

1 INTERNET-DRAFT 9 ISSUES are highlighted like this. Revision marks are changes for this alternative
2 <draft-ietf-ipp-not-spec-01.doc>

3
4 S. Isaacson
5 Novell, Inc.
6 J. Martin
7 Underscore
8 R. deBry
9 Utah Valley State College
10 T. Hastings
11 Xerox Corporation
12 M. Shepherd
13 Xerox Corporation
14 R. Bergman
15 Dataproducts Corp.
16 September 109, 1999

17 Internet Printing Protocol/1.0 & 1.1: **IPP Event Notification Specification - Alternative**

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

19 Status of this Memo

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

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

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

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

29 **Abstract**

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

37 A subscription request and the Subscription object includes: the names of Job and/or Printer events, the
38 Notification Recipient URL, the Notification content MIME type, possibly some opaque data, and the
39 charset and natural language. In addition, the ~~Per-Printer~~-subscription request includes: the requested lease

40 time and persistency for the subscription. When the event occurs, a notification is generated and delivered
41 using the information specified in the subscription.

42 The full set of IPP documents includes:

- 43 Design Goals for an Internet Printing Protocol [IPP-REQ]
- 44 Rationale for the Structure and Model and Protocol for the Internet Printing Protocol [IPP-RAT]
- 45 Internet Printing Protocol/1.1: Model and Semantics [IPP-MOD]
- 46 Internet Printing Protocol/1.1: Encoding and Transport [IPP-PRO]
- 47 Internet Printing Protocol/1.0: Implementer's Guide [IPP-IIG]
- 48 Mapping between LPD and IPP Protocols [IPP LPD]

49

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

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

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

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

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

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

76
77**Table of Contents**

78	1	Introduction	7
79		1.1 Notification Overview	7
80	2	Model for Per-Job and Per-Printer Subscription and Event Notification.....	9
81		2.1 Model for Per-Job Subscription and Notification	9
82		2.2 Model for Per-Printer Subscription and Notification	11
83		2.3 Relationship between the Printer object and the Notification Delivery Service	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)	20
91		5.2 notify-events (1setOf type2 keyword).....	20
92		5.3 notify-content-type (mimeMediaType)	23
93		5.4 subscriber-user-data (octetString(63)).....	23
94		5.5 notify-attributes-charset (charset).....	24
95		5.6 notify-attributes-natural-languages (naturalLanguage)	24
96		5.7 request-id	24
97		5.8 subscription-id (integer (1:MAX))	24
98		5.9 notify-lease-expiration-time (integer(0:MAX))	25
99		5.10 printer-uri (uri).....	25
100		5.11 subscriber-user-name (name(MAX))	25
101		5.12 notify-printer-up-time (integer(1:MAX)).....	26
102	6	Printer Description Attributes related to Notification	26
103		6.1 notify-schemes-supported (1setOf uriScheme)	27
104		6.2 notify-events-default (1setOf type2 keyword)	27
105		6.3 notify-events-supported (1setOf type2 keyword).....	27
106		6.4 max-events-supported (integer(5:MAX)).....	27
107		6.5 max-job-subscriptions-supported (integer(1:MAX))	27
108		6.6 max-printer-subscriptions-supported (integer(0:MAX)).....	27
109		6.7 notify-lease-time-supported (rangeOfInteger(0:MAX))	28
110		6.8 notify-lease-time-default (integer(0:MAX))	28
111		6.9 persistent-jobs-supported (boolean)	28
112		6.10 persistent-subscriptions-supported (boolean).....	28
113		6.11 printer-state-change-time (integer(1:MAX)).....	28
114		6.12 printer-state-change-date-time (dateTime).....	28
115	7	Notification Content	29
116		7.1 Notification content MIME media type formats	29
117		7.2 Notification content attributes common to Job and Printer events.....	29

118	7.3 Additional Notification content attributes for Job events <i>only</i>	31
119	7.4 Additional Notification content attributes for Printer events <i>only</i>	31
120	8 Operations for notification	32
121	8.1 Operations for Per-Job Subscriptions <i>only</i>	32
122	8.1.1 Job Creation Operations (Create-Job, Print-Job, Print-URI) and Validate-Job	32
123	8.2 Common Operations for Per-Job and Per-Printer Subscriptions.....	34
124	8.2.1 Create-Subscription operation.....	34
125	8.2.2 Get-Subscription-Attributes operation	37
126	8.2.3 Get-Subscriptions operation	38
127	8.2.4 Renew-Subscription operation	39
128	8.2.5 Cancel- Subscription operation	40
129	9 Comparison of Per-Job versus Per-Printer Subscriptions	41
130	10 Conformance Requirements	41
131	11 IANA Considerations	42
132	12 Internationalization Considerations.....	42
133	13 Security Considerations.....	42
134	14 Status Codes	42
135	14.1 client-error-uri-notification-scheme-not-supported (0x04??)	43
136	14.2 server-error-too-many-subscriptions (0x04??).....	43
137	14.3 server-error-too-many-events (0x04??).....	43
138	15 Additions to the IPP Encoding and Transport Document	43
139	3.7.1 Delimiter Tags.....	43
140	16 References	44
141	17 Author's Addresses	45
142	A. Appendix - Summary of Notification Attribute Usage	46
143	18 Appendix C: Full Copyright Statement.....	50
144		

145 **Tables**

146 Table 2 - Summary of Per-Job and Per-Printer Subscription operations9

147 Table 4 - Subscription object attributes.....19

148 Table 6 - Printer Description attributes associated with Notification26

149 Table 7 - Common Job and Printer Notification content attributes.....30

150 Table 9 - Additional Notification content attributes for Job events *only*31

151 Table 11 - Additional Notification content attributes for Printer events *only*.....31

152 Table 13 - Member attributes of the "job-notify" collection operation attribute.....32

153 Table 15 - Conformance Requirements for Operations41

154 **Figures**

155 Figure 1 - Client-Printer Per-Job Subscription and Notification Model10

156 Figure 3 - Client-Server-Printer Per-Job Subscription and Notification Model.....11

157 Figure 5 - Client-Printer Per-Printer Subscription and Notification Model11

158 Figure 7 - Opaque Use of a Notification Service12

159 Figure 9 - Object Model for Notification16

160

161

162 1 Introduction

163 This IPP notification specification is an extension to IPP/1.0 [RFC2568, RFC2569]. This document in
164 combination with the following documents is intended to meet the notification requirements described in
165 [ipp-not-req]:

- 166 Internet Printing Protocol/1.0 & 1.1: "Collection Attribute Syntax" [ipp-coll]
- 167 Internet Printing Protocol/1.0 & 1.1: "Job Progress Attributes" [ipp-prog]
- 168 Internet Printing Protocol/1.0 & 1.1: "Notification Change History" [ipp-not-hist]
- 169 Internet Printing Protocol/1.0 & 1.1: "Notification Delivery Method xxx [TBD]
- 170 Internet Printing Protocol/1.0 & 1.1: "Notification Delivery Method yyy [TBD]

171

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

- 173 - The rest of Section 1.1 is an overview of IPP Notification.
- 174 - Section 2 is the model for network entities that use IPP notification, including clients (desktop and
175 servers), IPP Printers (servers and devices), and Notification Recipients.
- 176 - Section 3 is the terminology used throughout the document.
- 177 - Section 4 is the object model for notification, including Job, Printer, and Subscription objects.
- 178 - Section 5 [and](#) 6 defines the notification attributes for each of the [Job, Printer, and Subscription](#) [and](#)
179 [Printer](#) objects.
- 180 - Section 7 defines the content of Human Consumable and Machine Consumable Event Notification
181 contents.
- 182 - Sections 8 and 9 define the Per-Job and Per-Printer Subscription [mechanisms](#)[operations](#).
- 183 - Section 10 and 11 define the conformance requirements and IANA requirements, respectively.
- 184 - Section 12 - 14 cover Internationalization, Security, and Status codes.

185 1.1 Notification Overview

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

188 One or more Per-Job Submission subscriptions are specified by the client when submitting a job. One or
189 more [Per-Job or](#) Per-Printer subscriptions are created by performing separate explicit Create-~~Printer-~~
190 Subscription operations on the Printer.

191 A Per-Job or Per-Printer subscription [request](#) includes:

- 192 1. the names of Job and/or Printer events that are of interest to the Notification Recipient
- 193 2. the delivery method and address to use to deliver the notification to one Notification Recipient
- 194 3. whether Human Consumable or Machine Consumable notification content is to be sent

- 195 4. some opaque data that the subscriber wants to be send to the Notification Recipient in the
- 196 Notification, perhaps to identify either the subscriber or the ultimate recipient
- 197 5. the charset to use in the Notification, if it is to be different than the one used in the request that
- 198 created the subscription
- 199 6. the natural language to use in the Human Consumable Notification, if it is to be different than the
- 200 one used in the request that created the subscription

201 ~~In addition, a Per-Printer subscription includes:~~

- 202 7. the requested lease time in seconds for the subscription
- 203 8. whether or not the subscription is requested to be persistent across power cycles.

