp-mod], then the following rules apply. Notifications are
967 generated using the following content formats:

968 ~~'application/ipp' Machine Consumable~~ The notification content MUST use ~~using~~ the 'application/ipp'
969 MIME media type [ipp-mod] using the Get-Job-Attributes response encoding for job events and Get-
970 Printer-Attributes response for printer events. The attributes listed in sections 7.4 and 7.5 are sent in an
971 notification for Job events. The attributes listed in sections 7.4 and 7.6 are sent in an notification for Printer
972 events. For any 'text' or 'name' attribute value in any notification, the charset and natural language rules
973 that apply to all IPP operations apply to these attributes as well, since they are represented as operation
974 responses. The Unsupported Attributes Group in the response is not sent. If the values of any of the
975 attributes sent in an notification content are not known, the value sent in the report content is the out-of-
976 band 'unknown' value, rather than omitting the attribute (see the beginning of [ipp-mod] section 4.1).

977 **Issue 2 - Should we change the Notification Model to allow notification delivery methods that are request**
978 **and response (in addition to the current model which has only one-directional notification delivery using**
979 **the 'application/ipp' operation response format?)**

980 **Issue 3** - If the answer to Issue 2 is yes, should we change the format of the notification content using
 981 'application/ipp' to always be a (new) Send-Notification operation request, whether the scheme returns a
 982 response or not?

983 An implementation MAY extend the contents of the Machine Consumable notification by adding additional
 984 attributes.

985 **7.3 Human Consumable form**

986 If the notification delivery method is defined to permit Human Consumable forms then the following
 987 RECOMMENDATIONS apply:

988 ~~'text/plain; charset=utf-8'~~ Human Consumable notification content type [RFC2046]. The text message
 989 SHOULD include information about the attributes in sections 7.4 and 7.5 for job events or in sections 7.4
 990 and 7.6 for printer events. This information is localized according to the information about natural
 991 language and charset in the subscription.

992 An implementation MAY extend the contents of a Human Consumable notification by adding additional
 993 information.

994 **1.27.4 Notification content attributes common to Job and Printer events**

995 This section lists the parameters and attributes that are included in both Job and Printer event Notifications.
 996 Some events do not include all of these attributes as shown in Table 4. Each notification content contains a
 997 single Job or Printer event, whether that event was subscribed using the Job Submission Subscription
 998 mechanism or the Per-Printer subscription mechanism. If either kind of subscription subscribed to both Job
 999 and Printer events, then they will be sent as separate Job notification content and Printer notification
 1000 contents to the same Notification Recipient. References of the form "mod m.n.o" refer to [ipp-mod]
 1001 sections.

1002 Table 4 lists the attributes that are defined for use in Notifications and indicates the Printer support required
 1003 for conformance: "R" indicates REQUIRED and "O" indicates OPTIONAL.

1004 **Table 4 - Common Job and Printer Notification content attributes**

	Reference	Events	
Attributes		'job-progress'	all others
1. version-number (integer (0:32767))	mod 3.1.1	R	R
2. status-code (integer (0:32767))	mod 3.1.1	R	R
3. request-id (integer (0:MAX))	5.7 & mod 3.1.1	R	R
4. attributes-charset (charset)	5.5 & mod 3.1.4	R	R
5. attributes-natural-language (naturalLanguage)	5.6 & mod 3.1.4	R	R
6. printer-uri (uri)	5.10	R	R
7. printer-name (name(127))	mod 4.4.4		R
8. job-id (integer(1:MAX))	mod 4.3.2	R	R**
9. job-name (name(MAX))	mod 4.3.5		R**

	Reference	Events	
Attributes		'job-progress'	all others
10. trigger-event (type2 keyword)	5.2	R	R
11. trigger-time (integer(MIN:MAX))		R	R
12. trigger-date-time (dateTime)			O
13. subscription-id (integer(1:MAX))	5.8	R	R
14. subscriber-user-name (name(MAX))	5.11	<u>R</u>	<u>RO</u>
15. subscriber-user-data (octetString(63))	5.4	<u>R</u>	R

1005 Attribute Notes:

1006 "status-code" - a value of 600 for a Job event and 601 for a Printer event.

1007 "request-id" - the sequence number for this subscription, starting at 1 for each subscription.

1008 "attributes-charset" - the value comes from the "notify-attributes-charset" attribute in the Subscription
1009 object.

1010 "attributes-natural-language" - the value comes from the "notify-attributes-natural-language" attribute in the
1011 Subscription object.

1012 "printer-uri" - the value comes from the "job-printer-uri" Job attribute for Per-Job subscriptions.

1013 **"job-id" and "job-name" - included in Printer event Notifications only for Per-Job subscriptions.

1014 "trigger-event" - the event that caused this Notification to be delivered.

1015 "trigger-time" - the "printer-up-time" value when the event occurred.

1016 "trigger-date-time" - the "printer-current-time" value when the event occurred - OPTIONAL to support.

1017 "job-name" - SNMP delivery method can truncate to less than 255 octets, since the Notification needs to fit
1018 into 484 octets or so on some transports that SNMP is defined for.

1019 "subscription-id" - the unique identifier for the Subscription object on this Printer.

1020 "subscriber-user-name" - the subscriber user name that created the Subscription object. SNMP delivery
1021 method can truncate to less than 255 octets, since the Notification needs to fit into 484 octets or so
1022 on some transports that SNMP is defined for.

1023 "subscriber-user-data" - opaque user data that may identify either the Subscriber and/or the ultimate
1024 Notification Recipient.

1.37.5 Additional Notification content attributes for Job events *only*

Table 5 [This section](#) lists the additional attributes that are included only in Job event Notifications [and indicates the Printer support required for conformance: "R" indicates REQUIRED, "O" indicates OPTIONAL, and "CR" indicates CONDITIONALLY REQUIRED, i.e., REQUIRED in a Notification if the corresponding Job attributes are supported.](#) Some events do not include all of these attributes as shown in Table 5.

Table 5 - Additional Notification content attributes for Job events *only*

Attributes	Reference	Events		
		'job-progress'	'job-completed'	all others
16. job-state (type1 enum)	mod 4.3.7	<u>R</u>	R	R
17. job-state-reasons (1setOf type2 keyword)	mod 4.3.8	<u>R</u>	R	R
18. job-k-octets-processed (integer(0:MAX))	mod 4.3.18.1	O	O	
19. job-impressions-completed (integer(0:MAX))	mod 4.3.18.2	<u>CR</u> \ominus	<u>CR</u> \ominus	
20. job-media-sheets-completed (integer(0:MAX))	mod 4.3.18.3	<u>CR</u> \ominus	<u>CR</u> \ominus	
21. job-collation-type (type2 enum)	[ipp-prog]	R		
22. sheet-completed-copy-number (integer(-2:MAX))	[ipp-prog]	R		
23. sheet-completed-document-number(integer(-2:MAX))	[ipp-prog]	R		
24. impressions-interpreted (integer(-2:MAX))	[ipp-prog]	R		
25. impressions-completed-current-copy (integer(-2:MAX))	[ipp-prog]	R		

1.47.6 Additional Notification content attributes for Printer events *only*

Table 6 lists the additional attributes that are included only in Printer event Notifications [and indicates the Printer support required for conformance: "R" indicates REQUIRED and "O" indicates OPTIONAL.](#)

Table 6 - Additional Notification content attributes for Printer events *only*

Attributes	Reference	Events
		all printer events
26. printer-state (type1 enum)	mod 4.4.11	R
27. printer-state-reasons (1setOf type2 keyword)	mod 4.4.12	R
28. printer-is-accepting-jobs (boolean)	mod 4.4.23	R

8 Operations for notification

This section defines all of the operations for notification.