204 For Per-Job subscriptions, a client requests job and printer event notification using the "job-notify"
 205 operation attribute when creating a job with any of the Job Creation operation: Print-Job, Print-URI, and
 206 Create-Job. The "job-notify" operation attributes may be submitted to the Validate-Job in order to be
 207 validated. ~~The "job-notify" operation attribute is copied to the Job object as a Job Description attribute and~~
 208 ~~so may be queried using the Get Job Attributes and Get Jobs operations (see [ipp-mod]). Also this "job-~~
 209 ~~notify" Job Description attribute may be changed using the Set Job Attributes (see [ipp-set2]).~~ The "job-
 210 notify" operation attribute contains one or more collection values, each consisting of a number of member
 211 attributes that specify a subscription, so that a Job can contain have more than one Per-Job subscription.
 212 The 'collection' is a new attribute syntax (see [ipp-coll]). The member attributes of each collection value
 213 are copied to separate Subscription objects to populate the corresponding Subscription Description
 214 attributes. ~~Table 1 summarizes the Per-Job Subscription operations and their salient input operation~~
 215 ~~attributes. All operations are REQUIRED, except the Set Job Attributes operation.~~

216 **Table 1 – Summary of Per-Job Subscription operations**

Operation:	salient inputs besides printer uri:
Print Job, Print URI, Create-Job	1setOf {recipient, [events,] [content type,] [user data,] [charset,] [natural language]}
Validate-Job	1setOf {recipient, [events,] [content type,] [user data,] [charset,] [natural language]}
Get Job Attributes	job id, [requested attributes]
Set Job Attributes	job id, 1setOf {[recipient,] [events,] [content type,] [user data,] [charset,] [natural language]}

217

218 For Per-Printer subscriptions and Per-Job subscriptions created after the Job has been created, a client
 219 requests job and printer event notification using new operations independent of any job. The Printer keeps
 220 each subscription in a separate Subscription object ~~which is similar to a Job object in its operations and~~
 221 ~~security.~~ The Get Printer Attributes (see [ipp-mod]) returns the supported notification capabilities
 222 supported. ~~The Create-Printer-Subscription operation creates an instance of the Subscription object~~
 223 ~~supplying these new operation attributes and returns a subscription-id (analogous to a job-id for a Job~~
 224 ~~object). These operation attributes are copied to the Subscription object as Subscription Description~~
 225 ~~attributes and so may be queried using the Get-Subscription-Attributes and Get-Subscriptions operations.~~
 226 The subscriber requests a lease time for each Per-Printer subscription which MAY be infinite. The Printer

227 grants a lease time according to its configured policy. A client MUST renew the Subscription before the
 228 granted lease time expires using the Renew-~~Printer~~-Subscription operation.

229 Table 2 summarizes the Per-Job and Per-Printer-Subscription operations and their salient input operation
 230 attributes.

231 **Table 2 - Summary of Per-Job and Per-Printer Subscription operations**

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

232

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

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

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

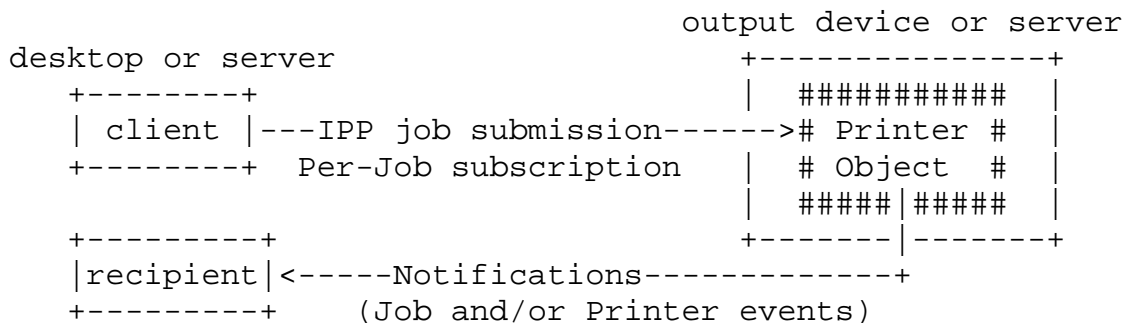
244 **2 Model for Per-Job and Per-Printer Subscription and Event Notification**

245 **1.1 Model for Per-Job Subscription and Notification**

246 Per-Job subscriptions are created by -a client (desktop or server acting as a client) as part of creation of the
 247 job in an IPP Printer (printing device or server). More than one subscription may be submitted with a job.
 248 Additional subscriptions may be associated with the job using the Create-Subscription operation. The IPP

249 Printer object delivers a Notifications to the Notification Recipient supplied by the Client in each
 250 subscription. A Notification Recipient can be the Job submitter or a third party.

251 Figure 1 shows the Per-Job subscription notification model for a simple Client - Printer relationship.
 252
 253 embedded printer:



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

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

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

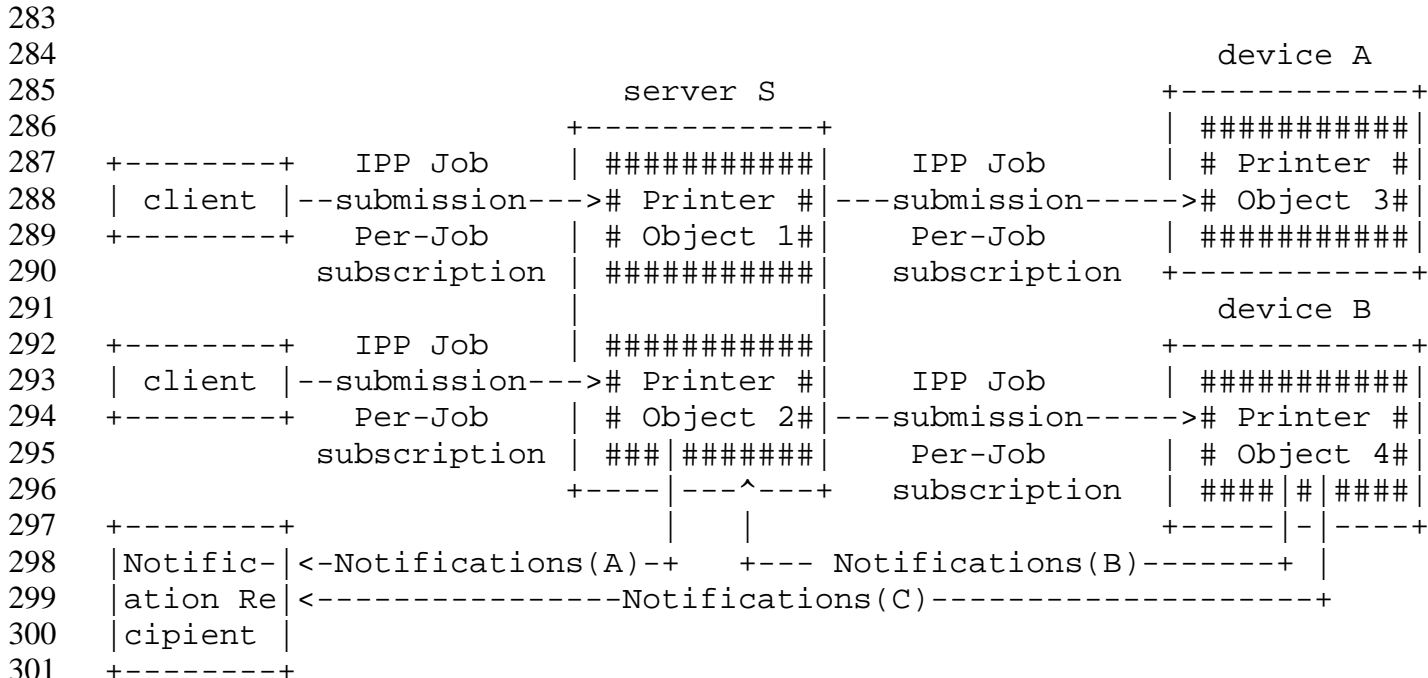


Figure 2 - Client-Server-Printer Per-Job Subscription and Notification Model

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.

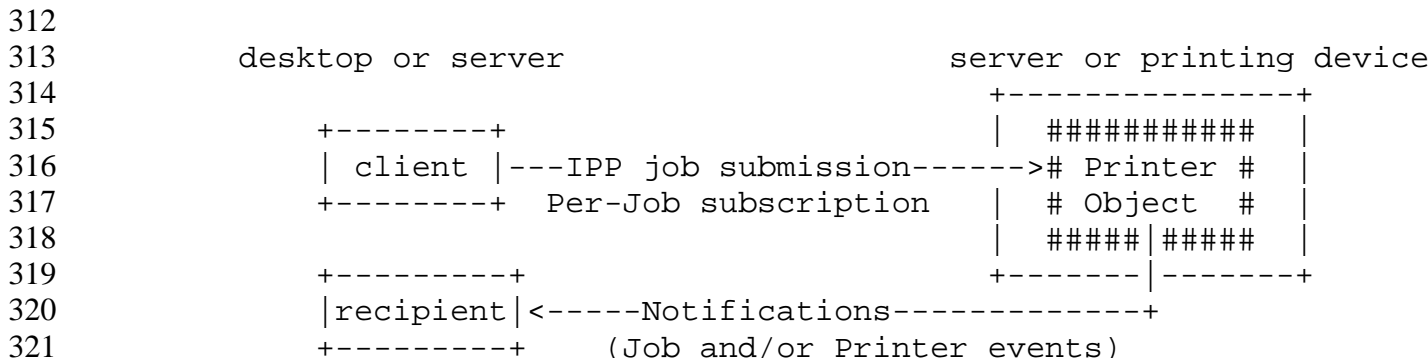
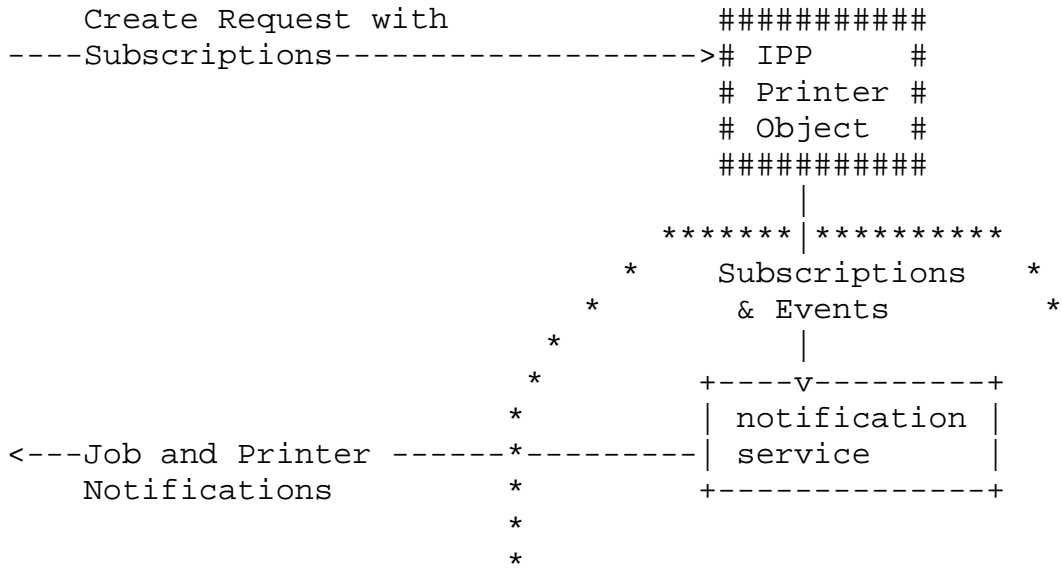


Figure 3 - Client-Printer Per-Printer Subscription and Notification Model

323 **2.3 Relationship between the Printer object and the Notification Delivery Service**

324 The IPP Notification model does not mandate that the IPP Printer object implement the full semantics of
 325 subscription, report generation, and multiple delivery methods itself. An implementation may be
 326 configured to use some other notification service. Figure 4 shows this partitioning.