8.1 Operations for Per-Job Subscriptions *only*

This section defines the ~~REQUIRED~~ operation requests and responses that are related to Per-Job subscriptions and its Subscription object. Section 8.3 defines the REQUIRED operation requests and responses associated with the REQUIRED Per-Printer subscription and its Subscription object.

8.1.1 Job Creation Operations (Create-Job, Print-Job, Print-URI) and Validate-Job

The usual method for a client to associate one subscription with a Job is to specify the subscription when the job is created. For a Per-Job Subscription, the client supplies the "job-notify (1setOf collection)" operation attribute with the member attributes listed in Table 7 with any of the job creation operations (Create-Job, Print-Job, Print-URI), plus Validate-Job (which doesn't create a job or subscription). If the client does not supply the "job-notify" attribute in the create operation, there is no subscription made (either implicitly or explicitly).

If a Printer does not support this notification specification, then it MUST ignore the "job-notify" operation attribute and return it in the response indicated as an attribute that is not supported. See [ipp-mod] section 3.1.7 for details on returning Unsupported Attributes.

Table 7 - Member attributes of the "job-notify" collection operation attribute

Member attribute of "job-notify" collection	Reference	REQUIRED in request	Printer support
notify-recipient (uri)	5.1	yes	REQUIRED
notify-events (1setOf type2 keyword)	5.2	no	REQUIRED
notify- content-type <u>text-format</u> (mimeMediaType)	5.3	no	REQUIRED
subscriber-user-data (octetString(63))	5.4	no	REQUIRED
notify- attributes -charset (charset)	5.5	no	OPTIONAL
notify- attributes -natural-language (naturalLanguage)	5.6	no	OPTIONAL

See the referenced sections for a definition of these operation attributes, since they are copied to the ~~Job~~ Subscription object as the ~~Job-Subscription~~ Description attributes described in section 5.

The following rules apply to Per-Job subscriptions created as part of the Job Creation operations:

- Any subscription can contain job events, printer events, or both.
- The Job Submission Subscription is only valid while the job is "not-completed". The job is "not-completed" while it is in the 'pending', 'pending-held', 'processing', and 'processing-stopped' states. The job changes from being "not-completed" to "retained" when it is done processing and enters any of the 'completed', 'canceled', or 'aborted' states. The job becomes "not-completed" again when it is restarted using the Restart-Job operation (see [ipp-mod]).
- Since no job is created for the Validate-Job operation, the only purpose of supplying the subscription operation attributes in the Validate-Job operation is to validate that the values are supported; the Printer object does not establish a notification subscription as a result of the Validate-Job operation.
- Since a Job Submission Subscription is included within a job submission operation, any interest in job events is limited to "this job" only (the Job object created because of this job creation operation). There

1069 is no mechanism to subscribe to events for all jobs or specifically some job other than this job in a
1070 create operation. But see the Create-Printer-Subscription operation (section 8.2.1) for an explicit
1071 operation to subscribe for job and/or printer events independently of any particular job submission, i.e.,
1072 Per-Printer subscriptions.

- 1073 5. Event reporting only occurs when a notification recipient has specified a subscription to any event(s).
- 1074 6. The notification implementation MAY allow an administrator to configure a policy on what events may
1075 be dropped.
- 1076 7. If the OPTIONAL "notify-~~attributes~~-charset" attribute is not supported or the supplied value is not
1077 supported, the IPP Printer MUST return the attribute in the Unsupported Attributes Group but still
1078 accept the operation, as with all Job create operations. In this case, the Printer MUST use the natural
1079 language supplied in the "attributes-charset" Job creation operation attribute, if that natural language
1080 value is supported by the Printer, else the Printer object MUST use the Printer's "charset-configured"
1081 value. See the Print-Job operation in [ipp-mod].
- 1082 8. If the OPTIONAL "notify-~~attributes~~-natural-language" attribute is not supported or the supplied value is
1083 not supported, the IPP Printer MUST return the attribute in the Unsupported Attributes Group but still
1084 accept the operation, as with all Job create operations. In this case, the Printer MUST use the natural
1085 language supplied in the "attributes-natural-language" Job creation operation attribute, if that natural
1086 language value is supported by the Printer, else the Printer object MUST use the Printer's "natural-
1087 language-configured" value. See the Print-Job operation in [ipp-mod].
- 1088 9. If a collection contains other unrecognized, unsupported member attributes and/or conflicting values,
1089 the attribute returned in the Unsupported Group is a collection containing the unrecognized,
1090 unsupported member attributes, and/or conflicting values. The Printer MUST return the unrecognized
1091 member attributes have-an-with the out-of-band value of 'unsupported'. The Printer MUST return the
1092 unsupported member attributes and conflicting values have-with their unsupported values. See [ipp-
1093 coll].
- 1094 10. If the number of events supplied in the "notify-events" attribute exceeds the Printer's "max-events-
1095 supported" attribute, the Printer MUST accept the request with the status code 'successful-ok-ignored-
1096 or-substituted-attributes' and return the "job-notify" collection in the Unsupported Attributes Group
1097 with only the "job-events" member attribute containing the events that exceed the maximum.
- 1098 11. If the Per-Job subscriptions would exceed the limit of Per-Job subscriptions supported per job as
1099 specified by the Printer's "max-job-subscriptions-supported" attribute or would exceed some
1100 implementation-defined limit on the total number of Per-Job subscriptions for the Printer (if any), the
1101 Printer MUST accept the request with the status code 'successful-ok-ignored-subscriptions', MUST
1102 return the "job-notify" attribute in the Unsupported Attributes Group with only the collection value(s)
1103 that represent the excess subscriptions that are being ignored, and MUST perform the Job Creation
1104 operation (see section 8.1.1), since the job can still be printed.
- 1105 If the job is accepted and one or more subscriptions are ignored, the status code returned is 'successful-
1106 ok-ignored-subscriptions. This status code is returned even if other job attributes are unsupported or in
1107 conflict. That is, if an IPP Printer finds a warning that would allow it to return 'successful-ok-ignored-
1108 subscriptions' and either 'successful-ok-ignored-or-substituted-attributes' and/or 'successful-ok-
1109 conflicting-attributes', it must return 'successful-ok-ignored-subscriptions'. In other words, the
1110 precedence for returning success codes is: 'successful-ok-ignored-subscriptions', 'successful-ok-
1111 conflicting-attributes', and 'successful-ok-ignored-or-substituted-attributes'.

8.1.2 Create-Job-Subscription operation

The ~~REQUIRED-OPTIONNAL~~ Create-Job-Subscription operation creates ~~either a Per-Job or a Per-Printer subscription by creating a~~ Subscription object. The client can specify one or more job and/or printer events to be delivered as notifications to one Notification Recipient. For the Per-Job subscription objects, the Job events are for this job only. ~~For the Per-Printer subscription, the job events are for any job submitted to the Printer. For both Per-Job and Per-Printer subscriptions,~~ The printer events are any events generated by that Printer for any job or when no job is involved at all (same as for Per-Job Subscriptions).