345 *** = Implementation configuration opaque boundary

347 **Figure 4 - Opaque Use of a Notification Service**

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

- 349 1. The IPP Printer forwards the Subscriptions and the Events to the other notification service
 350 which then sends the Notifications to the proper Recipients.
- 351 2. The IPP Printer keeps the subscriptions and uses them to send Notifications to the other
 352 notification service which forwards them to Recipients that have also subscribed directly with
 353 the other notification service. Examples of this latter approach are paging and immediate
 354 messaging services.

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

357 **3 Terminology**

358 This section defines terminology used throughout this document.

359 **3.1 Conformance Terminology**

360 Capitalized terms, such as **MUST, MUST NOT, REQUIRED, SHOULD, SHOULD NOT, MAY,**
 361 **NEED NOT, and OPTIONAL,** have special meaning relating to conformance to this specification.

362 These terms are defined in [ipp-mod section 13.1 on conformance terminology, most of which is
363 taken from RFC 2119 [RFC2119].

364 *Since support of this entire notification specification is OPTIONAL for conformance to*
365 *IPP/1.0 or IPP/1.1, the use of the term REQUIRED in this document means*
366 *"REQUIRED if this notification specification is implemented". Likewise, the term*
367 *RECOMMENDED means "RECOMMENDED if this notification specification is*
368 *implemented".*

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

371 3.2 Other terminology

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Per-Printer Subscription - A Subscription that a client specifies using an explicit -Create-Printer-Subscription operation with a Printer object as the target, ~~that is attached to a Printer object.~~

Notification Source - The entity that sends Event Notifications.

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

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

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

Events can be classified along two dimensions:

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

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

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

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

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

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

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

461 **Store and Forward Notification** - A Notification which are not necessarily delivered to Notification
462 Recipients immediately, but is queued for delivery by some intermediate network application, for
463 later retrieval. Email is an example of a store and forward notification delivery method.

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

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

471 4 Object Model for Notification

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

474 ~~Per Job subscriptions do not use the Subscription object. Instead, a Job can have one or more Per Job~~
475 ~~subscription which is represented entirely as a single Job attribute. Each Per Job subscription is~~
476 ~~represented as a collection value of the "job-notify" Job attribute.~~

477 The object relationships can be seen pictorially as:

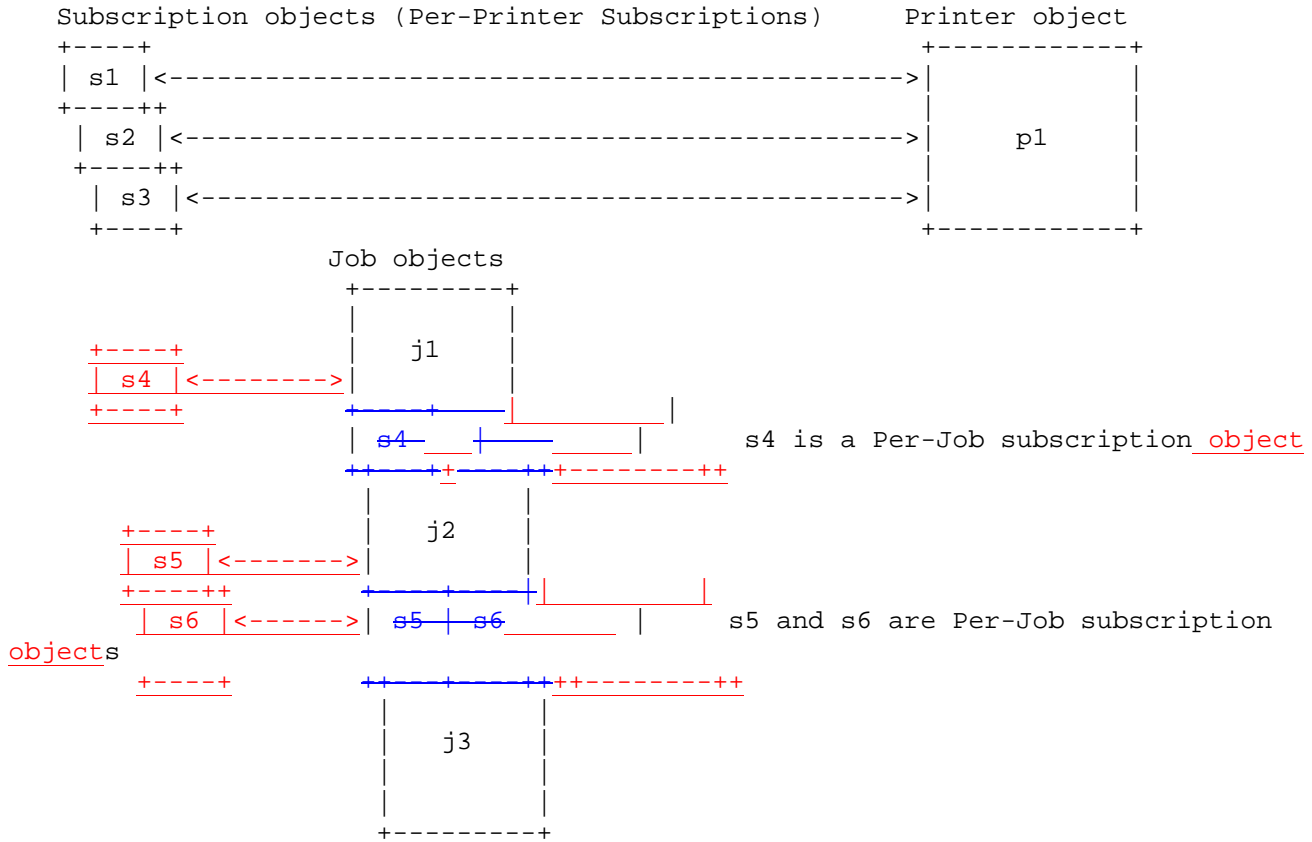


Figure 5 - Object Model for Notification

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

s4, s5, and s6 are Per-Job subscription objects (~~not objects~~) and can identify Printer and/or Job events.

4.1 Object relationships

The object relationships can be stated as follows:

1. The Printer object contains zero or more Per-Printer Subscription objects (p1 contains s1-s3 Per-Printer Subscription objects).
2. Each Per-Printer Subscription object (s1, s2, and s3) is contained in one Printer object (p1) and represents one Per-Printer subscription.
3. Each "Per-Printer" Subscription object identifies one or more Job and/or Printer events. Such Job events are for *all* jobs on the Printer. Such Printer events are for *any* Printer event, no matter which job is processing and when no jobs are processing.
4. A Job object contains-is associated with zero or more Per-Job subscriptions (~~not an objects~~). Job j1 is associated with contains Per-Job subscription object s4, Job j2 is associated with contains Per-Job subscription objects s5 and s6, and Job j3 does is not associated with contain-any Per-Job

523 subscription object. Note: the IPP notification interface semantics are defined so that an
524 implementation MAY associate Per-Job Subscription objects with Job objects by having the Job
525 objects actually contain its associated Per-Job Subscription objects or MAY just use some internal
526 bi-directional linking mechanism between the Job and Subscription objects. Either implementation
527 technique is transparent to the client.

- 528 5. Each "Per-Job" subscription (~~not an object by itself, but is attributes of the Job object~~) identifies one
529 or more Job and/or Printer events. Such Job events are only for *this* job (different than "per-Printer"
530 Subscriptions). Such Printer events are for *any* Printer event, no matter which job and when no jobs
531 are processing (same as for "per-Printer" Subscriptions).
- 532 6. A Per-Printer Subscription object cannot be contained in or associated with more than one Printer
533 object.
- 534 7. A Per-Job Subscription object cannot be contained in or associated with more than one Job object.

535 **5 Subscription Object attributes defined for Notification**

536 The following notification attributes are defined for ~~each of the Printer, Job, and the~~ Subscription objects.
537 The definitions of the object attributes are specified here so that they can be referred to from the
538 subsequence definitions of the operations that set them.

539 **5.1 "job-notify" (1setOf collection) Job Description attribute**

540 ~~In order to permit more than one Per-Job subscription, each subscription is represented as a collection value~~
541 ~~of the "job-notify (1setOf collection)" Job Description attribute. An implementation MUST support at least~~
542 ~~one collection value of the "job-notify" attribute. Table 3 lists the member attributes of the "job-notify"~~
543 ~~attribute. Section 5.3 defines each of the "job-notify" member attributes.~~

544 ~~The "job-notify" Job Description attribute MAY be modified by the Set-Job-Attributes operation (see~~
545 ~~section 7.4) in which case all of its member attributes are replaced and the READ-ONLY "request-id"~~
546 ~~member attribute is reset with a value of '0' (see section 5.3.7).~~

547

Table 3 – Job object 'job-notify' collection member attributes

Job object attributes:	Printer support	set by:
notify-recipient (uri)	R	job creation operation, Set-Job-Attributes
notify-events (1setOf type2 keyword)	R	job creation operation, Set-Job-Attributes
notify-content-type (mimeMediaType)	R	job creation operation, Set-Job-Attributes
subscriber-user-data (octetString(63))	R	job creation operation, Set-Job-Attributes
notify-charset (charset)	Ø	job creation operation, Set-Job-Attributes
notify-natural-languages (naturalLanguage)	Ø	job creation operation, Set-Job-Attributes
request-id (integer(0:MAX))	R	Printer event occurring

548

549

5.2 Subscription object attributes

551 ~~Each Per-Printer subscription is contained in a separate Subscription object. An implementation MUST~~
 552 ~~support at least one Subscription object instance. Table 4 lists the Subscription attributes defined for the~~
 553 ~~Subscription object. Sections 5.3 and 5.4 define each of the Subscription attributes.~~

554 ~~All of the Subscription object attributes are READ-ONLY. They are set by other operations.~~