The Printer returns a subscription id and the length of time for which it has granted a lease for the subscription. ~~If the requested subscription object is a Per-Job subscription, the Printer MUST grant an infinite lease by returning a 0 value for the "notify lease time granted".~~

~~The client must renew the Per-Printer subscription using the Renew-Subscription operation (see section 8.2.4) before the lease runs out in order to maintain the subscription.~~ A client can unsubscribe using the Cancel-Subscription operation (section 8.3.4) and the subscription id.

Two Create-Job-Subscription operations with the same events and same Notification Recipient MUST be kept as distinct subscriptions and be assigned distinct subscription ids. A Printer MUST allow such duplicate subscriptions such that Cancel-Subscription doesn't unsubscribe both subscriptions and MUST send the Notifications twice to the Notification Recipient, since the "request-id" is supposed to count monotonically for each subscription.

If the Printer has a bounded set of concurrent Per-Job subscriptions) and the request would exceed that bound, the Printer rejects the operation and returns the 'server-error-too-many-subscriptions' status code. The client SHOULD try again later.

Access Rights: To create Per-Job subscription objects, the authenticated user (see [IPP-MOD] section 8.3) performing this operation MUST either be the job owner or have operator or administrator access rights for the Printer object (see [IPP-MOD] sections 1 and 8.5). ~~To create Per-Printer subscription objects, the authenticated user performing this operation MUST have Per-Printer subscription rights for this Printer.~~ Otherwise the IPP object MUST reject the operation and return: client-error-forbidden, client-error-not-authenticated, and client-error-not-authorized as appropriate.

1139 Request:

Group 1: Operation Attributes	Printer support
"attributes-charset" (charset)	<u>REQUIRED</u>
"attributes-natural-language" (naturalLanguage)	<u>REQUIRED</u>
"printer-uri" (uri)	<u>REQUIRED</u>
["job-id" (integer(1:MAX))]	<u>REQUIRED</u>
["requesting-user-name" (name(MAX))]	RECOMMENDED
["notify-recipient" (uri)]	<u>REQUIRED</u>
["notify-events" (1setOf type2 keyword)]	<u>REQUIRED</u>
["notify-content-type-text-format" (mimeMediaType)]	<u>REQUIRED</u>
["subscriber-user-data" (octetString(63))]	<u>REQUIRED</u>
["notify-attributes-charset" (charset)]	<u>OPTIONAL</u>
["notify-attributes-natural-language" (naturalLanguage)]	<u>OPTIONAL</u>
["notify-lease-time-requested" (integer(0:MAX))]	<u>R</u>
["notify-persistence-requested" (boolean)]	<u>OPTIONAL</u>

1140 Response:

Group 1: Operation Attributes	Printer Support
"status-code" (type2 enum)	<u>REQUIRED</u>
"attributes-charset" (charset)	<u>REQUIRED</u>
"attributes-natural-language" (naturalLanguage)	<u>REQUIRED</u>
["status-message" (text(255))]	<u>OPTIONAL</u>
["detailed-status-message" (text(MAX))]	<u>OPTIONAL</u>
"subscription-id" (integer(1:MAX))	<u>REQUIRED</u>
"notify-lease-expiration-time" (integer(0:MAX))	<u>R</u>
"notify-lease-time-granted" (integer(0:MAX))	<u>R</u>
"notify-persistence-granted" (boolean)	<u>REQUIRED</u>
Group 2: Unsupported Attributes	<u>REQUIRED</u>

1141 Attribute Notes:

1142 "job-id" (integer(1:MAX)) - the client MUST supply yes this attribute in order to create a Per-Job
 1143 subscription for the Job identified by the "job-id" value. If "job-id" is not supplied, then the
 1144 subscription is a Per-Printer subscription.

1145 Note: Unlike all other operations on the Job object, the "job-uri" operation attribute is not defined
 1146 for use with this operation.

1147 ISSUE 4 - Ok that "job-uri" isn't defined for use with the Create-Job-Subscription operation?

1148 "notify-recipient" (uri) - the client MUST supply this attribute in order to have a subscription.

1149 "notify-event" (1setOf type2 keyword) - if the client does not supply this attribute, the Printer populates the
 1150 "notify-events" Subscription Description attribute from its "notify-events-default" Printer
 1151 Description attribute.

1152 "notify-text-format" (mimeMediaType) - if the client supplies this attribute, the value indicates which
 1153 Human Consumable text format is requested for use in the Notification using the delivery method
 1154 that the client supplies in the "notify-recipient" attribute. If the client does not supply this attribute,

- 1155 the Machine-Consumable form of the delivery method that the client supplies in the "notify-
1156 recipient" attribute is used.
- 1157 ~~"notify-lease-time-requested" (integer(0:MAX))—the number of seconds requested for the subscription~~
1158 ~~lease. A value of 0 indicates a request that the Per-Printer-Subscription lease never expire.~~
1159 ~~Supplying a 0 value MAY require authentication in order to be used, if 0 is supported at all.~~
1160 ~~If the client does not supply this attribute, the Printer uses its "notify-lease-time-default" value (see~~
1161 ~~section 6.8).~~
- 1162 "notify-persistence-requested" (boolean) - whether or not the Per-Printer-Job Subscription is to be
1163 persistent, i.e., saved across power cycles. Note: Persistent trap registrations is a client option in
1164 SNMPv3 [RFC2573].
- 1165 ~~"notify-lease-expiration-time" (integer(0:MAX))—The Printer object MUST return this attribute which is~~
1166 ~~the time in the future at which the subscription lease will expire, i.e., the "printer-up-time" value (in~~
1167 ~~time ticks—see [ipp-mod] section 4.4.29) at which the Printer will delete the Subscription. A value~~
1168 ~~of 0 indicates that the lease subscription will never expire. If the Subscription object is a Per Job~~
1169 ~~subscription, the Printer MUST return 0 indicating that the lease doesn't expire, since Per Job~~
1170 ~~subscriptions expire when the Job object is no longer retained.~~
- 1171 ~~"notify-lease-time-granted" (integer(0:MAX))—The Printer object MUST return this attribute which is the~~
1172 ~~number of seconds of time actually granted. The value of "notify-lease-time-granted" may be less~~
1173 ~~than the requester requested in the "notify-lease-time-requested" if it was greater than the MAX~~
1174 ~~supported or more than the requester requested if it was less than the MIN supported, as indicated in~~
1175 ~~the Printer's "notify-lease-time-supported" (rangeOfInteger(0:MAX)) attribute. If the Subscription~~
1176 ~~object is a Per Job subscription, the Printer MUST return 0 indicating that the lease doesn't expire,~~
1177 ~~since Per Job subscriptions expire when the Job object is no longer retained.~~
- 1178 ~~ISSUE 6—Instead of Create-Subscription response returning "notify-lease-time-granted" which is NOT a~~
1179 ~~Subscription object attribute, it is cleaner for implementation to return the "notify-printer-up-time"~~
1180 ~~Subscription attribute instead, which is a Subscription object attribute. The client subtracts the "notify-~~
1181 ~~printer-up-time" from the "notify-lease-expiration-time" to get the number of second remaining in the lease,~~
1182 ~~just as it would for a Get-Subscript-Attributes response. OK to replace the "notify-lease-time-granted" with~~
1183 ~~the "notify-printer-up-time" attribute in the response?~~
- 1184 "notify-persistence-granted" (boolean) - whether or not this Subscription object instance is persistent. This
1185 attribute MUST be returned whether "notify-persistence-requested" is supported or not, so that the
1186 client knows which.
- 1187 ~~ISSUE 7—Should "notify-persistence-granted" be a Subscription object attribute too, so that all attributes~~
1188 ~~returned in a response are Subscription object attributes?~~
- 1189 **8.2 Operations for Per-Printer Subscriptions only**
- 1190 This section defines the operation requests and responses associated with the Per-Printer subscription and
1191 its Subscription object.