555

Table 4 - Subscription object attributes

Subscription object attributes:	Printer support	READ-ONLY, set by:
notify-recipient (uri)	R	client - <u>Job Creation and Create-Printer-Subscription</u>
notify-events (1setOf type2 keyword)	R	client - <u>Job Creation and Create-Printer-Subscription</u>
notify-content-type (mimeMediaType)	R	client - <u>Job Creation and Create-Printer-Subscription</u>
subscriber-user-data (octetString(63))	R	client - <u>Job Creation and Create-Printer-Subscription</u>
notify-charset (charset)	O	client - <u>Job Creation and Create-Printer-Subscription</u>
notify-natural-languages (1setOf naturalLanguage)	O	client - <u>Job Creation and Create-Printer-Subscription</u>

556

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

557

558 Not in a Per Job subscription:

subscription-id (integer(1:MAX))	R	Printer - <u>Job Creation and Create-Printer-Subscription</u>
notify-lease-expiration-time (integer(0:MAX))	R	Printer - <u>Job Creation (set to 0) and Create-Printer-Subscription, Renew-Printer-Subscription</u>
printer-uri (uri)	R	Printer - <u>Job Creation and Create-Printer-Subscription</u>
subscriber-user-name (name(MAX))	R	Printer - <u>Job Creation and Create-Printer-Subscription</u>

559

notify-printer-up-time (integer(1:MAX))	R	Printer - returned by Get-Printer-Subscription-Attributes and Get-Printer-Subscriptions
---	---	---

560

561

562

563

564

565

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

566 **5.3 Common Per-Job and Per-Printer Subscription attributes**

567 ~~The following sub-sections define attributes that are in common to both Per-Job and Per-Printer~~
568 ~~subscriptions, i.e., are member attributes of the "job-notify" collection Job-Description attribute and are~~
569 ~~Subscription object attributes. Section 5.4 defines additional attributes that are only Subscription object~~
570 ~~attributes.~~

571 **5.1 notify-recipient (uri)**

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

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

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

- 579 1) The URI scheme used.
- 580 2) The supported and default delivery format, and if not one of the specified types in Section 5.3,
581 description of the notification content.
- 582 3) Any content length restrictions imposed by the delivery protocol.
- 583 4) The latency of the delivery protocol used.
- 584 5) The reliability of the transport and delivery protocol used.
- 585 6) The security aspects of the transport and delivery protocol used, e.g. how it is handled in firewalls.
- 586 7) How the delivery protocol is initiated, e.g. does it have to be initiated by the receiving user (pull), or
587 is it initiated by the notification service (push).

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

591

592 **5.2 notify-events (1setOf type2 keyword)**

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

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

599 The events are defined to be disjoint.

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

601 The standard job event keyword values are:

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

604 **'job-created'**: REQUIRED - the Printer object has accepted a job creation operation (Print-Job, Print-
605 URI, or Create-Job) and the job's "time-at-creation" attribute value is set (see [ipp-mod] section
606 4.3.14.1). The Printer puts the job in the 'pending', 'pending-held' or 'processing' states.

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

610 **'job-completed'**: REQUIRED - the job has reached one of the completed states, i.e., the value of the
611 job's "job-state" attribute has changed to: 'completed', 'aborted', or 'canceled'. The Job's "time-at-
612 completed" and "date-time-at-completed" (if supported) attributes are set (see [ipp-mod] section
613 4.3.14).

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

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

621
622 This event also indicates that one or more values have been added to or removed from the Job's
623 "job-state-reasons" attribute, such as 'job-queued' or 'job-printing', whether or not the job's state has
624 changed. If job state reasons are added when the job is created, only the 'job-created' event is
625 generated, in order to keep the events disjoint. If job state reasons are added or removed when the
626 job is completed, only the 'job-completed' event is generated, in order to keep the events disjoint.

627
628 A client that wants to subscribe to all job state changes, including creation and completion, includes
629 the 'job-created', 'job-state-changed', and 'job-completed' in the notification subscription. When a
630 job is finally removed from the Job History (see [ipp-mod] 4.3.7.1) no event is generated, i.e.,
631 neither a 'job-state-changed' event nor a 'job-purged' event is generated.

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

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

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

643
644 The standard Printer event keywords values are:

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

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

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

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

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

657 **'printer-state-changed'**: REQUIRED - the Printer changed state, i.e., the value of the Printer's
658 "printer-state", "printer-state-reasons" (whether "printer-state" changed or not), and/or "printer-is-
659 accepting-jobs" attributes changed, except when the Printer starts up or is shutdown. If printer state
660 reasons are added when the Printer is started up, only the 'printer-restarted' event is generated, in
661 order to keep the events disjoint. If printer state reasons are added or removed when the printer is
662 powered-down or shutdown, only the 'printer-shutdown' event is generated, in order to keep the
663 events disjoint.

664
665 A client that wants to subscribe to all printer state changes, including restart and power-
666 down/shutdown, includes the 'printer-restarted', 'printer-state-changed', and 'printer-shutdown' in the
667 notification subscription.

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

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

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

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

691 **'printer-almost-idle'**: OPTIONAL – when the Printer needs another Job in order to stay busy. This
692 event is used when a spooler is feeding more than one printer/server (fan-out), and the spooler holds

693 jobs until a Printer requests them, rather than committing jobs to IPP Printers before it is necessary.
 694 This event MAY be used by a Printer implementation to request a new job from any subscribers
 695 sufficiently ahead of time so that the device does not run out of work between jobs.
 696

697 5.3 notify-content-type (mimeMediaType)

698 This REQUIRED [READ-ONLY Subscription object](#) attribute specifies the type of content that is to be sent
 699 in the notifications. Thus the subscriber can control whether the event notification content is Human
 700 [Readable-Consumable](#) or Machine [ReadableConsumable](#). Most delivery methods support both forms of
 701 content type and MUST define one of them as the default. If a delivery method only supports one form of
 702 content type, then that form is also the default.

703 If the MIME media type registration permits a charset parameter, then such a specification MUST be used
 704 (instead of the "notify-attributes-charset" attribute) in order to indicate the charset to be used in the
 705 notification content.

706 If the Subscriber did not supply this attribute when requesting the subscription, the Printer object populates
 707 this attribute from the default value defined for the delivery method.

708 Standard mimeMediaType values are:

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

711
 712 'text/plain; charset=utf-8': A plain text document in ISO 10646 represented as UTF-8 [RFC-2044]
 713 as defined in section 7.

714 **ISSUE 2 - Should notify-content-type (mimeMediaType) be changed to notify-content-type (type2**
 715 **keyword), with values: 'human-consumable' and 'machine-readable'? Then content types that are neither**
 716 **'application/ipp' or 'text/plain; charset=utf-8' could be handled without having to be registered. For**
 717 **example, the 'snmp-ipp' is a trap MIB Machine Consumable format wouldn't not need to be registered (or**
 718 **have some special rule since the 'snmp-ipp' value doesn't have a registered MIME type and doesn't fit either**
 719 **'application/ipp' or 'text/plain'. Another advantage is that the charset would always be indicated in the**
 720 **"notify-attributes-charset" operation attribute, even for plain text.**

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

727 5.4 subscriber-user-data (octetString(63))

728 This REQUIRED [READ-ONLY Subscription object](#) attribute holds opaque information being sent from
 729 the Subscriber to the Notification Recipient, such as the identify of the Subscriber or a path or index to
 730 some Subscriber information. Or it MAY contain a key that the Notification Recipient needs in order to
 731 process the Notification, such as the ultimate recipient, if the Notification Recipient is a general application
 732 that in turn forwards notifications and the ultimate recipient isn't included in the value of the "notify-

733 recipient" attribute. An Instant Messaging Service is an example of such a general application where the
 734 "subscriber-user-data" might be the user's id for that messaging service and the "notification-recipient" is
 735 the URL of the messaging service.

736 5.5 notify-attributes-charset (charset)

737 This OPTIONAL [READ-ONLY Subscription object](#) attribute specifies the charset to be used in the
 738 Notification content sent to the notification recipient. This attribute MUST NOT be used when the "notify-
 739 content-type" attribute value specifies the charset parameter in its MIME media type value.

740 **ISSUE 4 - Should the Subscription object attributes "notify-attributes-charset (charset)" and "notify-
 741 attributes-natural-languages (naturalLanguage)" be renamed to simply "attributes-charset (charset)" and
 742 "attributes-natural-languages (naturalLanguage)" so that all IPP requests and objects have the same names?
 743 Then the "notify-attributes-charset (charset)" and "notify-attributes-natural-languages (naturalLanguage)"
 744 would only be operation attributes that set or reflect the "attributes-charset (charset)" and "attributes-
 745 natural-languages (naturalLanguage)" Subscription object attributes.**

746 5.6 notify-attributes-natural-languages (naturalLanguage)

747 This OPTIONAL [READ-ONLY Subscription object](#) attribute specifies the natural language for the IPP
 748 object to use in the Notification content that is sent to the Notification Recipient.

749 5.7 request-id

750 This REQUIRED READ-ONLY [Subscription object](#) attribute holds the most recent request-id sequence
 751 number delivered in a Notification content to the Notification Recipient. A value of 0 indicates that no
 752 Notifications have been sent for this subscription. The first request-id sent for a subscription MUST be 1.
 753 Each Notification Recipient has its own monotonically increasing series of request-ids, i.e., no gaps, in
 754 order to be able to detect a missing notification.

755 ~~For a Per Job subscription, "request-id" is a READ-ONLY member attribute of each collection value. All
 756 of the other member attributes (see section 5.1) are not READ-ONLY and MAY be modified by a Set Job
 757 Attributes operation (see 7.4).~~

758 ~~5.4 Additional attributes for the Subscription object only~~

759 ~~The following sub-sections define additional attributes that are only for the Subscription object and have no
 760 counterpart in Per Job subscriptions.~~

761 5.8 subscription-id (integer (1:MAX))

762 This REQUIRED READ-ONLY Subscription object attribute uniquely identifies this Subscription object
 763 instance on this Printer [object or this Job object](#). The Printer object, on acceptance of a Create-[Printer-
 764 Subscription request](#), generates an ID which identifies the new Subscription object on that Printer [or Job](#).
 765 The Printer returns the value of the "subscription-id" attribute as part of the response to a Create-[Printer-
 766 Subscription request](#). The 0 value is not included to allow for compatibility with "job-id" and with SNMP
 767 index values which also cannot be 0.