8.2.1 Create-Printer-Subscription operation

The REQUIRED Create-Printer-Subscription operation creates ~~either a Per-Job or a Per-Printer subscription by creating a~~ Subscription object. The client can specify one or more job and/or printer events to be delivered as notifications to one Notification Recipient. ~~For the Per-Job subscription, the Job events are for this job only.~~ For the Per-Printer subscription ~~objects~~, the job events are for any job submitted to the Printer. ~~For both Per-Job and Per-Printer subscriptions,~~ The printer events are any events generated by that Printer for any job or when no job is involved at all.

The Printer returns a subscription id and the length of time at which the subscription lease expires (which may be earlier or later than the client requested) ~~for which it has granted a lease for the subscription.~~ ~~If the requested subscription object is a Per-Job subscription, the Printer MUST grant an infinite lease by returning a 0 value for the "notify-lease-time-granted".~~

The client must renew the Per-Printer subscription using the Renew-Subscription operation (see section 8.3.3) before the lease runs out in order to maintain the subscription. A client can unsubscribe using the Cancel-Subscription operation (section 8.3.4) and the subscription id.

Two Create-Printer-Subscription operations with the same events and same Notification Recipient MUST be kept as distinct subscriptions and be assigned distinct subscription ids. A Printer MUST allow such duplicate subscriptions such that Cancel-Subscription doesn't unsubscribe both subscriptions and MUST send the Notifications twice to the Notification Recipient, since the "request-id" is supposed to count monotonically for each subscription.

If the Printer has a bounded set of concurrent Per-Printer subscriptions ~~and the request would exceed that bound~~, the Printer rejects the operation and returns the 'server-error-too-many-subscriptions' status code. The client SHOULD try again later.

Access Rights: ~~To create Per-Job subscription objects, the authenticated user (see [IPP-MOD] section 8.3) performing this operation MUST either be the job owner or have operator or administrator access rights for the Printer object (see [IPP-MOD] sections 1 and 8.5).~~ To create Per-Printer subscription objects, the authenticated user performing this operation MUST have Per-Printer subscription rights for this Printer. Otherwise the IPP object MUST reject the operation and return: client-error-forbidden, client-error-not-authenticated, and client-error-not-authorized as appropriate.

1220 Request:

Group 1: Operation Attributes	Printer support
"attributes-charset" (charset)	<u>REQUIRED</u>
"attributes-natural-language" (naturalLanguage)	<u>REQUIRED</u>
"printer-uri" (uri)	<u>REQUIRED</u>
["job-id" (integer(1:MAX))]	<u>R</u>
["requesting-user-name" (name(MAX))]	RECOMMENDED
["notify-recipient" (uri)]	<u>REQUIRED</u>
["notify-events" (1setOf type2 keyword)]	<u>REQUIRED</u>
["notify-content-type-text-format" (mimeType)]	<u>REQUIRED</u>
["subscriber-user-data" (octetString(63))]	<u>REQUIRED</u>
["notify-attributes-charset" (charset)]	<u>OPTIONAL</u>
["notify-attributes-natural-language" (naturalLanguage)]	<u>OPTIONAL</u>
["notify-lease-time-requested" (integer(0:MAX))]	<u>REQUIRED</u>
["notify-persistence-requested" (boolean)]	<u>OPTIONAL</u>

1221 Response:

Group 1: Operation Attributes	Printer Support
"status-code" (type2 enum)	<u>REQUIRED</u>
"attributes-charset" (charset)	<u>REQUIRED</u>
"attributes-natural-language" (naturalLanguage)	<u>REQUIRED</u>
["status-message" (text(255))]	<u>OPTIONAL</u>
["detailed-status-message" (text(MAX))]	<u>OPTIONAL</u>
"subscription-id" (integer(1:MAX))	<u>REQUIRED</u>
"notify-lease-expiration-time" (integer(0:MAX))	<u>REQUIRED</u>
"notify-lease-time-granted" <u>printer-up-time</u> (integer(1:MAX))	<u>REQUIRED</u>
"notify-persistence-granted" (boolean)	<u>REQUIRED</u>
Group 2: Unsupported Attributes	<u>REQUIRED</u>

1222 **ISSUE 5** - Ok that we aren't passing the operation attributes that are copied to the Subscription object in the
 1223 new Subscription object attributes group? Some of the "notify-xxx" attributes aren't Subscription
 1224 object attributes.

1225 Attribute Notes:

1226 ~~"job-id" (integer(1:MAX)) - the client supplies this attribute in order to create a Per-Job subscription for the~~
 1227 ~~Job identified by the "job-id" value. If "job-id" is not supplied, then the subscription is a Per-Printer~~
 1228 ~~subscription.~~

1229 "notify-recipient" (uri) - the client MUST supply this attribute in order to have a subscription.

1230 "notify-lease-time-requested" (integer(0:MAX)) - the number of seconds requested for the subscription
 1231 lease. A value of 0 indicates a request that the Per-Printer Subscription lease never expire.
 1232 Supplying a 0 value MAY require authentication in order to be used, if 0 is supported at all.
 1233 If the client does not supply this attribute, the Printer uses its "notify-lease-time-default" Printer
 1234 Description attribute value (see section 6.9).

- 1235 "notify-event" (1setOf type2 keyword) - if the client does not supply this attribute, the Printer populates the
1236 "notify-events" Subscription Description attribute from its "notify-events-default" Printer
1237 Description attribute.
- 1238 "notify-text-format" (mimeMediaType) - if the client supplies this attribute, the value indicates which
1239 Human Consumable text format is requested for use in the Notification using the delivery method
1240 that the client supplies in the "notify-recipient" attribute. If the client does not supply this attribute,
1241 the Machine-Consumable form of the delivery method that the client supplies in the "notify-
1242 recipient" attribute is used.
- 1243 "notify-persistence-requested" (boolean) - whether or not the Per-Printer Subscription is to be persistent,
1244 i.e., saved across power cycles. Note: Persistent trap registrations is a client option in SNMPv3
1245 [RFC2573].
- 1246 "notify-lease-expiration-time" (integer(0:MAX)) - The Printer object MUST return this attribute which is
1247 the time in the future at which the subscription lease will expire, i.e., the "printer-up-time" value (in
1248 time ticks - see [ipp-mod] section 4.4.29) at which the Printer will delete the Subscription. A value
1249 of 0 indicates that the lease subscription will never expire. ~~If the Subscription object is a Per Job~~
1250 ~~subscription, the Printer MUST return 0 indicating that the lease doesn't expire, since Per Job~~
1251 ~~subscriptions expire when the Job object is no longer retained.~~
- 1252 "notify-printer-up-time" (integer(1:MAX)) - The Printer object MUST return this attribute which is an alias
1253 for the Printer's "printer-up-time" Printer Description attribute. The client subtracts this value from
1254 the "notify-lease-expiration-time" value returned in order to determine the number of second in the
1255 future that the subscription will expire. This computed value "notify-lease-time-granted"
1256 ~~(integer(0:MAX)) - The Printer object MUST return this attribute which is the number of seconds of~~
1257 ~~time actually granted. The value of "notify-lease-time-granted" may be less than the requester~~
1258 ~~requested in the "notify-lease-time-requested" if it was greater than the MAX supported or more~~
1259 ~~than the requester requested if it was less than the MIN supported, as indicated in the Printer's~~
1260 ~~"notify-lease-time-supported" (rangeOfInteger(0:MAX)) attribute (see section 6.8). If the~~
1261 ~~Subscription object is a Per Job subscription, the Printer MUST return 0 indicating that the lease~~
1262 ~~doesn't expire, since Per Job subscriptions expire when the Job object is no longer retained.~~
- 1263 ~~ISSUE 6 - Instead of Create Subscription response returning "notify-lease-time-granted" which is NOT a~~
1264 ~~Subscription object attribute, it is cleaner for implementation to return the "notify-printer-up-time"~~
1265 ~~Subscription attribute instead, which is a Subscription object attribute. The client subtracts the "notify-~~
1266 ~~printer-up-time" from the "notify-lease-expiration-time" to get the number of second remaining in the lease,~~
1267 ~~just as it would for a Get-Subscript-Attributes response. OK to replace the "notify-lease-time-granted" with~~
1268 ~~the "notify-printer-up-time" attribute in the response?~~
- 1269 "notify-persistence-granted" - whether or not this Subscription object instance is persistent. This attribute
1270 MUST be returned whether "notify-persistence-requested" is supported or not, so that the client
1271 knows which.
- 1272 ~~ISSUE 7 - Should "notify-persistence-granted" be a Subscription object attribute too, so that all attributes~~
1273 ~~returned in a response are Subscription object attributes?~~