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

776 **5.9 notify-lease-expiration-time (integer(0:MAX))**

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

783 A client is able to extend a lease of a Per-Printer subscription using the Renew-~~Printer~~-Subscription
784 operation (see section 8.2.4). A value of 0 indicates an infinite time, if such a policy is supported as
785 indicated in the "notify-lease-time-supported" (integer(0:MAX)) Printer Description attribute (see section
786 6.7) and the subscriber is authorized to request an infinite lease. A Per-Job subscription cannot be renewed.

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

790 **5.10 printer-uri (uri)**

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

795 **5.11 subscriber-user-name (name(MAX))**

796 This REQUIRED READ-ONLY Subscription object attribute contains the name of the user that created the
797 Subscription object. The Printer object sets this attribute to the most authenticated printable name that it
798 can obtain from the authentication service over which the IPP operation was received. This attribute is
799 intended to help a human user determine for which Per-Printer Subscriptions they are the Subscriber. Only
800 if such is not available, does the Printer object use the value supplied by the client in the "requesting-user-
801 name" operation attribute of the create operation (see [IPP-MOD] Sections 4.4.2, 4.4.3, and 8). For Per-Job
802 subscriptions created as part of the Job creation operation, the value of the "subscriber-user-name" is the
803 same as the "job-originating-user-name" Job attribute (see [ipp-mod] section 4.3.6).

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

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

810 5.12 notify-printer-up-time (integer(1:MAX))

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

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

821 6 Printer Description Attributes related to Notification

822 This section defines the Printer Description attributes that are related to Notification.

823 **Table 5 - 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
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

824 **ISSUE 5 - Shouldn't we add a "number-of-printer-subscriptions" Printer Description attribute, since the**
 825 **client is NOT assured that it can determine the number of outstanding Per-Printer Subscriptions by doing**
 826 **Get-Subscriptions because of access control?**

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

831 **6.1 notify-schemes-supported (1setOf uriScheme)**

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

834 **6.2 notify-events-default (1setOf type2 keyword)**

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

838 **6.3 notify-events-supported (1setOf type2 keyword)**

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

841 **6.4 max-events-supported (integer(5:MAX))**

842 This REQUIRED Printer attribute specifies the maximum number of events that are supported in a single
843 Per-Job or Per-Printer subscription which must be the same. A Printer MUST support at least 5 events per
844 subscription, so that clients can depend on at least 5 events in a single subscription. If the number of events
845 supplied by a client in a subscription exceed this number, the Printer rejects the request and returns the
846 'server-error-too-many-events' (see section 14.3). If notification is not supported, this attribute MUST NOT
847 be supported.

848 **6.5 max-job-subscriptions-supported (integer(1:MAX))**

849 This REQUIRED Printer attribute specifies the maximum number of Per-Job subscriptions that are
850 supported in a single Job Creation request, i.e., the maximum number of collection values for the "job-
851 notify" operation attribute, and/or subsequent Create-Subscription operations adding Per-Job subscriptions
852 to the job object. A value of 0 indicates no effective maximum. A Printer MUST support at least 1 Per-Job
853 subscription. If the number of Per-Job subscriptions supplied by a client in a Job Creation request, the
854 Printer accepts the Job Creation and ignores the excess subscriptions. If a -subsequent Create-Subscription
855 request would exceed this number, the Printer rejects the request and returns the 'server-error-too-many-
856 subscriptions' (see section 14.2).

857 **6.6 max-printer-subscriptions-supported (integer(0:MAX))**

858 This REQUIRED Printer attribute specifies the maximum number of Per-Printer subscriptions that are
859 supported by multiple Create-Printer-Subscription requests, i.e., the maximum number of un-expired Per-
860 Printer Subscription objects that the Printer supports at a time. A value of 0 indicates no effective
861 maximum. A Printer MUST support at least 1 Per-Printer subscription. If the number of Per-Printer

862 subscriptions would exceed this number, the Printer rejects the Create-~~Printer~~-Subscription request and
863 returns the 'server-error-too-many-subscriptions' (see section 14.2).

864 **6.7 notify-lease-time-supported (rangeOfInteger(0:MAX))**

865 This REQUIRED Printer attribute specifies the range of values in seconds that are supported for the
866 "notify-lease-time-requested" operation attribute in a Create-~~Printer~~-Subscription or Renew-~~Printer~~-
867 Subscription request [for a Per-Printer subscription](#). When the lease time expires for a Per-Printer
868 Subscription without renewing, the Printer MUST delete the Subscription object. If the client requests a
869 value outside this range, the Printer MUST grant a value that is in this range (see section 5.9). A value of 0
870 indicates an infinite lease, i.e., one that does not expire.

871 **6.8 notify-lease-time-default (integer(0:MAX))**

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

875 **6.9 persistent-jobs-supported (boolean)**

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

880 **6.10 persistent-subscriptions-supported (boolean)**

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

886 **6.11 printer-state-change-time (integer(1:MAX))**

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

890 **6.12 printer-state-change-date-time (dateTime)**

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

894 7 Notification Content

895 This section defines the Notification content that is sent to a Notification Recipient when an event occurs.
896 The Notification MAY be sent by the IPP Printer or a third party Notification Service (see section 2.3).
897 There are two notification content types: Machine Consumable and Human Consumable. For most
898 notification delivery methods both content types are defined. Some of the notification delivery methods are
899 defined to support only one content type. A Printer MUST support all content types defined for a
900 notification delivery method, if it supports that notification delivery method.

901 7.1 Notification content MIME media type formats

902 This section defines the Notification content that the Notification Source sends asynchronously to each
903 Notification Recipient based on the subscription information stored with the subscription.

904 Notifications are generated using the following content formats:

905 **'application/ipp'** - Machine Consumable notification content using the 'application/ipp' MIME
906 media type [ipp-mod] using the Get-Job-Attributes response encoding for job events and Get-
907 Printer-Attributes for printer events. The attributes listed in sections 7.2 and 7.3 are sent in an
908 notification for Job events. The attributes listed in sections 7.2 and 7.4 are sent in an
909 notification for Printer events. For any ~~string~~ **'text' or 'name' attribute value** in any notification,
910 the charset and natural language rules that apply to all IPP operations apply to these **attributes**
911 ~~notification strings~~ as well, since they are represented as operation responses. The Unsupported
912 Attributes Group in the response is not sent. If the values of any of the attributes sent in an
913 notification content are not known, the value sent in the report content is the out-of-band
914 'unknown' value, rather than omitting the attribute (see the beginning of [ipp-mod] section 4.1).
915 An implementation MAY extend the contents of the Machine Consumable notification by
916 adding additional attributes.

917 **'text/plain; charset=utf-8'** - Human Consumable notification content type [RFC2046]. The text
918 message SHOULD include information about the attributes in sections 7.2 and 7.3 for job
919 events or in sections 7.2 and 7.4 for printer events. This information is localized according to
920 the information about natural language and charset in the subscription.

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

923 7.2 Notification content attributes common to Job and Printer events

924 This section lists the parameters and attributes that are included in both Job and Printer event Notifications.
925 Some events do not include all of these attributes as shown in Table 6. Each notification content contains a
926 single Job or Printer event, whether that event was subscribed using the Job Submission Subscription
927 mechanism or the Per-Printer subscription mechanism. If either kind of subscription subscribed to both Job
928 and Printer events, then they will be sent as separate Job notification content and Printer notification
929 contents to the same Notification Recipient. **References of the form "mod m.n.o" refer to [ipp-mod]**
930 **sections.**

931

Table 6 - 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**
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		R
15. subscriber-user-data (octetString(63))	5.4		R

932 Attribute Notes:

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

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

935 "attributes-charset" - the value comes from the "notify-attributes-charset" attribute in the "~~job-notify~~ Per-
936 ~~Job Subscription and the notify-attributes-charset in the Per Printer~~ Subscription object.937 "attributes-natural-language" - the value comes from the "notify-attributes-natural-language" attribute in the
938 "~~job-notify~~ Per-~~Job Subscription and the notify-attributes-natural-language in the Per Printer~~
939 Subscription object.

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

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

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

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

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

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

947 "subscription-id" - the unique identifier for the Subscription object on this Printer~~for Per-Job subscriptions,~~
 948 ~~the value is the index of the "job-notify" collection value, starting with 1 for the first collection~~
 949 ~~value.~~

950 "subscriber-user-name" - the subscriber user name that created the Subscription object~~comes from~~
 951 ~~"requesting-user-name" Job attribute for Per-Job subscriptions.~~ SNMP delivery method can
 952 truncate to less than 255 octets, since the Notification needs to fit into 484 octets or so on some
 953 transports that SNMP is defined for.

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

956 **7.3 Additional Notification content attributes for Job events only**

957 This section lists the additional attributes that are included only in Job event Notifications. Some events do
 958 not include all of these attributes as shown in Table 7.

959 **Table 7 - 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		R	R
17. job-state-reasons (1setOf type2 keyword)	mod 4.3.8		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	O	O	
20. job-media-sheets-completed (integer(0:MAX))	mod 4.3.18.3	O	O	
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		

960 **7.4 Additional Notification content attributes for Printer events only**

961 Table 8 lists the additional attributes that are included only in Printer event Notifications.

962 **Table 8 - Additional Notification content attributes for Printer events only**

		Events

Attributes	Reference	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

963

964 8 Operations for notification

965 [This section defines all of the operations for notification.](#)

966 8.1 Operations for Per-Job Subscriptions *only*

967 This section defines the REQUIRED operation requests and responses that are related to Per-Job
 968 subscriptions. Section 8.2 defines the REQUIRED operation requests and responses associated with the
 969 REQUIRED Per-Printer subscription and its Subscription object.

970 8.1.1 Job Creation Operations (Create-Job, Print-Job, Print-URI) and Validate-Job

971 The usual method for a client to associate one subscription with a Job is to specify the subscription when
 972 the job is created. For a Per-Job Subscription, the client supplies the "job-notify (1setOf collection)"
 973 operation attribute with the member attributes listed in Table 9 with any of the job creation operations
 974 (Create-Job, Print-Job, Print-URI), plus Validate-Job (which doesn't create a job or subscription). If the
 975 client does not supply the "job-notify" attribute in the create operation, there is no subscription made (either
 976 implicitly or explicitly).