8.3 Common Operations for Per-Job and Per-Printer Subscriptions

This section defines the operations that are common to both Per-Job and Per-Printer subscriptions.

1.1.28.3.1 Get-Subscription-Attributes operation

The REQUIRED Get-Subscription-Attributes returns the requested attributes of the identified Subscription object. See section 5.

Request:

Group 1: Operation Attributes	Printer support
"attributes-charset" (charset)	<u>REQUIRED</u>
"attributes-natural-language" (naturalLanguage)	<u>REQUIRED</u>
"printer-uri" (uri)	<u>REQUIRED</u>
["requesting-user-name" (name(MAX))]	RECOMMENDED
"subscription-id" (integer(1:MAX))	<u>REQUIRED</u>
["requested-attributes" (1setOf type2 keyword)]	<u>REQUIRED</u>

Response:

Group 1: Operation Attributes	Printer support
"status-code" (type2 enum)	<u>REQUIRED</u>
"attributes-charset" (charset)	<u>REQUIRED</u>
"attributes-natural-language" (naturalLanguage)	<u>REQUIRED</u>
["status-message" (text(255))]	<u>OPTIONAL</u>
["detailed-status-message" (text(MAX))]	<u>OPTIONAL</u>
Group 2: Unsupported Attributes	<u>REQUIRED</u>
Group 3: <the requested Subscription object attributes>	<u>REQUIRED</u>

This operation is similar to the Get-Printer-Attributes operation. If the client omits the "requested-attributes" operation attribute, the Printer MUST respond as if the client had supplied the 'all' value, i.e., return all of the attributes supported for the Subscription object.

1.1.38.3.2 Get-Subscriptions operation

The REQUIRED Get-Subscriptions operation returns the requested attributes of the Subscription objects (see section 5).

Request:

Group 1: Operation Attributes	Printer support
"attributes-charset" (charset)	<u>REQUIRED</u>
"attributes-natural-language" (naturalLanguage)	<u>REQUIRED</u>
"printer-uri" (uri)	<u>REQUIRED</u>
["job-id" (integer(1:MAX))]	<u>REQUIRED</u>
["requesting-user-name" (name(MAX))]	RECOMMENDED
["limit" (integer(1:MAX))]	<u>REQUIRED</u>
["requested-attributes" (1setOf type2 keyword)]	<u>REQUIRED</u>
["my-subscriptions" (boolean)]	<u>REQUIRED</u>

Response:

Group 1: Operation Attributes	Printer support
"status-code" (type2 enum)	REQUIRED
"attributes-charset" (charset)	REQUIRED
"attributes-natural-language" (naturalLanguage)	REQUIRED
["status-message" (text)]	OPTIONAL
["detailed-status-message" (text(MAX))]	OPTIONAL
Group 2: Unsupported Attributes	REQUIRED
Group 3 to N:<the requested Subscription Attributes for each Subscription object in a separate group>	REQUIRED

1289

1290 Attribute Notes:

1291 This operation is similar to the Get-Jobs operation (see [ipp-mod]). If the client wants any attributes
 1292 returned, including the "subscription-id", it must include the attribute keyword name in the "requested-
 1293 attributes" operation attribute. If the "requested-attributes" operation attribute is omitted, the Printer MUST
 1294 respond as if the client supplied the value: 'subscription-id'.

1295 "job-id" (integer(1:MAX)) - If the client supplies this attribute, all of the Per-Job Subscription objects for
 1296 the identified job are candidates for return. If this attribute is omitted, all of the Per-Printer
 1297 Subscription objects are candidates for return.

1298 "my-subscriptions" (boolean) - If the client supplies the "my-subscriptions" with a 'false' value or omits it,
 1299 the Printer returns all subscriptions, subject to the security policy in force.

1300 Groups 3 to N: Subscription Object Attributes: The Printer object responds with one set of Subscription
 1301 Object Attributes for each returned Subscription object. The Printer object ignores (does not
 1302 respond with) any requested attribute or value which is not supported or which is restricted by the
 1303 security policy in force, including whether the requesting user is the user that created the
 1304 Subscription object (subscribing user) or not (see [ipp-mod] section 8).

1305 **1.1.48.3.3 Renew-Subscription operation**

1306 The REQUIRED Renew-Subscription operation permits a client to request the IPP Printer to extend the
 1307 lease on a Subscription object instance. There is no way to renew a Per-Job subscription, since they are
 1308 automatically canceled after the job completes and no longer has any documents, i.e., the job is no longer
 1309 retained (see [ipp-mod] section 4.3.7.2). If the requested subscription object is a Per-Job subscription, the
 1310 Printer MUST grant an infinite lease by returning a 0 value for the "notify-lease-~~expiration-time-granted~~".

1311 *Access Rights:* The authenticated user (see [IPP-MOD] section 8.3) performing this operation MUST either
 1312 be the owner of the Subscription object or have operator or administrator access rights for the Printer object
 1313 (see [IPP-MOD] sections 1 and 8.5). Otherwise the IPP object MUST reject the operation and return:
 1314 client-error-forbidden, client-error-not-authenticated, and client-error-not-authorized as appropriate.

1315 Request:

Group 1: Operation Attributes	Printer support
"attributes-charset" (charset)	<u>REQUIRED</u>
"attributes-natural-language" (naturalLanguage)	<u>REQUIRED</u>
"printer-uri" (uri)	<u>REQUIRED</u>
["requesting-user-name" (name(MAX))]	RECOMMENDED
"subscription-id" (integer(1:MAX))	<u>REQUIRED</u>
["notify-lease-time-requested" (integer(0:MAX))]	<u>REQUIRED</u>

1316 Response:

Group 1: Operation Attributes	Printer support
"status-code" (type2 enum)	<u>REQUIRED</u>
"attributes-charset" (charset)	<u>REQUIRED</u>
"attributes-natural-language" (naturalLanguage)	<u>REQUIRED</u>
["status-message" (text(255))]	<u>OPTIONAL</u>
["detailed-status-message" (text(MAX))]	<u>OPTIONAL</u>
"notify-lease-expiration-time" (integer(0:MAX))	<u>REQUIRED</u>
" notify-lease-time-granted printer-up-time" (integer(<u>10</u> :MAX))	<u>REQUIRED</u>
Group 2: Unsupported Attributes	<u>REQUIRED</u>

1317 Attribute Notes:

1318 "notify-lease-time-requested" (integer(0:MAX) - the number of seconds requested for the Per-Printer
1319 subscription lease. Same as Create-Printer-Subscriptions (see section 8.2.1).

1320 "notify-lease-expiration-time" - the time in the future when the subscription will expire. Same as for the
1321 Create-Printer-Subscription operation (see section 8.2.1).

1322 "~~notify-lease-time-granted~~printer-up-time" - the Printer object MUST return this attribute which is an alias
1323 for the Printer's "printer-up-time" Printer Description attribute. The client subtracts this value from
1324 the "notify-lease-expiration-time" value returned in order to determine the number of second in the
1325 future that the subscription will expire (see further explanation in section 8.2.1). the lease time
1326 (number of seconds) actually granted. Same as for the Create Subscription operation (see section
1327 8.2.1).

1328 ~~ISSUE 8 - Instead of Renew-Subscription response returning "notify-lease-time-granted" which is NOT a~~
1329 ~~Subscription object attribute, it is cleaner for implementation to return the "notify printer up time"~~
1330 ~~Subscription attribute instead, which is a Subscription object attribute. Ok to replace the "notify lease-~~
1331 ~~time-granted" with the "notify printer up time" attribute in the response?~~

1332 Note: There is no way to change any of the Subscription object attributes, except the "notify-lease-
1333 expiration-time" attribute (using the Renew-Subscription operation). In order to change other attributes, a
1334 client can create a new Subscription object and then use the Cancel-Subscription operation to cancel the old
1335 one (or do this in the other order, in case there is a limit on the number of Subscription object instances, as
1336 long as a short window with no Notifications is ok).

1337 Note: There is no need to renew a Per-Job Subscription, since it is effectively the time that the Job is active
1338 (see section 8.1).

1.1.58.3.4 Cancel- Subscription operation

The REQUIRED Cancel- Subscription operation allows a client to remove a Subscription object. No more Notifications are delivered for that Subscription. Once performed, there is no way to use that Subscription in the future. Subscription-ids should not be reused immediately, so that a stale reference situation is not created. Same as for Cancel-Job and job-ids.

Access Rights: The authenticated user (see [IPP-MOD] section 8.3) performing this operation MUST either be the owner of the Subscription object or have operator or administrator access rights for the Printer object (see [IPP-MOD] sections 1 and 8.5). Otherwise the IPP object MUST reject the operation and return: client-error-forbidden, client-error-not-authenticated, and client-error-not-authorized as appropriate.

Request :

Group 1: Operation Attributes	Printer support
"attributes-charset" (charset)	REQUIRED
"attributes-natural-language" (naturalLanguage)	REQUIRED
"printer-uri" (uri)	REQUIRED
["requesting-user-name" (name(MAX))]	RECOMMENDED
"subscription-id" (integer(1:MAX))	REQUIRED

Response :

Group 1: Operation Attributes	Printer support
"status-code" (type2 enum)	REQUIRED
"attributes-charset" (charset)	REQUIRED
"attributes-natural-language" (naturalLanguage)	REQUIRED
["status-message" (text(255))]	OPTIONAL
["detailed-status-message" (text(MAX))]	OPTIONAL
Group 2: Unsupported Attributes	REQUIRED

9 Comparison of Per-Job versus Per-Printer Subscriptions

Per-Job and Per-Printer subscriptions are quite similar. Either type of subscription can subscribe to Job events, Printer events, or both. Both types of subscriptions can be ~~created using the Create-Subscription operation~~, queried using the Get-Subscriptions and Get-Subscription-Attributes operations, and canceled using the Cancel-Subscription operation. Both types of subscriptions create Subscription objects which have the same attributes defined. However, there are some semantic differences between Per-Job subscriptions and Per-Printer subscriptions. A Per-Job Submission Subscription is established by the client when submitting a job and after creating the job using the Create-~~Job~~-Subscription operation by specifying the "job-id" of the job. A Per-Printer Subscription is established between a client and a Printer using the Create-~~Printer~~-Subscription operation. Some specific differences are:

1. A client usually creates a Per-Job subscription as part of the Job Creation operations (Create-Job, Print-Job, and Print-URI), rather than using the **OPTIONAL** Create-~~Job~~-Subscription operation, especially since some Printer implementations MAY not support the Create-~~Job~~-Subscription operation, since it is **OPTIONAL**.

- 1365 2. For Per-Job subscriptions, the subscription is only valid while the job is "not-complete" (see sections
1366 7.4 and 7.5) while for the Per-Printer subscriptions, the subscription is valid until the time (in seconds)
1367 that the Printer returned in the "notify-lease-~~expiration-time-granted~~" operation attribute expires.
- 1368 3. Job Events in a Per-Job subscription apply only to "one job" (the Job created by the job creation
1369 operation or references by the Create-Job-Subscription operation) while Job Events in a Per-Printer
1370 subscription apply to ALL jobs contained in the IPP Printer object.

10 Conformance Requirements

1373 This section further enhances the Conformance Requirements detailed in [IPP-MOD] section 5. Extensions
1374 made to the events herein must be made such that new events or event attributes are backward compatible
1375 to clients who implemented early versions of this notification specification.

1376 It is OPTIONAL to implement this Event Notification specification. If implemented, IPP objects MUST
1377 support all of the REQUIRED object attributes as defined in this document in the indicated sections.

1378 If IPP Notification is implemented, the operations described in this document must be supported as
1379 described in Table 8:

1380 **Table 8 - Conformance Requirements for Operations**

Attribute	Conformance requirements
"job-notify" in Job Creation operations (section 8.1.1)	REQUIRED
Create- Printer -Subscription (section 8.2.1)	REQUIRED
<u>Create-Job-Subscription (section 8.1.2)</u>	<u>OPTIONAL</u>
Get-Subscription-Attributes (section 8.3.1)	REQUIRED
Get-Subscriptions (section 8.3.2)	REQUIRED
Renew-Subscription (section 8.3.3)	REQUIRED
Cancel-Subscription (section 8.3.4)	REQUIRED

11 IANA Considerations

1383 IANA will be called on to register URL schemes for notification delivery for use in the "notification-
1384 recipient" attribute, using the same procedures outlined in [ipp-mod].

12 Internationalization Considerations

1386 This IPP notification specification continues the internationalization of [ipp-mod] for attributes containing
1387 text strings and names. A subscribing client can specify a different natural language and charset for each
1388 Notification content delivered to a Notification Recipient.

1389 The Human Consumable Notification content is a 'text/plain; charset=utf-8' by default where the
1390 Notification Sender has localized the text message as requested by the subscriber for the intended
1391 Notification Recipient.

13 Security Considerations

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

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

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

The standard security concerns (delivery to the right user, privacy of content, tamper proof content) apply to the notification delivery. IPP should use the security mechanism of the delivery method used. Some delivery mechanisms are more secure than others. Therefore, sensitive notifications should use the delivery method that has the strongest security.