977 If a Printer does not support this notification specification, then it MUST ignore the "job-notify" operation
 978 attribute and return it in the response indicated as an attribute that is not supported. See [ipp-mod] section
 979 3.1.7 for details on returning Unsupported Attributes.

980 **Table 9 - 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 (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

981

982 [See the referenced sections for a definition of these operation attributes, since they are copied to the Job](#)
 983 [object as the Job Description attributes described in section 5.](#)

984 The following rules apply to Per-Job subscriptions:

- 985 1. Any subscription can contain job events, printer events, or both.
- 986 2. The Job Submission Subscription is only valid while the job is "not-completed". The job is "not-
987 completed" while it is in the 'pending', 'pending-held', 'processing', and 'processing-stopped' states. The
988 job changes from being "not-completed" to "retained" when it is done processing and enters any of the
989 'completed', 'canceled', or 'aborted' states. The job becomes "not-completed" again when it is restarted
990 using the Restart-Job operation (see [ipp-mod]).
- 991 3. Since no job is created for the Validate-Job operation, the only purpose of supplying the subscription
992 operation attributes in the Validate-Job operation is to validate that the values are supported; the Printer
993 object does not establish a notification subscription as a result of the Validate-Job operation.
- 994 4. Since a Job Submission Subscription is included within a job submission operation, any interest in job
995 events is limited to "this job" only (the Job object created because of this job creation operation). There
996 is no mechanism to subscribe to events for all jobs or specifically some job other than this job in a
997 create operation. But see the Create-~~Printer~~ Subscription operation (section 8.2.1) for an explicit
998 operation to subscribe for job and/or printer events independently of any particular job submission.
- 999 5. Event reporting only occurs when a notification recipient has specified a subscription to any event(s).
- 1000 6. The notification implementation MAY allow an administrator to configure a policy on what events may
1001 be dropped.
- 1002 7. If the OPTIONAL "notify-attributes-charset" attribute is not supported or the supplied value is not
1003 supported, the IPP Printer MUST return the attribute in the Unsupported Attributes Group but still
1004 accept the operation, as with all Job create operations. In this case, the Printer MUST use the natural
1005 language supplied in the "attributes-charset" Job creation operation attribute, if that natural language
1006 value is supported by the Printer, else the Printer object MUST use the Printer's "charset-configured"
1007 value. See the Print-Job operation in [ipp-mod].
- 1008 8. If the OPTIONAL "notify-attributes-natural-language" attribute is not supported or the supplied value is
1009 not supported, the IPP Printer MUST return the attribute in the Unsupported Attributes Group but still
1010 accept the operation, as with all Job create operations. In this case, the Printer MUST use the natural
1011 language supplied in the "attributes-natural-language" Job creation operation attribute, if that natural
1012 language value is supported by the Printer, else the Printer object MUST use the Printer's "natural-
1013 language-configured" value. See the Print-Job operation in [ipp-mod].
- 1014 9. If a collection contains other unrecognized, unsupported member attributes and/or conflicting value, the
1015 attribute returned in the Unsupported Group is a collection containing the unrecognized, unsupported
1016 member attributes, and/or conflicting values. The unrecognized member attributes have an out-of-band
1017 value of unsupported. The unsupported member attributes and conflicting values have their unsupported
1018 values. See [ipp-coll].

1019 ~~8.2 Get-Printer-Attributes operation~~

1020 ~~The REQUIRED (by [ipp-mod]) Get-Printer-Attributes can be used to request the additional Printer~~
1021 ~~Description attributes that indicate the notification supported. See section 6.~~

1022 ~~8.3 Get-Job-Attributes operation~~

1023 ~~The REQUIRED (by [ipp-mod]) Get-Job-Attributes can be used to request the additional Job Description~~
1024 ~~attributes that relate to notification. See section 6.~~

8.4 Set-Job-Attributes operation

The ~~OPTIONAL Set-Job-Attributes operation (see [ipp-set2]) allows a client to change the Job attributes of the Job's Per-Job subscription, including to add them in the first place if they had not been supplied in the Job creation operation and to remove them from the Job object. This operation replaces all the collection values of the supplied "job-notify" (1setOf collection) notification attribute.~~

~~In order to cancel all Per-Job subscriptions, the client supplies the "job-notify" attribute with an out-of-band 'none' value (see [ipp-coll]).~~

~~Request:-~~

Group 1: Operation Attributes		Printer support
"attributes-charset" (charset)		R
"attributes-natural-language" (naturalLanguage)		R
"printer-uri" (uri)		R
"requesting-user-name" (name(MAX))		RECOMMENDED
"job-notify" (1setOf collection)		R

~~Response:-~~

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

~~The "job-notify" operation attribute is copied to the "job-notify" Job Description attribute. See section 1.1 for a definition of the "job-notify" Job Description attribute.~~

8.2 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. ~~lists the REQUIRED Per-Printer Subscriptions operation requests and responses and the Subscription object that MUST be supported.~~

1.1.18.2.1 Create-Printer-Subscription operation

The REQUIRED Create-~~Printer~~-Subscription operation creates either a Per-Job or a Per-Printer subscription by creating a Per-Printer Subscription object which is added to the Printer 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, 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.

1047 The Printer returns a subscription id and the length of time for which it has granted a lease for the
1048 subscription. If the requested subscription object is a Per-Job subscription, the Printer MUST grant an
1049 infinite lease by returning a 0 value for the "notify-lease-time-granted".

1050 The client must renew the Per-Printer subscription using the Renew-~~Printer~~-Subscription operation (see
1051 section 8.2.4) before the lease runs out in order to maintain the subscription. A client can unsubscribe
1052 using the Cancel-~~Printer~~-Subscription operation (section 8.2.5) and the subscription id.

1053 Two Create-~~Printer~~-Subscription operations with the same events and same Notification Recipient MUST
1054 be kept as distinct subscriptions and be assigned distinct subscription ids. A Printer MUST allow such
1055 duplicate subscriptions such that Cancel-~~Printer~~-Subscription doesn't unsubscribe both subscriptions and
1056 MUST send the Notifications twice to the Notification Recipient, since the "request-id" is supposed to
1057 count monotonically for each subscription.

1058 If the Printer has a bounded set of concurrent subscriptions), the printer rejects the operation and returns the
1059 'server-error-too-many-subscriptions' status code. The client SHOULD try again later.

1060 Access Rights: To create Per-Job subscription objects, the authenticated user (see [IPP-MOD] section 8.3)
1061 performing this operation MUST either be the job owner or have operator or administrator access rights for
1062 the Printer object (see [IPP-MOD] sections 1 and 8.5). To create Per-Printer subscription objects, the
1063 authenticated user performing this operation MUST have Per-Printer subscription rights for this Printer.
1064 Otherwise the IPP object MUST reject the operation and return: client-error-forbidden, client-error-not-
1065 authenticated, and client-error-not-authorized as appropriate.

1066 Request:

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

1067 Response:

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

1068 Attribute Notes:

1069 "job-id" (integer(1:MAX)) - the client supplies this attribute in order to create a Per-Job subscription for the
 1070 Job identified by the "job-id" value. If "job-id" is not supplied, then the subscription is a Per-Printer
 1071 subscription.

1072 "notify-recipient" (uri) - the client MUST supply this attribute in order to have a subscription.

1073 "notify-lease-time-requested" (integer(0:MAX)) - the number of seconds requested for the Per-Printer
 1074 subscription lease. A value of 0 indicates a request that the Per-Printer Subscription lease never
 1075 expire. Supplying a 0 value MAY require authentication in order to be used, if 0 is supported at all.
 1076 If the client does not supply this attribute, the Printer uses its "notify-lease-time-default" value (see
 1077 section 6.8).

1078 "notify-persistence-requested" (boolean) - whether or not the Per-Printer Subscription is to be persistent,
 1079 i.e., saved across power cycles. Note: Persistent trap registrations is a client option in SNMPv3
 1080 [RFC2573].

1081 "notify-lease-expiration-time" (integer(0:MAX)) - The Printer object MUST return this attribute which is
 1082 the time in the future at which the subscription lease will expire, i.e., the "printer-up-time" value (in
 1083 time ticks - see [ipp-mod] section 4.4.29) at which the Printer will delete the Subscription. A value
 1084 of 0 indicates that the lease subscription will never expire. If the Subscription object is a Per-Job
 1085 subscription, the Printer MUST return 0 indicating that the lease doesn't expire, since Per-Job
 1086 subscriptions expire when the Job object is no longer retained.

1087 "notify-lease-time-granted" (integer(0:MAX)) - The Printer object MUST return this attribute which is the
 1088 number of seconds of time actually granted. The value of "notify-lease-time-granted" may be less
 1089 than the requester requested in the "notify-lease-time-requested" if it was greater than the MAX
 1090 supported or more than the requester requested if it was less than the MIN supported, as indicated in
 1091 the Printer's "notify-lease-time-supported" (rangeOfInteger(0:MAX)) attribute. If the Subscription
 1092 object is a Per-Job subscription, the Printer MUST return 0 indicating that the lease doesn't expire,
 1093 since Per-Job subscriptions expire when the Job object is no longer retained.

1094 **ISSUE 6** - Instead of Create-Subscription response returning "notify-lease-time-granted" which is NOT a
 1095 Subscription object attribute, it is cleaner for implementation to return the "notify-printer-up-time"
 1096 Subscription attribute instead, which is a Subscription object attribute. The client subtracts the "notify-
 1097 printer-up-time" from the "notify-lease-expiration-time" to get the number of second remaining in the lease,
 1098 just as it would for a Get-Subscript-Attributes response. OK to replace the "notify-lease-time-granted" with
 1099 the "notify-printer-up-time" attribute in the response?

1100 "notify-persistence-granted" - whether or not this ~~Per-Printer~~ Subscription object instance is persistent.
 1101 This attribute MUST be returned whether "notify-persistence-requested" is supported or not, so that
 1102 the client knows which.

1103 **ISSUE 7** - Should "notify-persistence-granted" be a Subscription object attribute too, so that all attributes
 1104 returned in a response are Subscription object attributes?