14 Status Codes

The following status codes are defined as extensions for notification:

14.1 'successful-ok-ignored-subscriptions' (0x0003)

The number of subscriptions supplied in a Job Creation operation (Create-Job, Print-Job, Print-URI) exceeds either the limit of Per-Job subscriptions supported per job as specified by the Printer's "max-job-subscriptions-supported" attribute or some implementation-defined limit on the total number of Per-Job subscriptions for the Printer (if any). The Printer MUST accept the request with this status code, MUST return the "job-notify" attribute in the Unsupported Attributes Group with only the collection value(s) that represent the excess subscriptions that are being ignored, and MUST perform the Job Creation operation (see section 8.1.1), since the job can still be printed.

This status code is returned even if other job attributes are unsupported or in conflict. That is, if an IPP Printer finds a warning that would allow it to return 'successful-ok-ignored-subscriptions' and either 'successful-ok-ignored-or-substituted-attributes' and/or 'successful-ok-conflicting-attributes', it must return 'successful-ok-ignored-subscriptions'. In other words, the precedence for returning success codes is: 'successful-ok-ignored-subscriptions', 'successful-ok-conflicting-attributes', and 'successful-ok-ignored-or-substituted-attributes'.

1429 **~~1.1~~14.2 client-error-uri-notification-scheme-not-supported (0x04??)**

1430 The scheme of the client-supplied URI in a "notify-recipient" operation attribute in a Create-Printer-
1431 Subscription or Create-Printer-Subscription operation is not supported. See [ipp-mod] section 3.1.7.

1432 There is no corresponding Per-Job subscription error for a Job Creation operation, since the Printer object
1433 MUST ignore any errors in the "job-notify" operation attribute, MUST return the "notify-recipient"
1434 attribute in the Unsupported Attributes Group, and perform the Job Creation operation (see section 8.1.1),
1435 since the job can still be printed.

1436 **~~1.2~~14.3 server-error-too-many-subscriptions (0x04??)**

1437 The bounded set of concurrent Per-Printer subscriptions supported by the Printer object would be exceeded
1438 if this request were accepted.

1439 Note: There is no corresponding Per-Job subscription error, since the Printer object MUST ignore any
1440 errors in the "job-notify" operation attribute and perform the Job Creation operation (see section 8.1.1),
1441 since the job can still be printed.

1442 **~~1.3~~14.4 server-error-too-many-events (0x04??)**

1443 The client supplied more events in the "notify-events" operation attribute in a Create-Job-Subscription or
1444 Create-Printer-Subscription operation than the Printer supports, as indicated in its "max-events-supported"
1445 attribute (see section 6.4).

1446 There is no corresponding Per-Job subscription error for a Job Creation operation, since the Printer object
1447 MUST ignore any errors in the "job-notify" operation attribute, MUST return the "notify-events" attribute
1448 in the Unsupported Attributes Group with only the excess events that are being ignored, and perform the
1449 Job Creation operation (see section 8.1.1), since the job can still be printed.

1450 **15 Additions to the IPP Encoding and Transport Document**

1451 The Subscription object tag needs to be assigned in section 3.7.1 Delimiter Tags:

1452 **3.7.1 Delimiter Tags**

1453 The following table specifies the values for the delimiter tags:

Tag Value (Hex)	Delimiter
0x00	reserved
0x01	operation-attributes-tag
0x02	job-attributes-tag
0x03	end-of-attributes-tag
0x04	printer-attributes-tag
0x05	unsupported-attributes-tag
0x06	subscription-attributes-tag
0x07-0x0e	reserved for future delimiters
0x0F	reserved for future chunking-end-of-attributes-tag

1454 When an xxx-attributes-tag occurs in the protocol, it MUST mean that zero or more following attributes up
1455 to the next delimiter tag are attributes belonging to group xxx as defined in the model document, where xxx
1456 is operation, job, printer, unsupported, subscription.

1457 Doing substitution for xxx in the above paragraph, this means the following. When an operation-attributes-
1458 tag occurs in the protocol, it MUST mean that the zero or more following attributes up to the next delimiter
1459 tag are operation attributes as defined in the model document. When an job-attributes-tag occurs in the
1460 protocol, it MUST mean that the zero or more following attributes up to the next delimiter tag are job
1461 attributes or job template attributes as defined in the model document. When a printer-attributes-tag occurs
1462 in the protocol, it MUST mean that the zero or more following attributes up to the next delimiter tag are
1463 printer attributes as defined in the model document. When an unsupported-attributes-tag occurs in the
1464 protocol, it MUST mean that the zero or more following attributes up to the next delimiter tag are
1465 unsupported attributes as defined in the model document. When a subscription-attributes-tag occurs in the
1466 protocol, it MUST mean that the zero or more following attributes up to the next delimiter tag are
1467 subscription attributes as defined in the [ipp-not] document.

1468 Add a reference to [ipp-not].

1469 16 References

1470 [ipp-coll]

1471 deBry, R., , Hastings, T., Herriot, R., "Internet Printing Protocol/1.0 & 1.1: collection attribute
1472 syntax", <draft-ietf-ipp-collection-00.doc>, work in progress, September 9, 1999.

1473 [ipp-mod]

1474 deBry, R., , Hastings, T., Herriot, R., Isaacson, S., Powell, P., "Internet Printing Protocol/1.1: Model
1475 and Semantics", < draft-ietf-ipp-model-v11-04.txt>, work in progress, June 23, 1999.

1476 [ipp-not-hist]

1477 deBry, R., Lewis, H., Hastings, T., "Internet Printing Protocol/1.1: Requirements for IPP
1478 Notifications", <draft-ietf-ipp-not-change-history-00.doc>, work in progress, August 22, 1999.

1479 [ipp-not-req]

1480 deBry, R., Lewis, H., Hastings, T., "Internet Printing Protocol/1.1: Requirements for IPP
1481 Notifications", <draft-ietf-ipp-not-03.txt>, work in progress, August 11, 1999.

1482 [ipp-pro]

1483 Herriot, R., Butler, S., Moore, P., Tuner, R., "Internet Printing Protocol/1.1: Encoding and
1484 Transport", <draft-ietf-ipp-protocol-v11-03.txt>, work in progress, June, 1999.

1485 [ipp-prog]

1486 Hastings, T., Bergman, R., Lewis, H., "Proposed Job Progress Attributes for IPP", <draft-ietf-ipp-
1487 prog.txt> work in progress, May 18, 1999.

1488 [ipp-set2]

1489 Kugler, C., , Hastings, T., "Internet Printing Protocol/1.0 & 1.1: Additional Operations, Set 2",
1490 <draft-ietf-ipp-set2.txt>, work in progress, August 22, 1999.

1491 [RFC2046]
1492 Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types. N. Freed & N. Borenstein.
1493 November 1996. (Obsoletes RFC1521, RFC1522, RFC1590), RFC 2046.

1494 [RFC2119]
1495 S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119 , March 1997

1496 [RFC2279]
1497 F. Yergeau , "UTF-8, a transformation format of ISO 10646", RFC 2279. January 1998.

1498 [RFC2566]
1499 deBry, R., , Hastings, T., Herriot, R., Isaacson, S., Powell, P., "Internet Printing Protocol/1.0: Model
1500 and Semantics", RFC 2566, April 1999.