1105 1.1.28.2.2 **Get-Printer-Subscription-Attributes operation**

1106 The REQUIRED Get-~~Printer~~ Subscription-Attributes returns the requested attributes of the identified ~~Per-~~
 1107 ~~Printer~~ Subscription object. See sections ~~5~~ and ~~5.4~~.

1108 Request:

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

1109 Response:

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

1110 This operation is similar to the Get-Printer-Attributes operation. If the client omits the "requested-
 1111 attributes" operation attribute, the Printer MUST respond as if the client had supplied the 'all' value, i.e.,
 1112 return all of the attributes supported for the Subscription object.

1113 **1.1.38.2.3 Get-Printer-Subscriptions operation**

1114 The REQUIRED Get-Printer-Subscriptions operation returns Per-Printer subscriptions, i.e.,, the requested
 1115 attributes of the Subscription objects (see sections 5.5.3 and 5.4).

1116 Request:

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

1117 Response:

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

1118
 1119 Attribute Notes:

1120 This operation is similar to the Get-Jobs operation (see [ipp-mod]). If the client wants any attributes
 1121 returned, including the "subscription-id", it must include the attribute keyword name in the "requested-
 1122 attributes" operation attribute. If the "requested-attributes" operation attribute is omitted, the Printer MUST
 1123 respond as if the client supplied the value: 'subscription-id'.

1124 "job-id" (integer(1:MAX)) - If the client supplies this attribute, all of the Per-Job Subscription objects for
 1125 the identified job are candidates for return. If this attribute is omitted, all of the Per-Printer
 1126 Subscription objects are candidates for return.

1127 "my-subscriptions" (boolean) - If the client supplies the "my-subscriptions" with a 'false' value or omits it,
 1128 the Printer returns all subscriptions, subject to the security policy in force.

1129 Groups 3 to N: Subscription Object Attributes: The Printer object responds with one set of Subscription
 1130 Object Attributes for each returned Subscription object. The Printer object ignores (does not
 1131 respond with) any requested attribute or value which is not supported or which is restricted by the
 1132 security policy in force, including whether the requesting user is the user that created the
 1133 Subscription object (subscribing user) or not (see [ipp-mod] section 8).

1134 **1.1.48.2.4 Renew-Printer-Subscription operation**

1135 The REQUIRED Renew-Printer-Subscription operation permits a client to request the IPP Printer to extend
 1136 the lease on a Per-Printer-Subscription object instance. There is no way to renew a Per-Job subscription,
 1137 since they are automatically canceled after the job completes and no longer has any documents, i.e., the job
 1138 is no longer retained (see [ipp-mod] section 4.3.7.2). If the requested subscription object is a Per-Job
 1139 subscription, the Printer MUST grant an infinite lease by returning a 0 value for the "notify-lease-time-
 1140 granted".

1141 *Access Rights:* The authenticated user (see [IPP-MOD] section 8.3) performing this operation MUST either
 1142 be the owner of the Subscription object or have operator or administrator access rights for the Printer object
 1143 (see [IPP-MOD] sections 1 and 8.5). Otherwise the IPP object MUST reject the operation and return:
 1144 client-error-forbidden, client-error-not-authenticated, and client-error-not-authorized as appropriate.

1145 Request :

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

1146 Response :

Group 1: Operation Attributes	Printer support
"status-code" (type2 enum)	R
"attributes-charset" (charset)	R
"attributes-natural-language" (naturalLanguage)	R
["status-message" (text(255))]	O
["detailed-status-message" (text(MAX))]	O
"notify-lease-expiration-time" (integer(0:MAX))	R
"notify-lease-time-granted" (integer(0:MAX))	R
Group 2: Unsupported Attributes	R

1147 Attribute Notes:

1148 "notify-lease-time-requested" (integer(0:MAX) - the number of seconds requested for the Per-Printer
 1149 subscription lease. Same as Create-Printer-Subscriptions (see section 8.2.1).

1150 "notify-lease-expiration-time" - the time in the future when the subscription will expire. Same as for the
 1151 Create-Printer-Subscription operation (see section 8.1.1).

1152 "notify-lease-time-granted" - the lease time (number of seconds) actually granted. Same as for the Create-
 1153 Printer-Subscription operation (see section 8.1.1).

1154 **ISSUE 8 - Instead of Renew-Subscription response returning "notify-lease-time-granted" which is NOT a**
 1155 **Subscription object attribute, it is cleaner for implementation to return the "notify-printer-up-time"**
 1156 **Subscription attribute instead, which is a Subscription object attribute. Ok to replace the "notify-lease-**
 1157 **time-granted" with the "notify-printer-up-time" attribute in the response?**

1158 Note: There is no way to change any of the Subscription object attributes, except the "notify-lease-
 1159 expiration-time" attribute. In order to change other attributes, a client can create a new Subscription object
 1160 and then use the Cancel-Printer-Subscription operation to cancel the old one (or do this in the other order,
 1161 in case there is a limit on the number of Subscription object instances, as long as a short window with no
 1162 Notifications is ok).

1163 Note: There is no need to renew a Per-Job Subscription, since it is effectively the time that the Job is active
 1164 (see section 8.1.1).

1165 **1.1.58.2.5 Cancel- Printer-Subscription operation**

1166 The REQUIRED Cancel- Printer-Subscription operation allows a client to remove a Per Printer
 1167 Subscription object from the Printer. No more Notifications are delivered for that Subscription. Once
 1168 performed, there is no way to use that Subscription in the future. Subscription-ids should not be reused
 1169 immediately, so that a stale reference situation is not created. Same as for Cancel-Job and job-ids.

1170 *Access Rights:* The authenticated user (see [IPP-MOD] section 8.3) performing this operation MUST either
 1171 be the owner of the Subscription object or have operator or administrator access rights for the Printer object
 1172 (see [IPP-MOD] sections 1 and 8.5). Otherwise the IPP object MUST reject the operation and return:
 1173 client-error-forbidden, client-error-not-authenticated, and client-error-not-authorized as appropriate.

1174 Request :

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

1175 Response :

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

1176

9 Comparison of Per-Job ~~Submission Subscriptions~~ 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-Subscription operation by specifying the "job-id" of the job. A Per-Printer Subscription is established between a client and a Printer using the ~~new~~ 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 Create-Subscription operation.
2. For Per-Job subscriptions, the subscription is only valid while the job is "not-complete" (see sections 7.2 and 7.3); while for the Per-Printer subscriptions, the subscription is valid until ~~it is explicitly canceled with a Cancel-Printer-Subscription operation or~~ the time (in seconds) that the Printer returned in the "notify-lease-time-granted" operation attribute expires, ~~whichever occurs first.~~
3. Job Events in a Per-Job subscription apply only to "one job" (the Job created by the job creation operation); while Job Events in a Per-Printer-~~subscription~~ apply to ALL jobs contained in the IPP Printer object.

10 Conformance Requirements

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

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

If IPP Notification is implemented, the operations described in this document must be supported as described in Table 10:

Table 10 - Conformance Requirements for Operations

Attribute	Conformance requirements
"job-notify" in Job Creation operations (section 8.1.1)	REQUIRED
Set Job Attributes (see [ipp-set2])	OPTIONAL
Create- Printer -Subscription (section 8.2.1)	REQUIRED
Get- Printer -Subscription-Attributes (section 8.2.2)	REQUIRED
Get- Printer -Subscriptions (section 8.2.3)	REQUIRED
Renew- Printer -Subscription (section 8.2.4)	REQUIRED
Cancel- Printer -Subscription (section 8.2.5)	REQUIRED

1207

1208 11 IANA Considerations

1209 IANA will be called on to register URL schemes for notification delivery for use in the "notification-
1210 recipient" attribute, using the same procedures outlined in [ipp-mod].

1211 12 Internationalization Considerations

1212 This IPP notification specification continues the internationalization of [ipp-mod] for attributes containing
1213 text strings and names. A subscribing client can specify a different natural language and charset for each
1214 Notification content delivered to a Notification Recipient.

1215 The Human Consumable Notification content is a 'text/plain; charset=utf-8' by default where the
1216 Notification Sender has localized the text message as requested by the subscriber for the intended
1217 Notification Recipient.

1218 13 Security Considerations

1219 By far the biggest security concern is the abuse of notification: sending unwanted notifications to third
1220 parties (i.e., spam). The problem is made worse by notification addresses that may be redistributed to
1221 multiple parties (e.g. mailing lists). There exist scenarios where third party notification is required (see
1222 Scenario #2 and #3 in [ipp-not-req]). The fully secure solution would require active agreement of all
1223 recipients before sending out anything. However, requirement #9 in [ipp-req] ("There is no requirement for
1224 IPP Printer receiving the print request to validate the identity of an event recipient") argues against this.
1225 Certain systems may decide to disallow third party notifications (a traditional fax model).

1226 Clients submitting notification requests to the IPP Printer has the same security issues as submitting an
1227 IPP/1.1 print job request. The same mechanisms used by IPP/1.1 can therefore be used by the client
1228 notification submission. Operations that require authentication can use the HTTP authentication.
1229 Operations that require privacy can use the HTTP/TLS privacy.

1230 The notification access control model should be similar to the IPP access control model for Jobs. Creating
1231 a Per-Printer Notification Subscription object is associated with a user. Only the creator or an operator can
1232 cancel the subscription. The system may limit the listing of items to only those items owned by the user.
1233 Some subscriptions (e.g. those that have a lifetime longer than a job) can be done only by privileged users
1234 (users having operator and/or administrator access rights), if that is the authorization policy.

1235 The standard security concerns (delivery to the right user, privacy of content, tamper proof content) apply
1236 to the notification delivery. IPP should use the security mechanism of the delivery method used. Some
1237 delivery mechanisms are more secure than others. Therefore, sensitive notifications should use the delivery
1238 method that has the strongest security.

1239 14 Status Codes

1240 The following status codes are defined as extensions for notification:

1241 **14.1 client-error-uri-notification-scheme-not-supported (0x04??)**

1242 The scheme of the client-supplied URI in a "notify-recipient" operation attribute in a Create-~~Printer~~
 1243 Subscription operation is not supported. See [ipp-mod] section 3.1.7.

1244 There is no corresponding Per-Job subscription error [for a Job Creation operation](#), since the Printer object
 1245 MUST ignore any errors in the "job-notify" operation attribute, MUST return the "notify-recipient"
 1246 attribute in the Unsupported Attributes Group, and perform the Job Creation operation (see section 1.1),
 1247 since the job can still be printed.

1248 **14.2 server-error-too-many-subscriptions (0x04??)**

1249 The bounded set of concurrent Per-Printer subscriptions supported by the Printer object would be exceeded
 1250 if this request were accepted.

1251 Note: There is no corresponding Per-Job subscription error, since the Printer object MUST ignore any
 1252 errors in the "job-notify" operation attribute and perform the Job Creation operation (see section 1.1), since
 1253 the job can still be printed.

1254 **14.3 server-error-too-many-events (0x04??)**

1255 The client supplied more events in the "notify-events" operation attribute in a Create-~~Printer~~ Subscription
 1256 that the Printer supports, as indicated in its "max-events-supported" attribute (see section 6.4).

1257 There is no corresponding Per-Job subscription error [for a Job Creation operation](#), since the Printer object
 1258 MUST ignore any errors in the "job-notify" operation attribute, MUST return the "notify-events" attribute
 1259 in the Unsupported Attributes Group with only the excess events that are being ignored, and perform the
 1260 Job Creation operation (see section 1.1), since the job can still be printed.

1261 **15 Additions to the IPP Encoding and Transport Document**

1262 The Subscription object tag needs to be assigned in section 3.7.1 Delimiter Tags:

1263 **3.7.1 Delimiter Tags**

1264 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

1265 When an xxx-attributes-tag occurs in the protocol, it MUST mean that zero or more following attributes up
1266 to the next delimiter tag are attributes belonging to group xxx as defined in the model document, where xxx
1267 is operation, job, printer, unsupported, subscription.

1268 Doing substitution for xxx in the above paragraph, this means the following. When an operation-attributes-
1269 tag occurs in the protocol, it MUST mean that the zero or more following attributes up to the next delimiter
1270 tag are operation attributes as defined in the model document. When an job-attributes-tag occurs in the
1271 protocol, it MUST mean that the zero or more following attributes up to the next delimiter tag are job
1272 attributes or job template attributes as defined in the model document. When a printer-attributes-tag occurs
1273 in the protocol, it MUST mean that the zero or more following attributes up to the next delimiter tag are
1274 printer attributes as defined in the model document. When an unsupported-attributes-tag occurs in the
1275 protocol, it MUST mean that the zero or more following attributes up to the next delimiter tag are
1276 unsupported attributes as defined in the model document. When a subscription-attributes-tag occurs in the
1277 protocol, it MUST mean that the zero or more following attributes up to the next delimiter tag are
1278 subscription attributes as defined in the [ipp-not] document.

1279 Add a reference to [ipp-not].

1280 16 References

1281 [ipp-coll]

1282 deBry, R., , Hastings, T., Herriot, R., "Internet Printing Protocol/1.0 & 1.1: collection attribute
1283 syntax", <draft-ietf-ipp-collection-00.doc>, work in progress, September 9, 1999.

1284 [ipp-mod]

1285 deBry, R., , Hastings, T., Herriot, R., Isaacson, S., Powell, P., "Internet Printing Protocol/1.1: Model
1286 and Semantics", < draft-ietf-ipp-model-v11-04.txt>, work in progress, June 23, 1999.

1287 [ipp-not-hist]

1288 deBry, R., Lewis, H., Hastings, T., "Internet Printing Protocol/1.1: Requirements for IPP
1289 Notifications", <draft-ietf-ipp-not-change-history-00.doc>, work in progress, August 22, 1999.

1290 [ipp-not-req]

1291 deBry, R., Lewis, H., Hastings, T., "Internet Printing Protocol/1.1: Requirements for IPP
1292 Notifications", <draft-ietf-ipp-not-03.txt>, work in progress, August 11, 1999.

1293 [ipp-set2]

1294 Kugler, C., , Hastings, T., "Internet Printing Protocol/1.0 & 1.1: Additional Operations, Set 2",
1295 <draft-ietf-ipp-set2.txt>, work in progress, August 22, 1999.

1296 [ipp-prog]

1297 Hastings, T., Bergman, R., Lewis, H., "Proposed Job Progress Attributes for IPP", <draft-ietf-ipp-
1298 prog.txt> work in progress, May 18, 1999.

1299 [RFC2046]

1300 Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types. N. Freed & N. Borenstein.
1301 November 1996. (Obsoletes RFC1521, RFC1522, RFC1590), RFC 2046.

- 1302 [RFC2119]
1303 S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119 , March 1997
- 1304 [RFC2566]
1305 deBry, R., , Hastings, T., Herriot, R., Isaacson, S., Powell, P., "Internet Printing Protocol/1.0: Model
1306 and Semantics", RFC 2566, April 1999.

1307 **17 Author's Addresses**

- 1308 Scott A. Isaacson (Editor)
1309 Novell, Inc.
1310 122 E 1700 S
1311 Provo, UT 84606
1312
1313 Phone: 801-861-7366
1314 Fax: 801-861-2517
1315 e-mail: sisacson@novell.com
1316
- 1317 Tom Hastings
1318 Xerox Corporation
1319 737 Hawaii St. ESAE 231
1320 El Segundo, CA 90245
1321
1322 Phone: 310-333-6413
1323 Fax: 310-333-5514
1324 e-mail: hastings@cp10.es.xerox.com
1325
- 1326 Roger deBry
1327 Utah Valley State College
1328 Orem, UT 84058
1329
1330 Phone: (801) 222-8000
1331 EMail: debryro@uvsc.edu
1332
- 1333 Jay Martin
1334 e-mail: jkm@underscore.com
1335
- 1336 Michael Shepherd
1337 Xerox Corporation
1338 800 Phillips Road MS 128-51E
1339 Webster, NY 14450
1340
1341 Phone: 716-422-2338
1342 Fax: 716-265-8871
1343 e-mail: mshepherd@crt.xerox.com

1344
1345 Ron Bergman (Editor)
1346 Dataproducts Corp.
1347 1757 Tapo Canyon Road
1348 Simi Valley, CA 93063-3394
1349
1350 Phone: 805-578-4421
1351 Fax: 805-578-4001
1352 Email: rbergman@dpc.com
1353

1354 **A. Appendix - Summary of Notification Attribute Usage**

1355 This appendix summarizes the usage of Notification attributes in the Job operation attribute collection, Job
1356 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	Set-Job-Subscription	Create- Printer -Subscription	Renew-Subscription	Cancel-Subscription
1. notify-recipient (uri)	R	R			[R]	[R]	[R]		
2. notify-events (1setOf type2 keyword)	R	R		100	[R]	[R]	[R]		
3. notify-content-type (mimeMediaType)	R	R			[R]	[R]	[R]		
4. subscriber-user-data (octetString(63))	R	R	R	R	[R]	[R]	[R]		
5. attributes-charset (charset)	R*		R (6)	R (6)	R	R	R	R	R
6. notify-attributes-charset (charset)	O	O			[O]	[O]	[O]		
7. attributes-natural-language (naturalLanguage)	R*		R (8)	R (8)	R	R	R	R	R
8. notify-attributes-natural-language (naturalLanguage)	O	O			[O]	[O]	[O]		
9. notify-lease-time-requested (integer(0:MAX))							[R]	[R]	
10. notify-lease-time-granted (integer(0:MAX))							Return	Return	
11. notify-lease-expiration-time (integer(0:MAX))		R*					Return	Return	
12. notify-persistence-requested (boolean)							[O]		
13. notify-persistence-granted (boolean)							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	R					
18. subscription-id (integer(1:MAX))	**	R	R	R			Return	R	R

1357 Legend: "R" - REQUIRED for Printer to support; "O" - OPTIONAL for Printer to support; "[]" - OPTIONAL for sender to send;
1358 Return - Printer MUST return; * - MUST be 0 for Per-Job Subscription objects; ~~*~~ - Job object, not collection;
1359 ~~** - 1setOf collection index is the implicit subscription id for a Per-Job subscription and is sent in the Notification.~~

1360

Attributes	Objects		Notification		Job operations		Subscription operations		
	Job	Subscription	Job Notification	Printer Notification	Job Creation Validate -Job	Set- Job- Subscription	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-(21)	R	R	R	R	R	R
23. printer-name (name(127))			R	R					
24. job-id (integer(1:MAX))	R		R	Per-Jb			Per-Jb		
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	R-(26)	R-(26)					
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							
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]						

1361 Legend: "R" - REQUIRED for Printer to support; "O" - OPTIONAL for Printer to support; [] - OPTIONAL for sender to send;
1362 "Per-Jb" - REQUIRED if Notification from a Per-Job subscription;
1363 "[pg/cp]" - OPTIONAL in a 'job-progress' or 'job-completed' notification; (nn) - an alternate source of the data in a Notification.

Attributes	Objects		Notification		Job operations		Subscription operations		
	Job object	Printer object	Job Notification	Printer Notification	Job Creation Validate- Job	Set- Job- Subscription	Create- Printer- Subscription	Renew - Subscription	Cancel - Printer- 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. max-job-subscriptions-supported (integer(1:MAX))		R							
47. max-printer-subscriptions-supported (integer(0:MAX))		R							
48. notify-lease-time-supported (rangeOfInteger(0:MAX))		R							
49. notify-lease-time-default (integer(0:MAX))		R							
50. persistent-jobs-supported (boolean)		O							
51. persistent-subscriptions-supported (boolean)		O							
52. printer-state-change-time (integer(1:MAX))		O							
53. printer-state-change-date-time (dateTime)		O							

1364 Legend: "R" - REQUIRED for Printer to support; "O" - OPTIONAL for Printer to support; "[]" - OPTIONAL for sender to send;

1365

1366 **18 Appendix C: Full Copyright Statement**

1367 Copyright (C) The Internet Society (1998,1999). All Rights Reserved

1368 This document and translations of it may be copied and furnished to others, and derivative works that
1369 comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and
1370 distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice
1371 and this paragraph are included on all such copies and derivative works. However, this document itself
1372 may not be modified in any way, such as by removing the copyright notice or references to the Internet
1373 Society or other Internet organizations, except as needed for the purpose of developing Internet standards in
1374 which case the procedures for copyrights defined in the Internet Standards process must be followed, or as
1375 required to translate it into languages other than English.

1376 The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its
1377 successors or assigns.

1378 This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET
1379 SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES,
1380 EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE
1381 OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED
1382 WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

1383