1501 **17 Author's Addresses**

1502 Scott A. Isaacson (Editor)
1503 Novell, Inc.
1504 122 E 1700 S
1505 Provo, UT 84606
1506
1507 Phone: 801-861-7366
1508 Fax: 801-861-2517
1509 e-mail: sisaacson@novell.com
1510

1511 Tom Hastings
1512 Xerox Corporation
1513 737 Hawaii St. ESAE 231
1514 El Segundo, CA 90245
1515
1516 Phone: 310-333-6413
1517 Fax: 310-333-5514
1518 e-mail: hastings@cp10.es.xerox.com
1519

1520 Roger deBry
1521 Utah Valley State College
1522 Orem, UT 84058
1523

1524 Phone: (801) 222-8000
1525 EMail: debryro@uvsc.edu
1526

1527 Jay Martin
1528 e-mail: jkm@underscore.com
1529

1530 Michael Shepherd
1531 Xerox Corporation

1532 800 Phillips Road MS 128-51E
1533 Webster, NY 14450
1534
1535 Phone: 716-422-2338
1536 Fax: 716-265-8871
1537 e-mail: mshepherd@crt.xerox.com
1538
1539 Ron Bergman (Editor)
1540 Dataproducts Corp.
1541 1757 Tapo Canyon Road
1542 Simi Valley, CA 93063-3394
1543
1544 Phone: 805-578-4421
1545 Fax: 805-578-4001
1546 Email: rbergman@dpc.com
1547

1548 **A. Appendix - Summary of Notification Attribute Usage**

1549 This appendix summarizes the usage of Notification attributes in the Job operation attribute collection, Job
1550 object, Subscription object, Notification content, Job operations, and Subscriptions operations.

Attributes	Objects		Notification content		Job operations		Subscription operations		
	Job	Subscription	Job Notification Content	Printer Notification Content	Job Creation Validate -Job	<u>Create-Job-Subscription</u>	Create-Printer-Subscription	Renew-Subscription	Cancel-Subscription
1. notify-recipient (uri)		R			[R]	<u>R</u>	[R]		
2. notify-events (1setOf type2 keyword)		R		100	[R]	<u>[R]</u>	[R]		
3. notify- content-type <u>text-format</u> (mimeType)		R			[R]	<u>[R]</u>	[R]		
4. subscriber-user-data (octetString(63))		R	R	R	[R]	<u>[R]</u>	[R]		
5. attributes-charset (charset)	R		R (6)	R (6)	R	<u>R</u>	R	R	R
6. notify- attributes -charset (charset)		O			[O]	<u>[O]</u>	[O]		
7. attributes-natural-language (naturalLanguage)	R		R (8)	R (8)	R	<u>R</u>	R	R	R
8. notify- attributes -natural-language (naturalLanguage)		O			[O]	<u>[O]</u>	[O]		
9. notify-lease-time-requested (integer(0:MAX))							[R]	[R]	
10. notify-lease-time-granted (integer(0:MAX))							<u>Return</u>	<u>Return</u>	
11. notify-lease-expiration-time (integer(0:MAX))		R*					Return	Return	
12. notify-persistence-requested (boolean)						<u>[O]</u>	[O]		
13. notify-persistence-granted (boolean)		<u>R</u>				<u>Return</u>	Return		
14. trigger-event (type2 keyword)			R (2)	R (2)					
15. trigger-time (integer(MIN:MAX))			R	R					
16. trigger-date-time (dateTime)			[O]	[O]					
17. request-id (integer (0:MAX))		R	R	R					
18. subscription-id (integer(1:MAX))		R	R	R			Return	R	R

1551 Legend: "R" - REQUIRED for Printer to support; "O" - OPTIONAL for Printer to support; "[]" - OPTIONAL for sender to send;
 1552 Return - Printer MUST return; * - MUST be 0 for Per-Job Subscription objects; (n) means the value comes from the indicated
 1553 Subscription Attribute.

1554

Attributes	Objects		Notification		Job operations		Subscription operations		
	Job	Subscription	Job Notification	Printer Notification	Job Creation Validate -Job	<u>Create -Job-Subscription</u>	Create-Printer-Subscription	Renew - Subscription	Cancel - Subscription
19. version-number (integer (0:32767))			R	R					
20. status-code (integer (0:32767))			(600)	(601)					
21. job-printer-uri (uri)	R								
22. printer-uri (uri)		R	R	R	R	<u>R</u>	R	R	R
23. printer-name (name(127))			R	R					
24. job-id (integer(1:MAX))	R		R	Per-Jb		<u>R</u>	<u>Per-Jb</u>		
25. job-name (name(MAX)) [SNMP can truncate]	R		R	Per-Jb					
26. requesting-user-name (name(MAX))	R								
27. subscriber-user-name (name(MAX))		R	<u>OR</u>	<u>OR</u>					
28. job-state (type1 enum)	R		R						
29. job-state-reasons (1setOf type2 keyword)	R		R						
30. notify-printer-up-time (integer(1:MAX))		R				<u>Return</u>	<u>Return</u>		
31. job-k-octets-processed (integer(0:MAX))	O		[pg/cp]						
32. job-impressions-completed (integer(0:MAX))	O		[pg/cp]						
33. job-media-sheets-completed (integer(0:MAX))	O		[pg/cp]						
34. job-collation-type (type2 enum)	O		[pg/cp]						
35. sheet-completed-copy-number (integer(-2:MAX))	O		[pg/cp]						
36. sheet-completed-document-number(integer(-2:MAX))	O		[pg/cp]						
37. impressions-interpreted (integer(-2:MAX))	O		[pg/cp]						
38. impressions-completed-current-copy (integer(-2:MAX))	O		[pg/cp]						

1555 Legend: "R" - REQUIRED for Printer to support; "O" - OPTIONAL for Printer to support; [] - OPTIONAL for sender to send;
1556 "Per-Jb" - REQUIRED if a Per Job subscription;
1557 "[pg/cp]" - OPTIONAL-REQUIRED in a 'job-progress' or 'job-completed' notification if the corresponding Job attributes are
1558 supported; (nn) - an alternate source of the data in a Notification.

Attributes	Objects		Notification		Job operations		Subscription operations		
	Printer object		Job Notification	Printer Notification	Job Creation, Validate -Job	<u>Create -Job-Subscription</u>	Create-Printer-Subscription	Renew - Subscription	Cancel - Subscription
39. printer-state (type1 enum)	R			R					
40. printer-state-reasons (1setOf type2 keyword)	R			R					
41. printer-is-accepting-jobs (boolean)	R			R					
42. notify-schemes-supported (1setOf uriScheme)	R								
43. notify-events-default (1setOf type2 keyword)	R								
44. notify-events-supported (1setOf type2 keyword)	R								
45. max-events-supported (integer(5:MAX))	R								
46. <u>notify-text-format-supported (1setOf mimeType)</u>		<u>CR</u>							
47. max-job-subscriptions-supported (integer(1:MAX))	R								
48. max-printer-subscriptions-supported (integer(0:MAX))	R								
49. notify-lease-time-supported (rangeOfInteger(0:MAX))	R								
50. notify-lease-time-default (integer(0:MAX))	R								
51. persistent-jobs-supported (boolean)	O								
52. persistent-subscriptions-supported (boolean)	O								
53. printer-state-change-time (integer(1:MAX))	O								
54. printer-state-change-date-time (dateTime)	O								

1559 Legend: "R" - REQUIRED for Printer to support; "O" - OPTIONAL for Printer to support; "[]" - OPTIONAL for sender to send;

1560 "CR" - CONDITIONALLY REQUIRED, i.e., REQUIRED if Human Consumable format supported;

1561

1562

18 Appendix C: Full Copyright Statement

1563

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

1564

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

1565

1566

1567

1568

1569

1570

1571

1572

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

1573

1574

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

1575

1576

1577

1578

1579